

Proyecto: HachComp

Especificación de Requisitos Software

(Formato IEEE Std. 830-1998)

Miembros del equipo:

Jaime Antonio Amaya Rengifo	Jon González Vilela
Kei Rodríguez Hachimaru	Marcos Sánchez Díez
Pablo Sánchez Redondo	Ricardo Saugar Algarra
Ainara Sautua Molina	Cristina Trueba Raya

Control de cambios

[illegible]

Índice

1 Introducción	4
1.1 Propósito	4
1.2 Alcance	4
1.3 Definiciones, acrónimos y abreviaturas	4
1.4 Referencias	4
Para el proyecto hemos empleado el “IEEE 1058 - 1998” del autor The Institute of Electrical and Electronics Engineers Inc. Que nos proporciona un conjunto de directrices para la creación, desarrollo y gestión de un plan de gestión de proyectos de software. Específicamente, se enfoca en cómo se deben documentar y organizar los aspectos clave de un proyecto de software, para asegurar su éxito a través de un control adecuado y un seguimiento efectivo., a fecha de 8 de diciembre de 1998, obtención del documento a través de la biblioteca con el siguiente link:	4
https://ucm.on.worldcat.org/oclc/956664922	4
1.5 Organización del documento	5
2 Descripción general	6
2.1 Perspectiva del producto	6
2.2 Funciones del producto	6
2.3 Características del usuario	6
2.4 Restricciones	6
2.5 Supuestos y dependencias	7
2.6 Requisitos futuros	7
3 Requisitos específicos	8
3.1 Interfaces externo	8
3.1.1 Interfaces hardware	8
3.1.2 Interfaces software	8
3.1.3 Interfaces de Comunicación	11
3.2 Requisitos funcionales	13
Subsistema 1	13
3.2.1 Subsistema 2	18
3.2.2 Subsistema 3 DEVOLUCIONES	22
3.2.3 Subsistema 4	26
3.2.4 Subsistema 5	31
3.2.5 Subsistema 6	36
3.2.6 Subsistema 7	40
3.2.7 Subsistema 8	45
3.3 Requisitos de rendimiento	50
3.4 Requisitos lógicos de la base de datos	50
3.5 Restricciones de diseño	51
3.6 Atributos del sistema software	51
3.7 Otros requisitos	52
4 Apéndices	53

1 Introducción

1.1 Propósito

- El propósito de este documento es detallar las funcionalidades de nuestro software.
- Va dirigido a todos aquellos emprendedores que necesiten un software de gestión de sus empleados y del inventario, concretamente a aquellos que están creando un negocio de venta de productos tecnológicos.
- El tipo de organización al que va destinado es a empresas pequeñas y medianas.

1.2 Alcance

- El nombre de nuestro software es HachComp (provisional).
- El software permitirá administrar los datos de los empleados de la empresa, administrar el inventario, añadiendo y quitando productos del stock según sea necesario (compra y venta) y también tendrá algunas funcionalidades extra como aplicar ofertas a los productos en venta.
- El software NO tendrá acceso a los proveedores para poder realizar un reabastecimiento del inventario y NO tendrá funcionalidades para los clientes, solo para empleados.

1.3 Definiciones, acrónimos y abreviaturas

- Stock: Inventario
- Restock: Reabastecimiento del inventario

1.4 Referencias

Para el proyecto hemos empleado el “IEEE 1058 - 1998” del autor The Institute of Electrical and Electronics Engineers Inc. Que nos proporciona un conjunto de directrices para la creación, desarrollo y gestión de un plan de gestión de proyectos de software. Específicamente, se enfoca en cómo se deben documentar y organizar los aspectos clave de un proyecto de software, para asegurar su éxito a través de un control adecuado y un seguimiento efectivo., a fecha de 8 de diciembre de 1998, obtención del documento a través de la biblioteca con el siguiente link:

<https://ucm.on.worldcat.org/oclc/956664922>

1.5 Organización del documento

Introducción: Explica el propósito del proyecto HachComp, define términos clave y acrónimos, incluye referencias relevantes y una guía sobre cómo navegar el documento.

Descripción General: Explica las funciones principales del software (como la gestión de empleados e inventario), las características de los usuarios y las restricciones del sistema y plantea supuestos y dependencias para su funcionamiento.

Requisitos Específicos: Explica los detalles técnicos y funcionales, como las interfaces externas, requisitos funcionales de subsistemas (como gestión de empleados, inventario, devoluciones), requisitos de rendimiento, lógicos de la base de datos y restricciones de diseño.

Atributos del Sistema Software: Enumera características esenciales, como seguridad, y usabilidad.

2 Descripción general

2.1 Perspectiva del producto

- Este software SI estará integrado con el resto de productos de la empresa, ya que permitirá la gestión y organización de estos.
- Para una correcta integración del programa con el resto de productos, se deberá garantizar acceso al sistema, a los usuarios y sobre todo a los datos de la empresa.
- El programa tendría restricciones como el acceso al contacto de proveedores, esto deberá gestionarse aparte ya que el programa solo gestiona los datos de empleados y del inventario.

2.2 Funciones del producto

- Gestión de datos de empleados:** permite dar de alta a nuevos empleados con sus datos, eliminarlos si es necesario, y acceder a los datos (DNI, edad, sexo, etc). Los empleados dados de alta tendrán acceso con sus credenciales al resto de funciones del producto.
- Gestión de inventario:** permite añadir a la base de datos del inventario nuevos productos adquiridos, eliminarlos si se venden o dañan.
- Modificación de datos:** se pueden modificar los datos tanto de los empleados como de los productos, como por ejemplo el precio de estos para añadir ofertas.
- Creación de categorías y productos:** Si a la hora de añadir un nuevo producto no existiese este o una categoría para él, estas pueden ser creadas.

2.3 Características del usuario

Este producto va dirigido a los empleados de una pequeña o mediana empresa de compraventa de productos tecnológicos. El uso del programa es muy sencillo, por lo que no será necesario ningún tipo de especialización técnica ni de conocimientos previos para su uso.

2.4 Restricciones

Seguridad y consideraciones de robustez: el programa no debe tener la capacidad de modificar en gran medida la estructura de las bases de datos, ya que en caso de hacerlo podría generar errores. Para garantizar una buena robustez y seguridad, el único caso en el que el programa podrá modificar la estructura de la base de datos, será a la hora de crear nuevas categorías para los productos que lo requieran.

2.5 Supuestos y dependencias

-Cambios esperados: la adquisición de nuevos productos para la empresa, además de la venta de estos, son cambios que se esperan que se produzcan, en caso de no producirse, el programa no tendría utilidad alguna, ya que este sirve para manejar el flujo de clientes, empleados y actividad de la tienda.

-Cambios imprevistos: en caso de que en un futuro la empresa decida proporcionar algún servicio adicional a la compraventa de productos informáticos, el programa no serviría para gestionarlo y sería necesaria su modificación o incluso la creación de uno nuevo.

2.6 Requisitos futuros

En caso de que la empresa decida proporcionar servicios adicionales a la compraventa, la gestión de estos o alguna utilidad para desarrollar el servicio, podrían ser implementados en el programa. Sin embargo, si esta actividad distase mucho de la compraventa tal vez deba crearse un programa nuevo para ella.

Si no se da el caso, el programa no necesitará actualizarse a versiones posteriores, a menos que sean necesarias para corregir errores que nos fuesen comunicados.

3 Requisitos específicos

3.1 Interfaces externo

3.1.1 Interfaces hardware

Interfaz de base de datos

Para conectarse a servidores de bases de datos que almacenen los datos de nuestro software, se necesitará servidores de bases de datos que pueden ser físicos, o discos duros o discos SSD.

Interfaz con dispositivos de red

Nuestro software requiere interfaces hardware para la comunicación de red entre los diferentes dispositivos y servidores.

Interfaz móvil

Necesitamos pantallas táctiles para interactuar con la aplicación y permitir a empleados y administradores navegar entre opciones como el inventario, la gestión de empleados u otras funcionalidades.

3.1.2 Interfaces software

Nombre: Consulta de datos

Descripción del propósito: Búsqueda de información que el usuario necesite, ya sea acerca de clientes, categorías, clientes etc.

Origen de entrada/destino de salida: Origen de entrada, teclado. Destino de salida, la pantalla del PC.

Precisión: Alta, deben especificarse adecuadamente los datos para obtener la información deseada.

Unidades de medida: No aplicable.

Temporización: La consulta debe ser relativamente rápida (al ser una base de datos no muy grande no habrá problema), menos de 5 segundos.

Relaciones con otras entradas/salidas: El resultado de la consulta se puede usar para las

interfaces de modificación. También, la salida de la consulta hace uso de las interfaces de muestra de mensaje y la de muestra de datos.

Organización/formatos de pantalla:

```
"-----"
"|          Resultado de la búsqueda          |"
"-----"
"|  #   | Type | Field1 | Field2 | ... | FieldN |"
"-----"
"| ..... | ..... | ..... | ..... | ..... | ..... |"
"-----"
```

Organización/formatos de ventana:

No habrá GUI.

Formato de fechas:

Formato de comandos:

"GET %type %fields" ej: GET USER DNI:Juan

Si no se especifica el %fields, o depende de cuantos %fields se especifiquen, se mostrarán todos los tipos de entidad que validen el %type y los %fields especificados (si los hay).

Mensajes de fin:

"Consulta exitosa"

"No existen datos relacionados con tu búsqueda"

Nombre: Modificación de campo

Descripción del propósito: Tras una búsqueda, permite seleccionar uno de los valores para modificar algunos o todos sus valores (los que estén permitidos).

Origen de entrada/destino de salida: Origen de entrada, teclado. Destino de salida, la pantalla del PC.

Precisión: Alta, deben especificarse adecuadamente los datos para modificar la información deseada.

Unidades de medida: No aplicable.

Temporización: La modificación debe ser relativamente rápida (al ser una base de datos no muy grande no habrá problema), menos de 5 segundos.

Relaciones con otras entradas/salidas: El resultado de la modificación se puede usar para las interfaces de guardado. Su entrada proviene de la interfaz de búsqueda. Para los mensajes y la muestra de los datos actualizados se relaciona con las interfaces de mensajes y de muestra de datos.

Organización/formatos de pantalla:

```

"-----"
"|          Dato actualizado          |"
"-----"
"|  #   | Type | Field1 | Field2 | ... | FieldN |"
"-----"
"| ..... | ..... | ..... | ..... | ..... | ..... |"
"-----"

```

Organización/formatos de ventana:

No habrá GUI.

Formato de fechas:**Formato de comandos:**

"MODIFY %# %fields" ej: MODIFY #1 NOMBRE:Marcos

Se especifica la fila, del valor a modificar, mostrando los resultados de la búsqueda hechos previamente.

Mensajes de fin:

"¿Desea guardar sus cambios?"

"Campos inválidos."

Nombre: Guardar cambios

Descripción del propósito: Tras una modificación, guardar en la base de datos los cambios realizados.

Origen de entrada/destino de salida: Origen de entrada, teclado. Destino de salida, la pantalla del PC.

Precisión: No aplicable.

Unidades de medida: No aplicable.

Temporización: El guardado debe ser relativamente rápido, menos de 5 segundos.

Relaciones con otras entradas/salidas: Su entrada proviene de la interfaz de modificación.

Esta interfaz permite guardar los cambios realizados. Tras el cambio, su salida se relaciona con la interfaz de mensajes.

Organización/formatos de pantalla: No aplicable

Organización/formatos de ventana:

No habrá GUI.

Formato de fechas:**Formato de comandos:**

"SAVE %y/%n"

SAVE y : guardar

SAVE n: no guardar, cancelar la modificación

Mensajes de fin:

"Cambios guardados"

"Modificación cancelada"

"Errores al guardar"

3.1.3 Interfaces de Comunicación

Nombre: Muestra de mensaje

Descripción del propósito: Mostrar los mensajes tras una operación o resultado de una interfaz.

Origen de entrada/destino de salida: Origen de entrada, teclado. Destino de salida, la pantalla del PC.

Precisión: No aplicable

Unidades de medida: No aplicable.

Temporización: Debe mostrar el mensaje rápidamente, menos de 1 segundo. Al ser texto plano no habrá problema.

Relaciones con otras entradas/salidas: Cualquier otra interfaz tendrá que mostrar un mensaje, así que sus salidas se relacionarán con esta.

Organización/formatos de pantalla:

```
"-----"  
"| Mensaje: %mensaje      |"  
"-----"
```

Organización/formatos de ventana:

No habrá GUI.

Formato de fechas:

Formato de comandos: No hay

Mensajes de fin: No hay

Nombre: Muestra de datos

Descripción del propósito: Mostrar los datos tras una operación o resultado de una interfaz.

Origen de entrada/destino de salida: Origen de entrada, teclado. Destino de salida, la pantalla del PC.

Precisión: No aplicable

Unidades de medida: No aplicable.

Temporización: Debe mostrar los datos rápidamente, menos de 1 segundo. Al ser texto plano no habrá problema.

Relaciones con otras entradas/salidas: Cualquier otra interfaz tendrá que mostrar un mensaje, así que sus salidas se relacionarán con esta.

Organización/formatos de pantalla:

```
"-----"
"|           %fuente de datos           |"
"-----"
"|  #   | Type | Field1 | Field2 | ... | FieldN |"
"-----"
"| ..... | ..... | ..... | ..... | ..... | ..... |"
"-----"
```

Organización/formatos de ventana:

No habrá GUI.

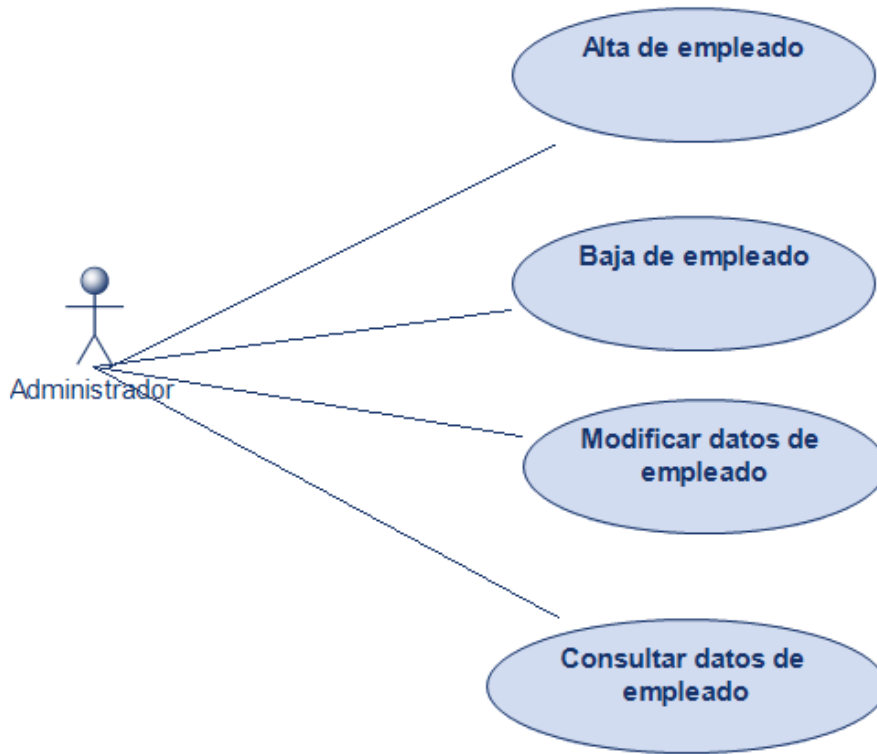
Formato de fechas: "dd/mm/aaaa"

Formato de comandos: No hay

Mensajes de fin: No hay

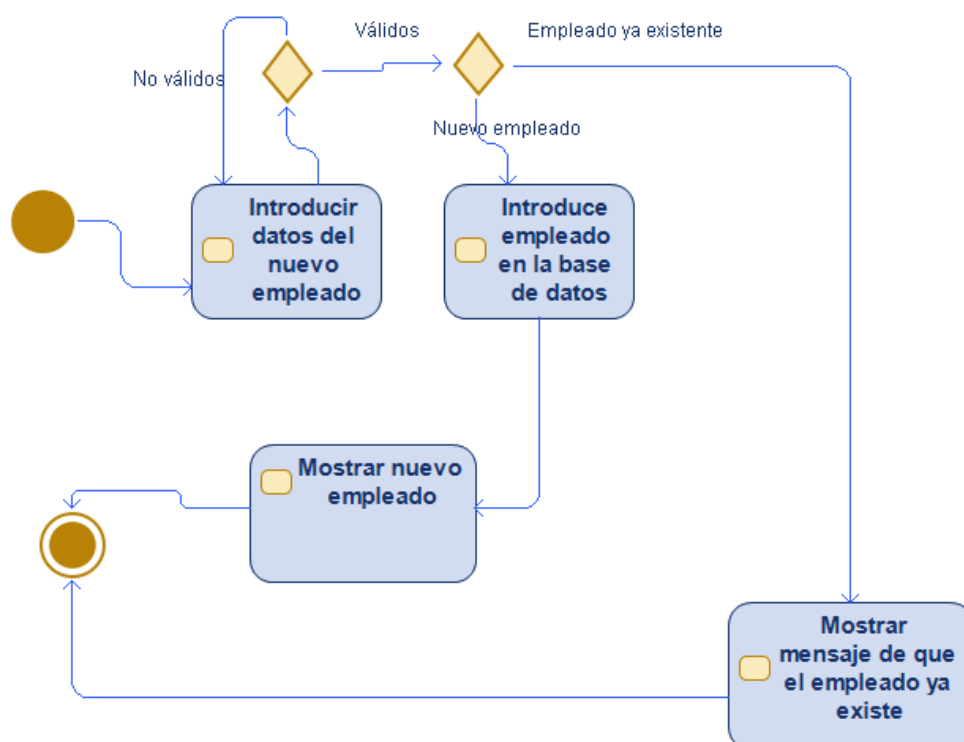
3.2 Requisitos funcionales

3.2.1 EMPLEADOS

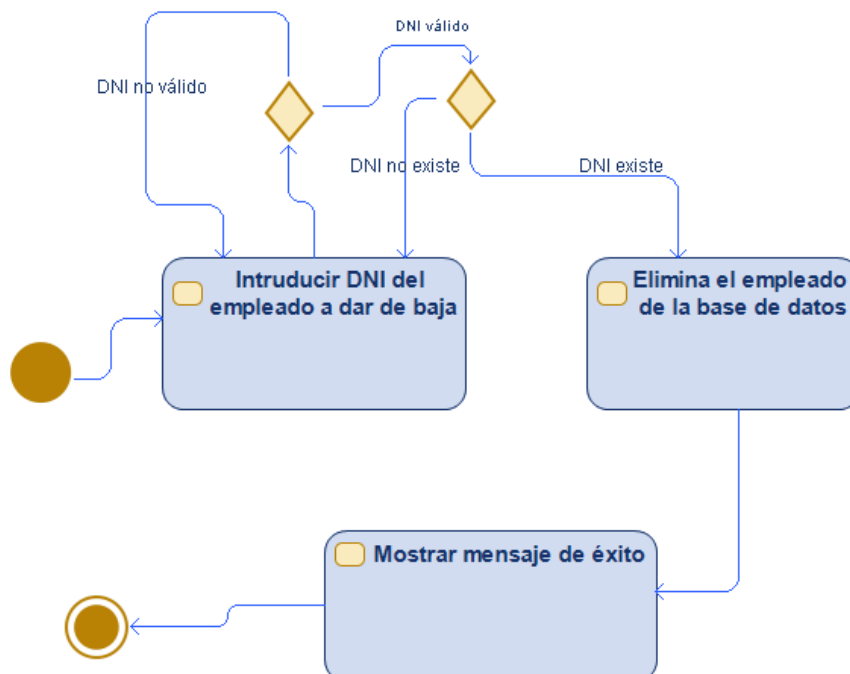


CU-EMP-1	Alta empleado	
Subsistema	Gestión de empleados	
Descripción	Dar de alta a un nuevo empleado, introduciendo en nuestra base de datos su información	
Actores	Actor principal que invoca el Caso de Uso: Administrador Actores secundarios: Base de Datos	
Descrito por	Cristina	
Prioridad	Alta	
Estabilidad	media	
Precondiciones	no hay identificadores ni DNIs repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Introducir nombre, DNI, sueldo, departamento
	2	Comprobar datos válidos.
	3	Comprobar que no exista ya el empleado en la base de datos
	4	Introducir empleado en la base de datos
	5	Mostrar empleado al administrador
Postcondiciones	no hay identificadores ni DNIs repetidos en la base de datos	
Éxito:	Nuevo empleado en la base de datos	
Fallo:	Enviar mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción
	2.a	Vuelve al paso 1.
	3.a	Si los datos no son válidos, volver a pedir datos correctos

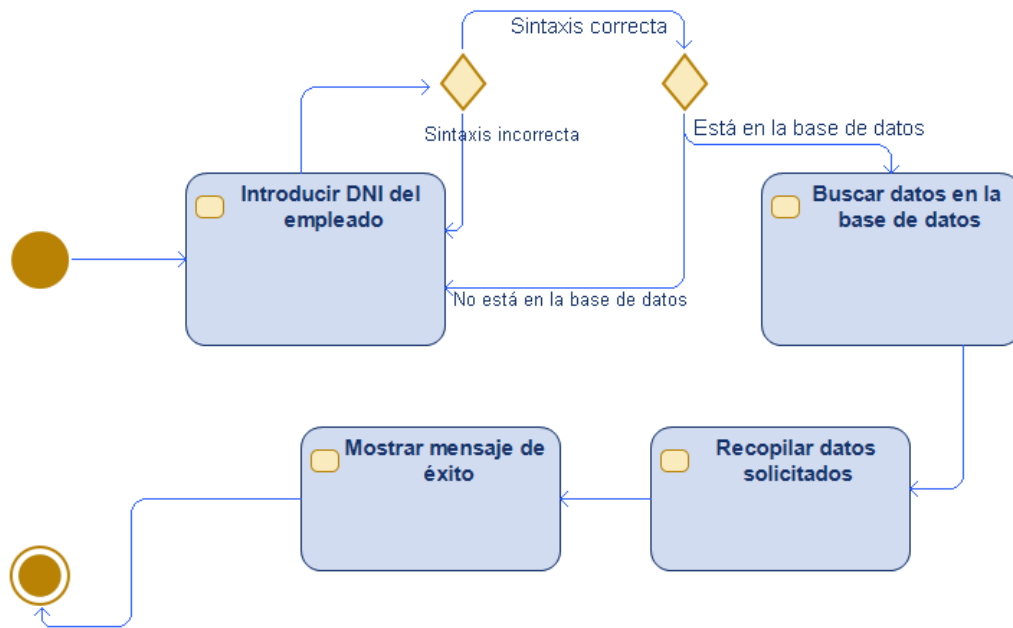
CU-EMP-1	Alta empleado	
Subsistema	Gestión de empleados	
Descripción	Dar de alta a un nuevo empleado, introduciendo en nuestra base de datos su información	
Actores	Actor principal que invoca el Caso de Uso: Administrador Actores secundarios: Base de Datos	
Descrito por	Cristina	
Prioridad	Alta	
Estabilidad	media	
Precondiciones	no hay identificadores ni DNIs repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Introducir nombre, DNI, sueldo, departamento
	2	Comprobar datos válidos.
	3	Comprobar que no exista ya el empleado en la base de datos
	4	Introducir empleado en la base de datos
	5	Mostrar empleado al administrador
Postcondiciones	no hay identificadores ni DNIs repetidos en la base de datos	
Éxito:	Nuevo empleado en la base de datos	
Fallo:	Enviar mensaje explicando la causa del fallo	
	4.a	Si el empleado ya existía, no introducir a la base de datos y comunicar al administrador
Requisitos funcionales	no	Consentimiento del empleado para almacenar sus datos
Comentarios		



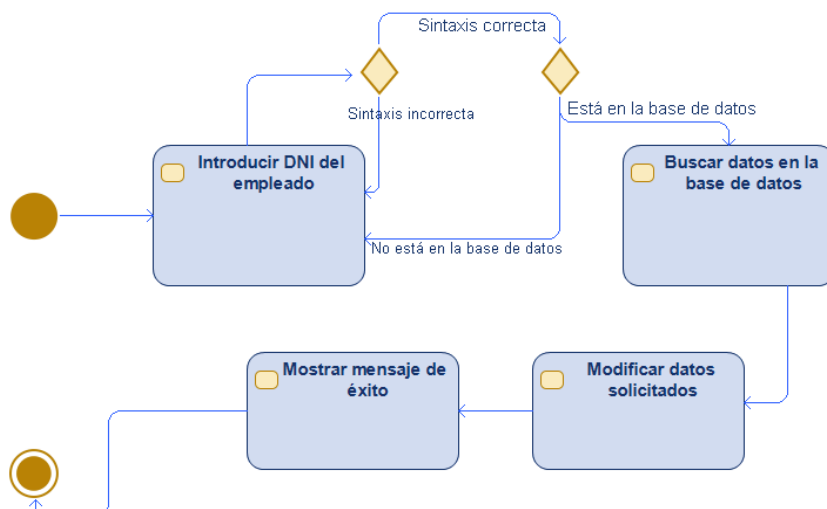
CU-EMP-2	Baja empleado	
Subsistema	Gestión de empleados	
Descripción	Dar de baja a un empleado, eliminando sus datos de nuestra base de datos	
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por	Cristina	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	DNI pertenece a la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Introducir DNI del empleado a dar de baja
	2	Comprobar sintaxis del DNI
	3	Comprobar que dicho DNI forma parte de la base de datos
	4	Eliminar de la base de datos toda la información relacionada con ese DNI
	5	Mostrar mensaje de que los datos se han borrado con éxito
Postcondiciones	DNI pertenece a la base de datos	
Éxito:	Datos asociados al DNI borrados	
Fallo:	Mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción
	3.a	Solicitar otro DNI con sintaxis correcta
	4.a	Mensaje indicando que dicho DNI no se encuentra en la base de datos
	5.a	Hay datos que no han podido ser borrados
Requisitos no funcionales	No modificar información inherente al trabajo de la empresa con la eliminación de los datos del empleado	
Comentarios	Aclaraciones al Caso de Uso	



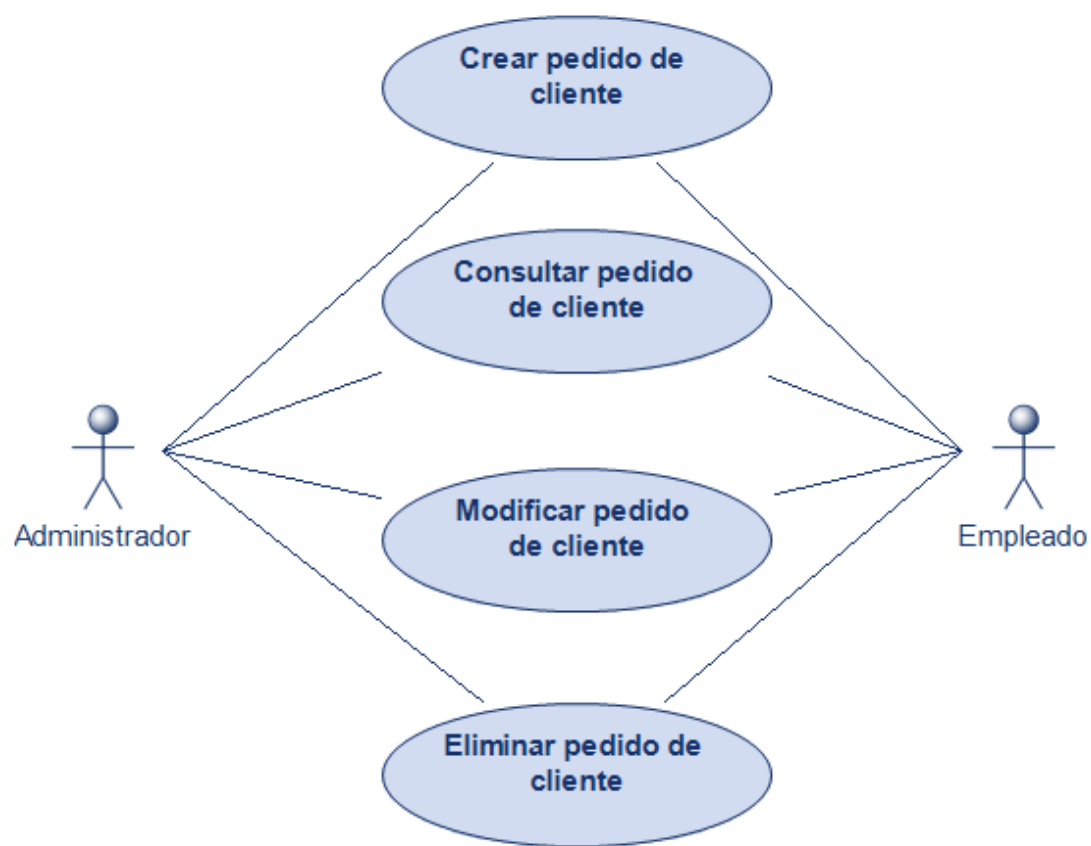
CU-EMP-3		Consulta datos empleado
Subsistema	Gestión de empleados	
Descripción	Consultar los datos de un empleado, los cuales están guardados en nuestra base de datos	
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por	Cristina	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	DNI pertenece a la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Introducir DNI del empleado cuyos datos se quieren consultar
	2	Comprobar sintaxis del DNI
	3	Comprobar que dicho DNI forma parte de la base de datos
	4	Buscar todos los datos asociados a ese DNI
	5	Recopilar los datos solicitados
	6	Mostrar mensaje de que la consulta ha sido exitosa
Postcondiciones	DNI pertenece a la base de datos	
Éxito:	Datos asociados al DNI encontrados	
Fallo:	Mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción
	3.a	Solicitar otro DNI con sintaxis correcta
	4.a	Mensaje indicando que dicho DNI no se encuentra en la base de datos
	5.a	No se han encontrado datos asociados a ese DNI
Requisitos no funcionales	No modificar ningún dato o información de la base de datos	
Comentarios	Aclaraciones al Caso de Uso	



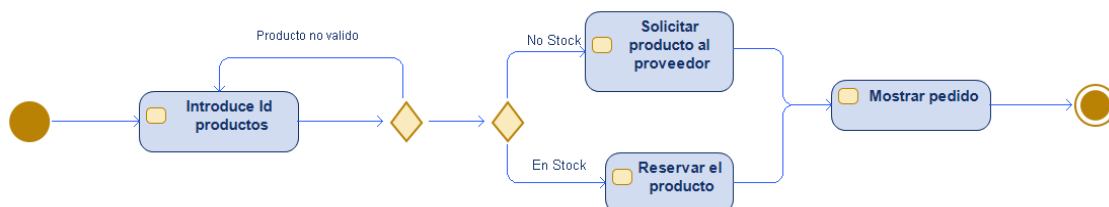
CU-EMP-4		Modificación datos empleado
Subsistema	Gestión de empleados	
Descripción	Modificar los datos de un empleado	
Actores	Actor principal que invoca el Caso de Uso: Administrador Actores secundarios: Base de Datos	
Descrito por	Cristina	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	DNI pertenece a la base de datos.	
Flujo de eventos principal	Paso	Acción
	1	El sistema pide un identificador del empleado del que se quiere modificar los datos.
	2	El sistema le muestra una ventana con los datos del Usuario consultado.
	3	El sistema habilita la modificación de los datos del Usuario consultado en el paso anterior.
	4	El Sistema realiza comprobaciones de los datos modificados.
	5	Los datos se guardan en la base de datos del Sistema
	6	se muestra un mensaje donde se notifica del éxito en la modificación de los datos del Usuario.
Postcondiciones Éxito:	Se realiza una modificación en la información del usuario.	
Fallo:	El Sistema y la base de datos quedan consistentes y en el mismo estado que al inicio.	
Flujo de eventos alternativo	Paso	Acción
	2.a	El sistema muestra un mensaje de error y pide de nuevo el identificador del usuario a modificar.
	5.a	Se muestra un mensaje indicando que los datos introducidos son erróneos.
Requisitos no funcionales		
Comentarios	Aclaraciones al Caso de Uso	



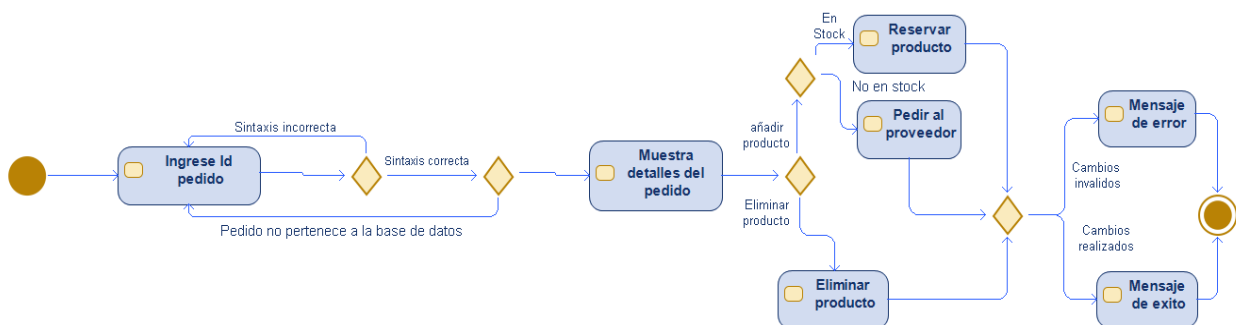
3.2.2 PEDIDOS DE CLIENTES



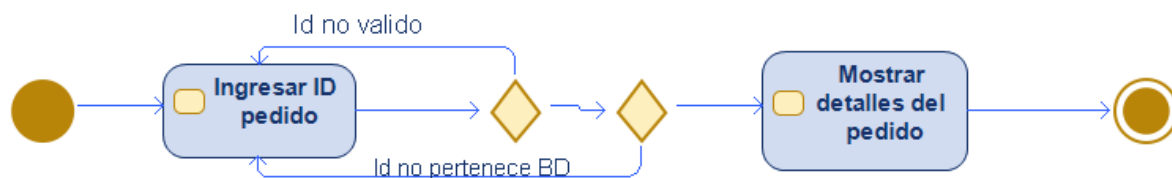
CU-PED-1		Crear pedido del cliente	
Subsistema	Pedidos de clientes		
Descripción	Analiza los datos de los productos que requiere el cliente y dependiendo de si dichos productos forman parte de nuestro stock, o no se genera una reserva de dichos productos o se encargan a los proveedores.		
Actores	Actor principal que invoca el Caso de Uso: Empleado, administrador Actores secundarios: Base de Datos		
Descrito por	Marcos		
Prioridad	Alta		
Estabilidad	media		
Precondiciones	Cliente perteneciente a la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir código de los productos	
	2	Comprobar si proveemos de dichos productos	
	3	Comprobar si hay stock de los productos pedidos	
	4	Generar reserva de dichos productos	
	5	Mostrar el pedido	
Postcondiciones	Cliente pertenece a la base de datos		
Éxito:			
Fallo:			
Flujo de eventos alternativo	Paso	Acción	
	3.a	Producto no perteneciente a la base de datos	
	4.a	Pedir producto a los proveedores	
Requisitos funcionales	no		
Comentarios			



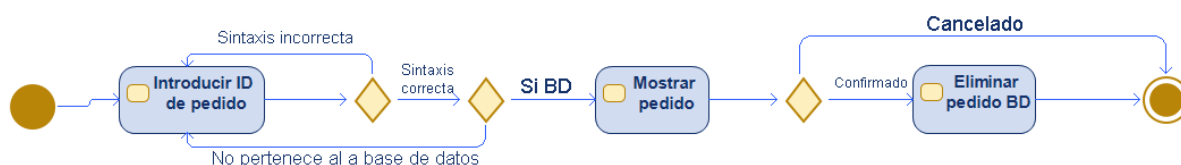
CU-PED-2 Modificar pedido del cliente		
Subsistema	Pedidos de clientes	
Descripción	Modificar el pedido cuando el cliente desee añadir o eliminar un artículo de su pedido	
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos	
Descrito por	Marcos	
Prioridad	Alta	
Estabilidad	media	
Precondiciones	Cliente y Pedido en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Se ingresa el identificador del pedido que se desea modificar
	2	Se comprueba que el identificador sea válido
	3	Se comprueba que el pedido esté en la base de datos
	4	Se muestra los detalles del pedido actual
	5	Desea añadir o eliminar
	6	si añadir → comprueba la disponibilidad en stock del producto
	7	Reservar el producto
	8	El sistema valida las modificaciones realizadas
	9	Se muestra un mensaje de éxito
Postcondiciones	Cliente y pedido en la base de datos	
Éxito:	Pedido modificado	
Fallo:	pedido mismo estado	
Flujo de eventos alternativo	Paso	Acción
	3.a	Volver a pedir id
	4.a	Volver a pedir id
	6.a	Elimina el producto del pedido
	7.a	Pedir al proveedor
	9.a	Mensaje de que el pedido no ha sido cambiado
Requisitos funcionales	no	
Comentarios		



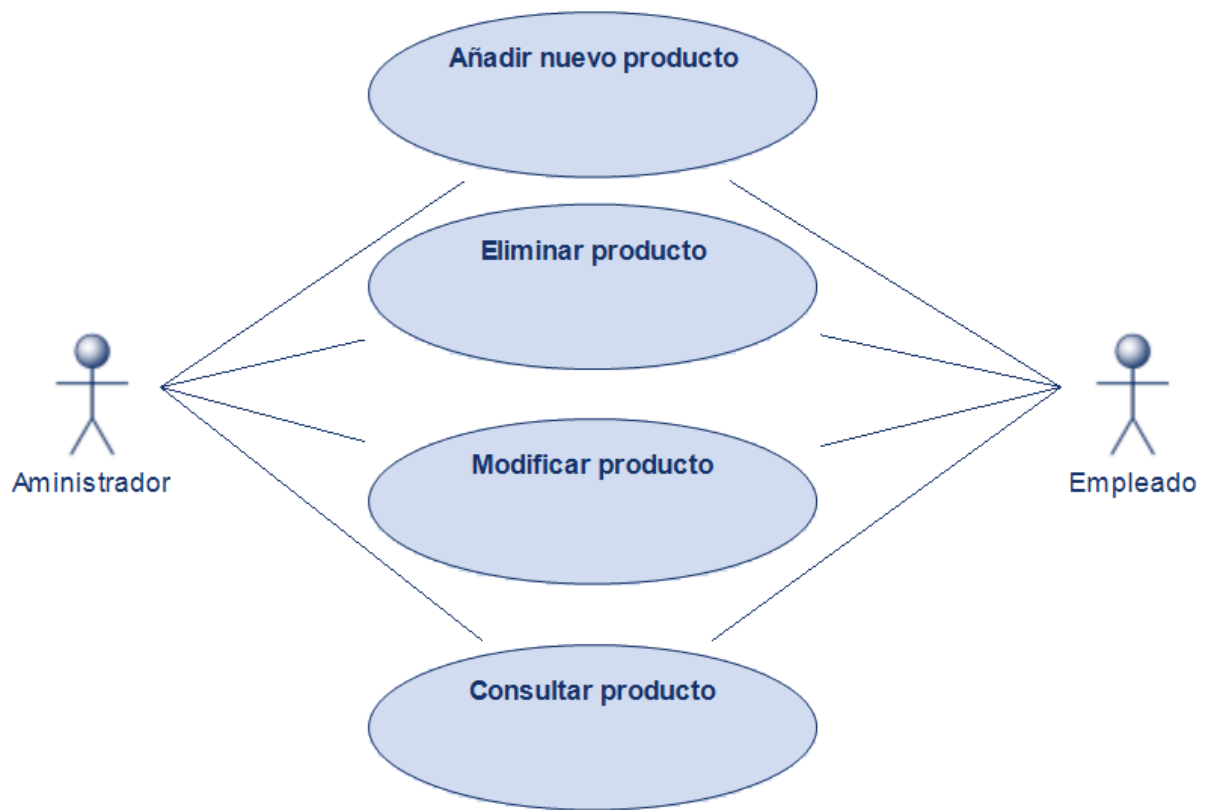
CU-PED-3 Consulta pedido del cliente		
Subsistema	Pedidos de clientes	
Descripción	Consultar la información de un pedido previamente creado	
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos	
Descrito por	Marcos	
Prioridad	Baja	
Estabilidad	Alta	
Precondiciones	Cliente y Pedido en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Se ingresa el identificador del pedido que se desea modificar
	2	Se comprueba que el identificador sea válido
	3	Se comprueba que el pedido esté en la base de datos
	4	Se muestra los detalles del pedido actual
Postcondiciones	Cliente y pedido en la base de datos	
Éxito:	Pedido modificado	
Fallo:	pedido mismo estado	
Flujo de eventos alternativo	Paso	Acción
	3.a	Ingresar otro id pedido
	4.a	Ingresar otro id pedido
	6.a	Elimina el producto del pedido
	7.a	Pedir al proveedor
	8.a	Mensaje de que el pedido no ha sido cambiado
Requisitos funcionales	no	
Comentarios		



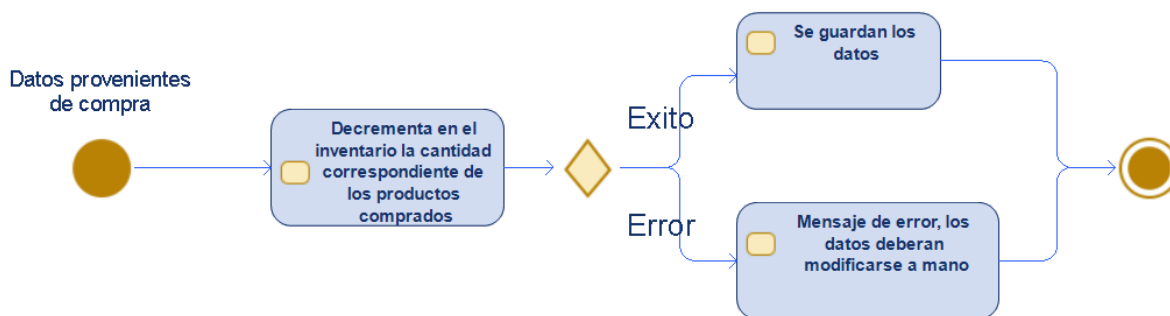
CU-PED-4	Baja pedido del cliente	
Subsistema	Pedidos de clientes	
Descripción	Eliminar un pedido ya sea porque se ha cancelado o porque se ha completado y por tanto convertido en venta	
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos	
Descrito por	Marcos	
Prioridad	Alta	
Estabilidad	media	
Precondiciones	Cliente y Pedido en la base de datos	
Flujo de eventos principal	Paso	
	1	Se ingresa el identificador del pedido que se desea eliminar
	2	Comprobar sintaxis correcta
	3	Comprobar existe en la base de datos
	4	Mostrar el pedido
	5	Mensaje de confirmación para eliminar el pedido
	6	Pedido eliminado
Postcondiciones	Cliente y pedido en la base de datos	
Éxito:	Pedido eliminado	
Fallo:	pedido no existente o no eliminado	
Flujo de eventos alternativo	Paso	
	3.a	Volver a pedir id de pedido
	4.a	Volver a pedir id de pedido
	6.a	Cancelar el proceso
Requisitos funcionales	no	
Comentarios		



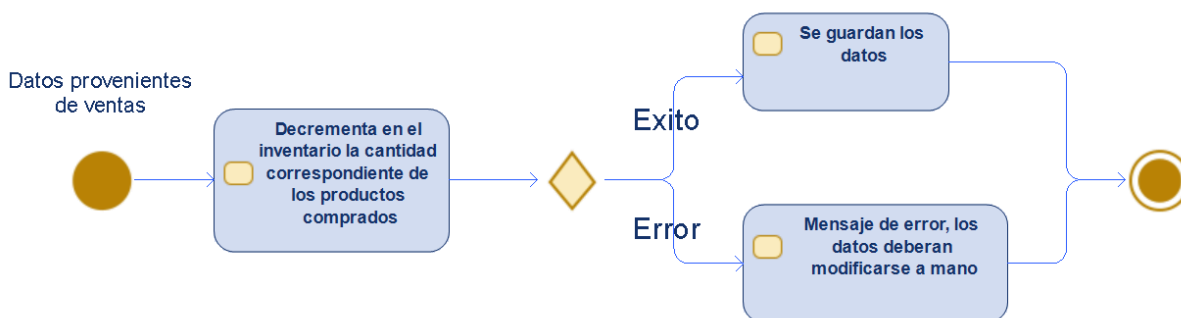
3.2.3 GESTIÓN DE PRODUCTOS



CU-GP-1		Añadir nuevo producto (Alta)
Subsistema	Gestión de productos	
Descripción	Añade a la base de datos un nuevo producto tras su compra	
Actores	Actor principal que invoca el Caso de Uso: <i>Empleado o Administrador</i> Actores secundarios: <i>Base de datos</i>	
Descrito por	Ricardo	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	Se debe de haber realizado una compra con anterioridad	
Flujo de eventos principal	Paso	
	1	Recibe los datos de la compra del modulo de compras
	2	Aumenta en el inventario la cantidad correspondiente de los productos comprados
	3	
	4	
	5	
Postcondiciones		
Éxito:		Cantidad aumentada
Fallo:		Mensaje de error si la modificación no se hubiese ejecutado correctamente, avisando de que la modificación debe ser hecha a mano con el submódulo de modificación
Flujo de eventos alternativo	Paso	
	2.1	Si la cantidad no puede ser modificada por algún motivo, aparece el mensaje de error
Requisitos no funcionales	Debe de requerirse la compra de nuevos productos	
Comentarios		

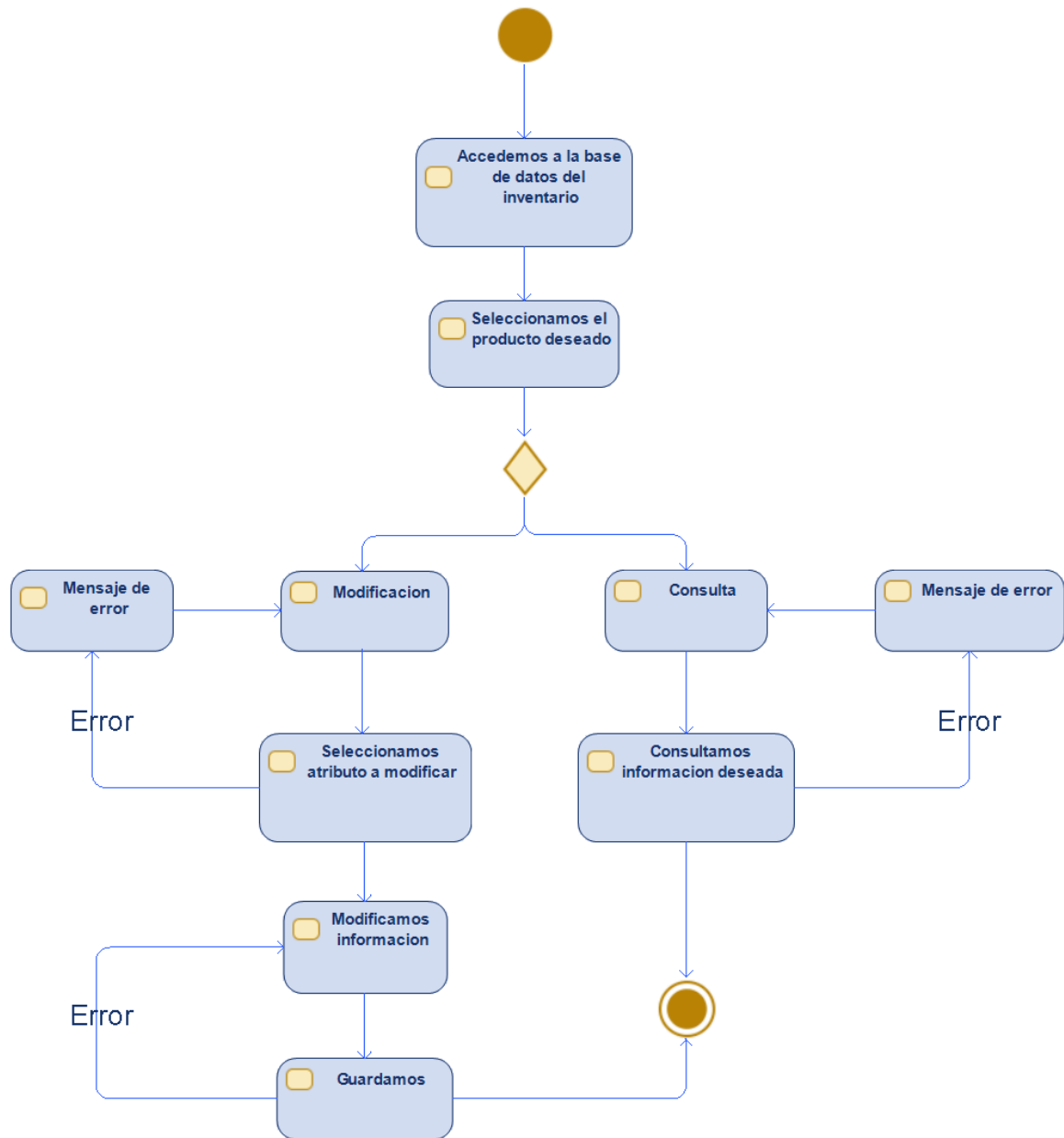


CU-GP-2		Eliminar producto (Baja)
Subsistema	Gestión de productos	
Descripción	Elimina de la base de datos un nuevo producto tras su venta	
Actores	Actor principal que invoca el Caso de Uso: <i>Empleado o Administrador</i> Actores secundarios: <i>Base de datos</i>	
Descrito por	Ricardo	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	Se debe de haber realizado una venta	
Flujo de eventos principal	Paso	
	1	Recibe los datos de la venta del módulo de ventas
	2	Decrementa en el inventario la cantidad correspondiente de los productos vendidos
	3	
	4	
	5	
Postcondiciones		
Éxito:		Cantidad reducida
Fallo:		Mensaje de error si la modificación no se hubiese ejecutado correctamente, avisando de que la modificación debe ser hecha a mano con el submódulo de modificación
Flujo de eventos alternativo	Paso	
	2.1	Si la cantidad no puede ser modificada por algún motivo, aparece el mensaje de error
Requisitos no funcionales	Debe de requerirse la venta de algún producto	
Comentarios		

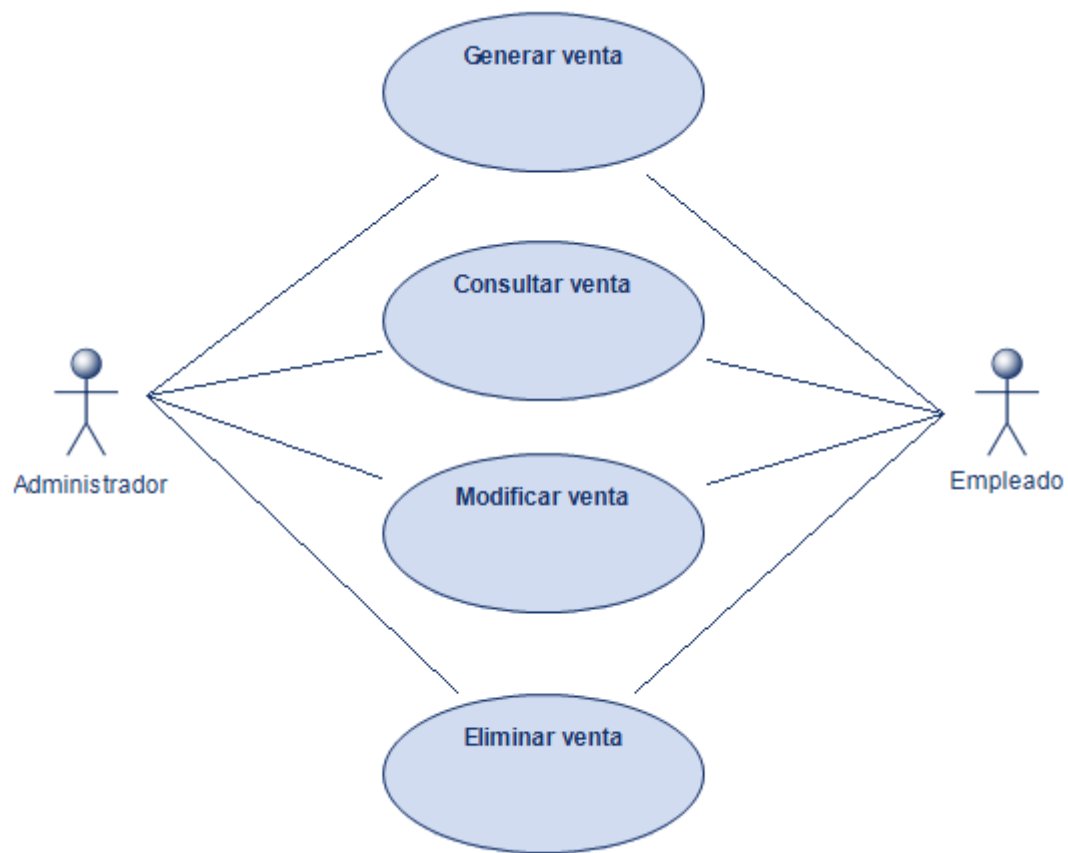


CU-GP-3		Modificar producto (Modificación)
Subsistema	Gestión de productos	
Descripción	Modifica los datos de un producto de la base de datos	
Actores	Actor principal que invoca el Caso de Uso: <i>Empleado o Administrador</i> Actores secundarios: <i>Base de datos</i>	
Descrito por	Ricardo	
Prioridad	Media	
Estabilidad	Media	
Precondiciones	El producto que queremos modificar debe de existir en la base de datos	
Flujo de eventos principal	Paso	
	1	Accedemos a la base de datos y seleccionamos el producto deseado
	2	Seleccionamos qué atributo deseamos modificar (por ejemplo la cantidad)
	3	Modificamos el dato
	4	Guardamos
	5	
Postcondiciones		
Éxito:		Datos deseados modificados
Fallo:		Mensaje de error si la modificación no se hubiese ejecutado correctamente
Flujo de eventos alternativo	Paso	
	2.1	Si el dato no puede ser modificado nos avisa un mensaje
	4.1	Si ha habido un error en la modificación, aparece un mensaje y se regresa al paso 2
Requisitos no funcionales	Se debe de querer modificar algún dato de algún producto	
Comentarios		

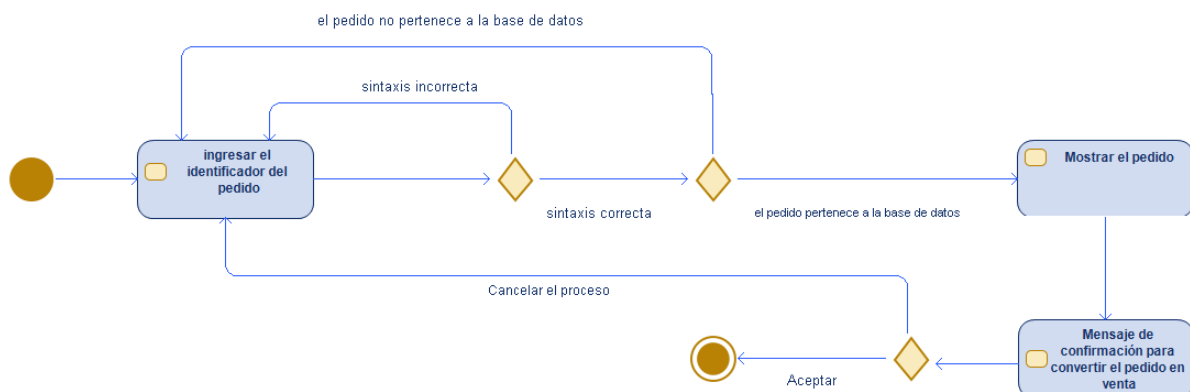
CU-GP-4	Consultar producto (Consulta)	
Subsistema	Gestión de productos	
Descripción	Permite consultar los datos de un producto de la base de datos	
Actores	Actor principal que invoca el Caso de Uso: <i>Empleado o Administrador</i> Actores secundarios: <i>Base de datos</i>	
Descrito por	Ricardo	
Prioridad	Baja	
Estabilidad	Alta	
Precondiciones	El producto que queremos consultar debe de existir en la base de datos	
Flujo de eventos principal	Paso	
	1	Accedemos a la base de datos y seleccionamos el producto deseado
	2	Seleccionamos la opción de ver información
	3	Consultamos el dato deseado
	4	
	5	
Postcondiciones		
Éxito:		Consultamos la información deseada
Fallo:		Mensaje de error si no se pudiese acceder a los datos
Flujo de eventos alternativo	Paso	
	2.1	Si no se puede ver la información, salta un mensaje de error y se vuelve al paso 1
Requisitos no funcionales	Se debe de querer consultar algún dato de algún producto	
Comentarios		



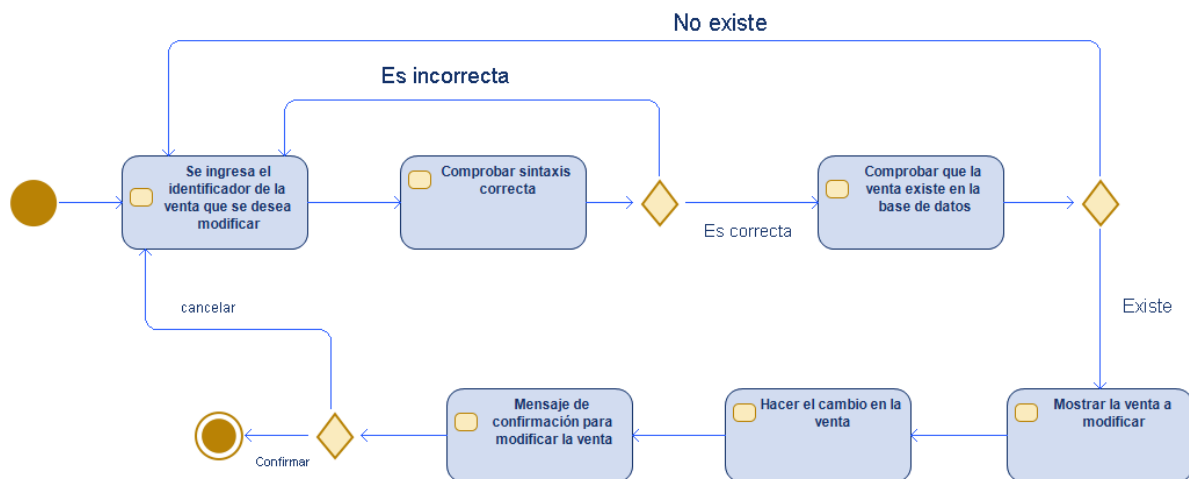
3.2.4 VENTAS



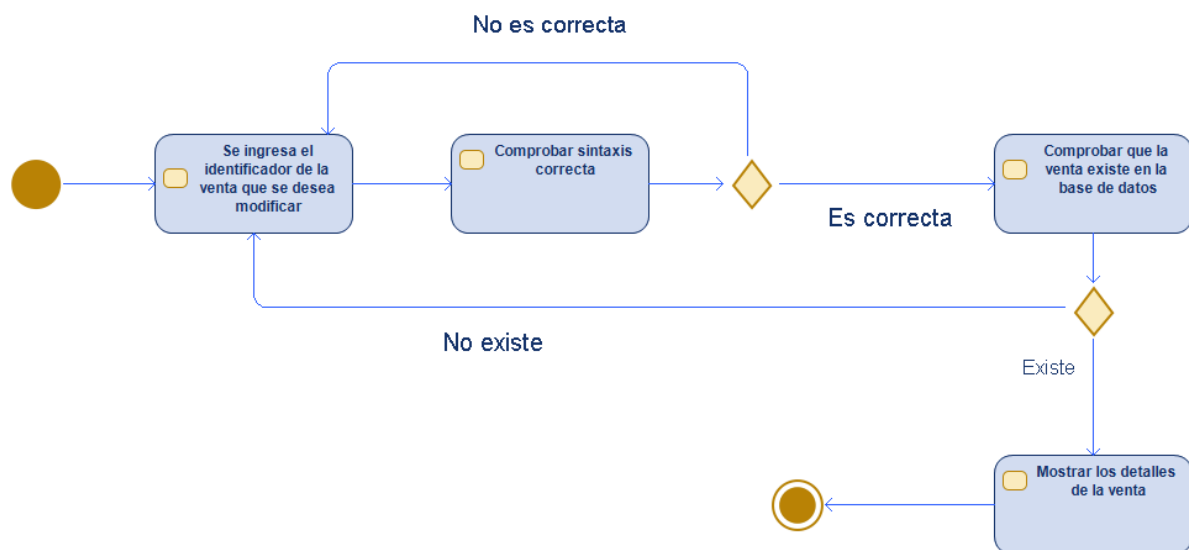
CU-VEN-1		Generar una venta
Subsistema	Ventas	
Descripción	Convierte un pedido de la base de datos en una venta	
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos	
Descrito por	Jaime	
Prioridad	Alta	
Estabilidad	media	
Precondiciones	Cliente y Pedido en la base de datos y Pedido entregado	
Flujo de eventos principal	Paso	
	1	Se ingresa el identificador del pedido que se desea convertir en venta
	2	Comprobar sintaxis correcta
	3	Comprobar que el pedido existe en la base de datos
	4	Mostrar el pedido
	5	Mensaje de confirmación para convertir el pedido en venta
	6	Mensaje de éxito de pedido eliminado y venta creada
Postcondiciones	Cliente y pedido en la base de datos y Pedido entregado	
Éxito:	Pedido eliminado y venta creada	
Fallo:	pedido no existente o venta no creada	
Flujo de eventos alternativo	Paso	
	3.a	Volver a pedir id de pedido
	4.a	Volver a pedir id de pedido
	6.a	Cancelar el proceso
Requisitos funcionales	no	
Comentarios		



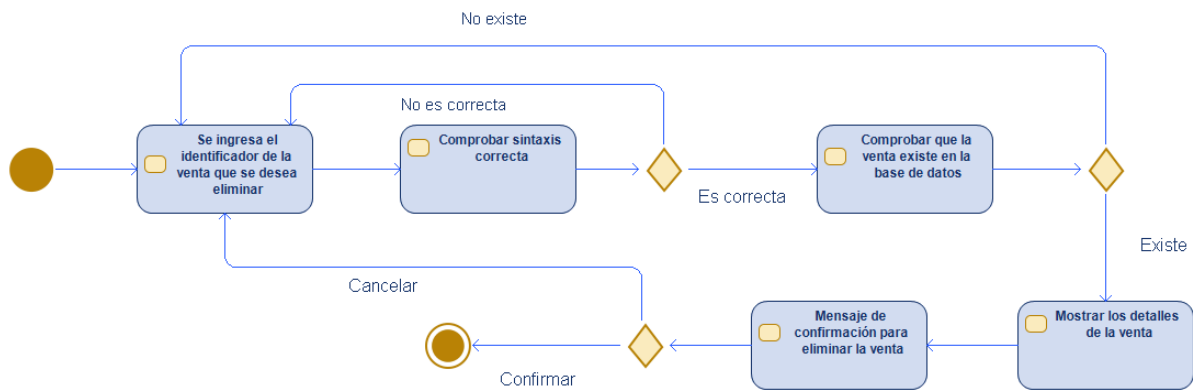
CU-VEN-2		Modifica una venta
Subsistema	Ventas	
Descripción	Modificar una venta si el cliente termina haciendo un cambio	
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos	
Descrito por	Jaime	
Prioridad	Alta	
Estabilidad	baja	
Precondiciones	Venta y Cliente en la base de datos	
Flujo de eventos principal	Paso	
	1	Se ingresa el identificador de la venta que se desea modificar
	2	Comprobar sintaxis correcta
	3	Comprobar que la venta existe en la base de datos
	4	Mostrar la venta a modificar
	5	Hacer el cambio en la venta
	6	Mensaje de confirmación para modificar la venta
	7	Mensaje de éxito de venta modificada
Postcondiciones	Venta y Cliente en la base de datos	
Éxito:	Venta modificada	
Fallo:	Venta no existente o no modificada	
Flujo de eventos alternativo	Paso	
	3.a	Volver a pedir id de pedido
	4.a	Volver a pedir id de pedido
	7.a	Cancelar el proceso
Requisitos funcionales	no	
Comentarios		



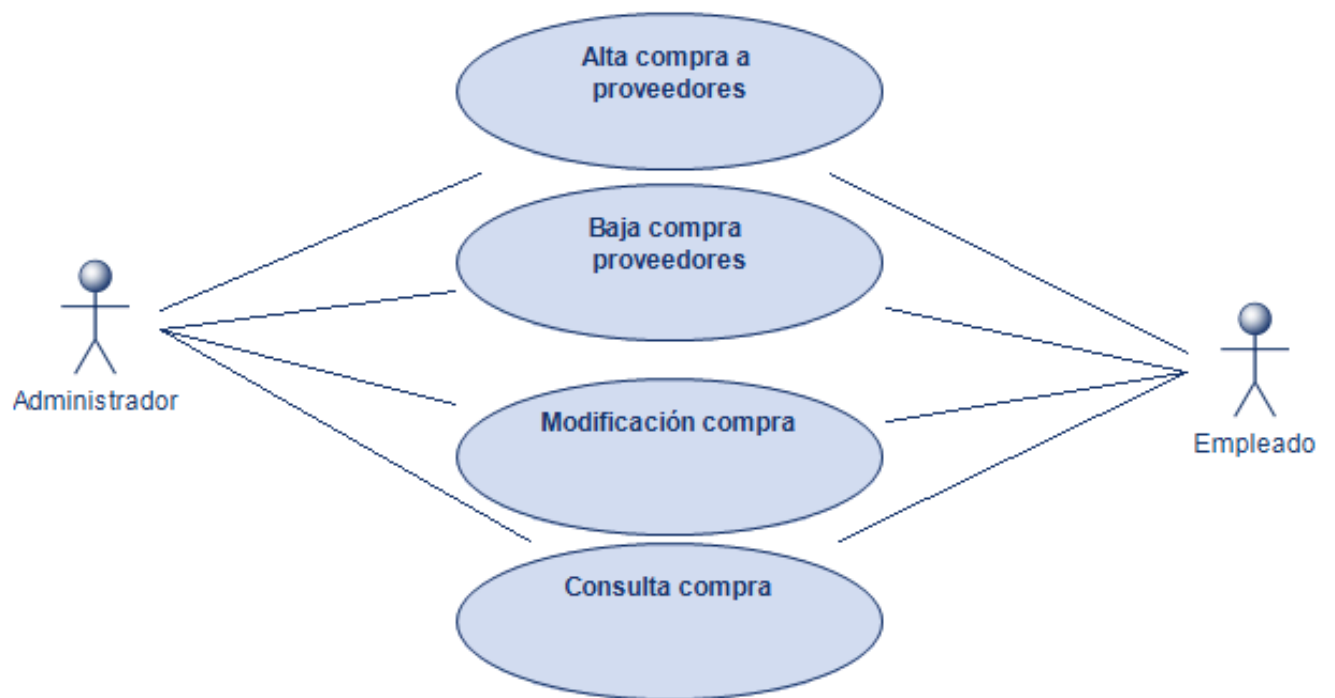
CU-VEN-3		Consultar una venta	
Subsistema	Ventas		
Descripción	Consultar la información de una venta previamente creada		
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos		
Descrito por	Jaime		
Prioridad	Media		
Estabilidad	Alta		
Precondiciones	Venta y Cliente en la base de datos		
Flujo de eventos principal	Paso		
	1	Se ingresa el identificador de la venta que se desea modificar	
	2	Comprobar sintaxis correcta	
	3	Comprobar que la venta existe en la base de datos	
	4	Mostrar los detalles de la venta	
Postcondiciones	Venta y Cliente en la base de datos		
Éxito:	Venta mostrada		
Fallo:	Venta no existente o no mostrada		
Flujo de eventos alternativo	Paso		
	3.a	Volver a pedir id de venta	
	4.a	Volver a pedir id de venta	
Requisitos funcionales	no		
Comentarios			



CU-VEN-4		Eliminar venta	
Subsistema	Ventas		
Descripción	Eliminar una venta ya sea por que proviniera de un error o porque el cliente haya llevado a cabo una devolución del/de los artículo/s		
Actores	Actor principal que invoca el Caso de Uso: Administrador, Empleado Actores secundarios: Base de Datos		
Descrito por	Jaime		
Prioridad	Alta		
Estabilidad	Media		
Precondiciones	Venta y Cliente en la base de datos		
Flujo de eventos principal	Paso		
	1	Se ingresa el identificador de la venta que se desea eliminar	
	2	Comprobar sintaxis correcta	
	3	Comprobar que la venta existe en la base de datos	
	4	Mostrar los detalles de la venta	
	5	Mensaje de confirmación para eliminar la venta	
	6	Mensaje de éxito en la eliminación de la venta	
Postcondiciones	Venta y Cliente en la base de datos		
Éxito:	Venta eliminada		
Fallo:	Venta no existente o no eliminada		
Flujo de eventos alternativo	Paso		
	3.a	Volver a pedir id de venta	
	4.a	Volver a pedir id de venta	
	6.a	Cancelar el proceso	
Requisitos funcionales	no		
Comentarios			



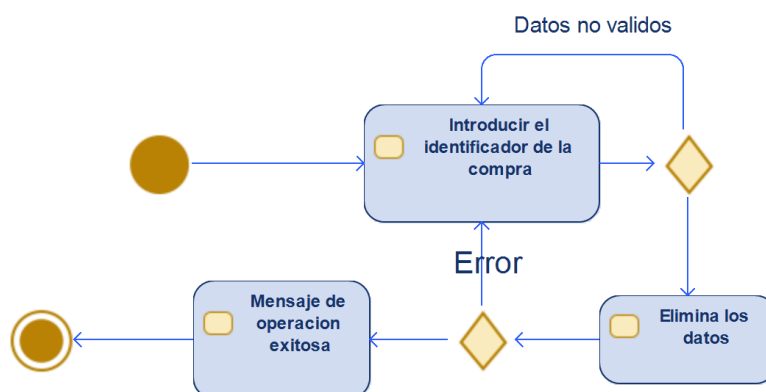
3.2.5 COMPRAS



CU-COM-1 Alta compra productos	
Subsistema	Compra a proveedores
Descripción	Permite realizar una compra de los productos deseados a un proveedor
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador y empleados</i> Actores secundarios: <i>Base de Datos</i>
Descrito por	Ainara
Prioridad	Alta
Estabilidad	Media
Precondiciones	Tener acceso a los proveedores de los que queremos comprar
Flujo de eventos principal	Paso Acción
	1 Seleccionar al proveedor del que queremos realizar la compra
	2 Seleccionar los productos deseados y la cantidad de cada uno
	3 Se registra la compra en la base de datos
	4 Mostrar un mensaje de que la compra se ha realizado correctamente
Postcondiciones	
Éxito:	La compra se realiza y se guarda la información en la base de datos de compras
Fallo:	Mensaje explicando la causa del fallo
Flujo de eventos alternativo	Paso Acción
	2.a Vuelve al paso 1 para pedir que vuelva a introducir los datos
	4.a Notificar al usuario de que se ha producido un error
Requisitos no funcionales	Querer realizar una compra
Comentarios	Aclaraciones al Caso de Uso

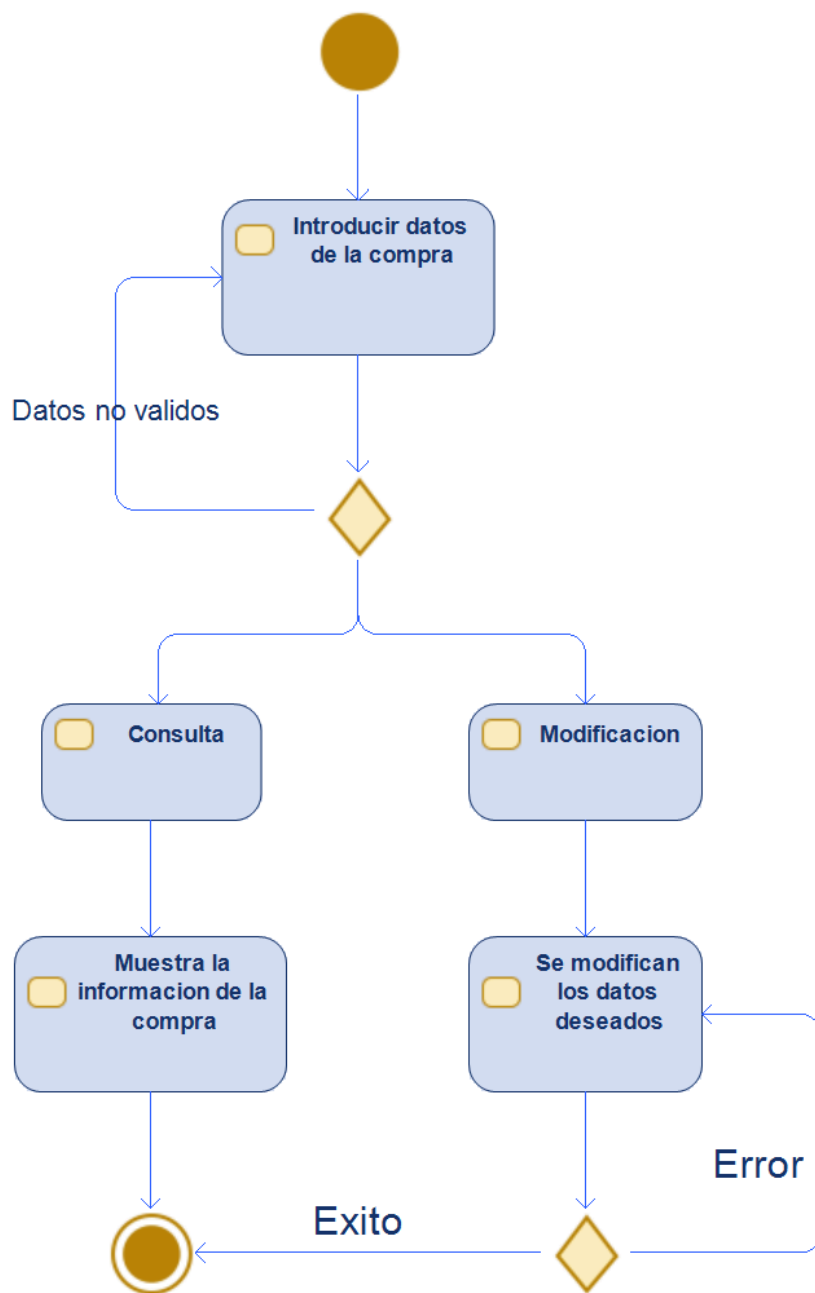


CU-COM-2 Baja compra de productos	
Subsistema	Compra a proveedores
Descripción	Se elimina la información de una compra ya realizada
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador y empleados</i> Actores secundarios: <i>Base de Datos</i>
Descrito por	Ainara
Prioridad	Media
Estabilidad	Media
Precondiciones	Que el identificador de la compra exista en la base de datos
Flujo de eventos principal	Paso Acción
	1 Introducir el identificador de la compra que se desea eliminar
	2 Comprobar sintaxis del identificador
	3 Comprobar que dicha compra forma parte de la base de datos
	4 Elimina los datos
	5 Mostrar mensaje de que la operación se ha realizado correctamente
Postcondiciones Éxito: Fallo:	Se cancela la compra y se borran los datos de la base de datos Mensaje explicando la causa del fallo
Flujo de eventos alternativo	Paso Acción
	2.a Solicitar otro identificador con sintaxis correcta
	3.a Mensaje indicando que dicho identificador no pertenece a ninguna de las compras realizadas y vuelve al paso 1
	4.a Muestra un mensaje de error
Requisitos no funcionales	Querer cancelar una compra
Comentarios	Aclaraciones al Caso de Uso

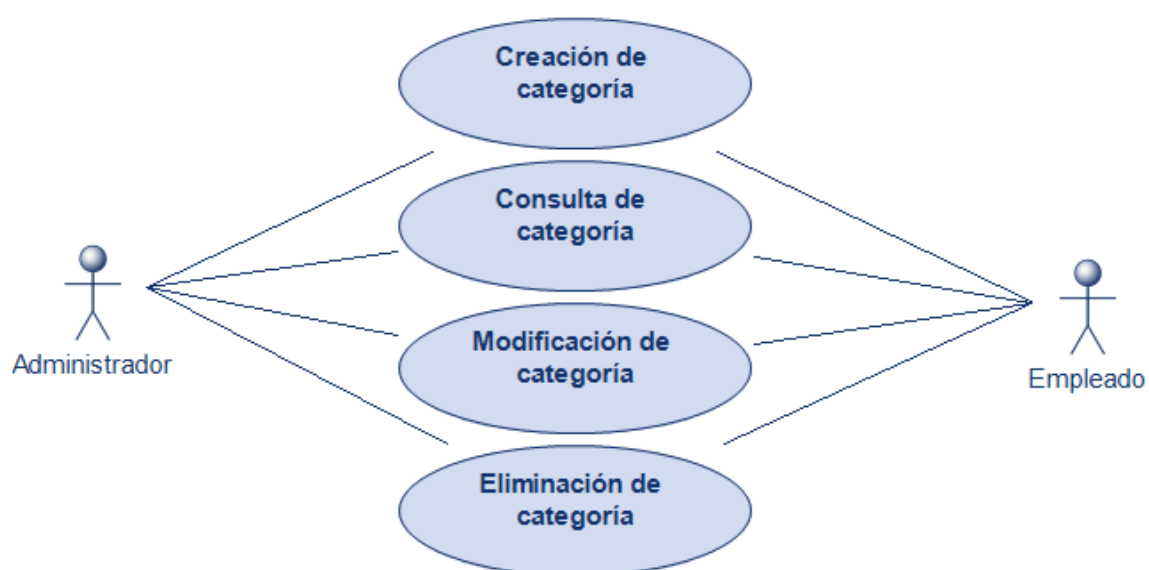


CU-COM-3		Modificación de una compra	
Subsistema	Compra a proveedores		
Descripción	Modificar los datos de una de las compras realizadas		
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador y empleados</i> Actores secundarios: <i>Base de Datos</i>		
Descrito por	Ainara		
Prioridad	Alta		
Estabilidad	Media		
Precondiciones	Identificador de la compra a modificar pertenece a la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir identificador de la compra de la cual se quieren modificar los datos	
	2	Comprobar que dicho identificador pertenece a alguna compra de la base de datos	
	3	El sistema habilita la modificación de los datos	
	4	Se comprueba la validez de los datos introducidos	
	5	Se guardan los cambios en la base de datos	
	6	Se muestra un mensaje del éxito de la modificación de los datos	
Postcondiciones			
Éxito:	Quedan modificados los datos de la compra		
Fallo:	La información de la compra no se modifica y se muestra un mensaje de error		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Vuelve al paso 1	
	4.a	Vuelve al paso 3	
	5.a	Se muestra un mensaje de error	
Requisitos no funcionales	Querer modificar los datos de una compra, tal vez por que los datos de esta hayan sido guardados de forma incorrecta		
Comentarios	Aclaraciones al Caso de Uso		

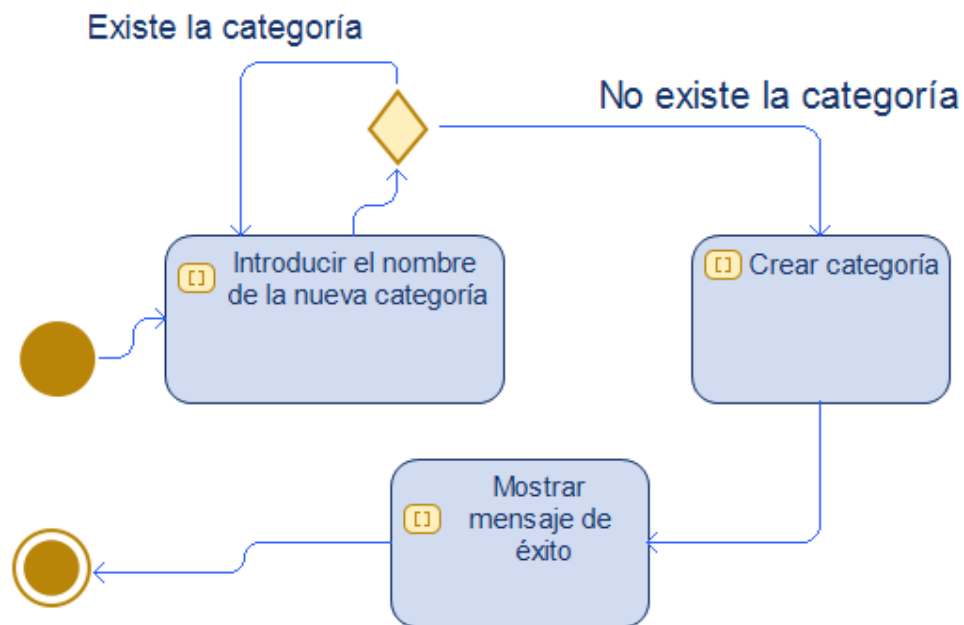
CU-COM-4		Consulta de una compra	
Subsistema	Compra a proveedores		
Descripción	Consultar la información de las compras realizadas a proveedores		
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador y empleados</i> Actores secundarios: <i>Base de Datos</i>		
Descrito por	Ainara		
Prioridad	Baja		
Estabilidad	Alta		
Precondiciones	Identificador de la compra pertenece a la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir el identificador de la compra de la cual se quiere consultar la información	
	2	Comprobar que el identificador pertenece a alguna compra de la base de datos	
	3	Buscar la información de dicha compra	
	4	Mostrar la información recopilada	
Postcondiciones			
Éxito:	Se muestra la información de la compra		
Fallo:	Mensaje explicando la causa del fallo		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Vuelve al paso 1	
	4.a	Muestra un mensaje de error	
Requisitos no funcionales	Querer consultar información sobre una compra ya realizada		
Comentarios	Aclaraciones al Caso de Uso		



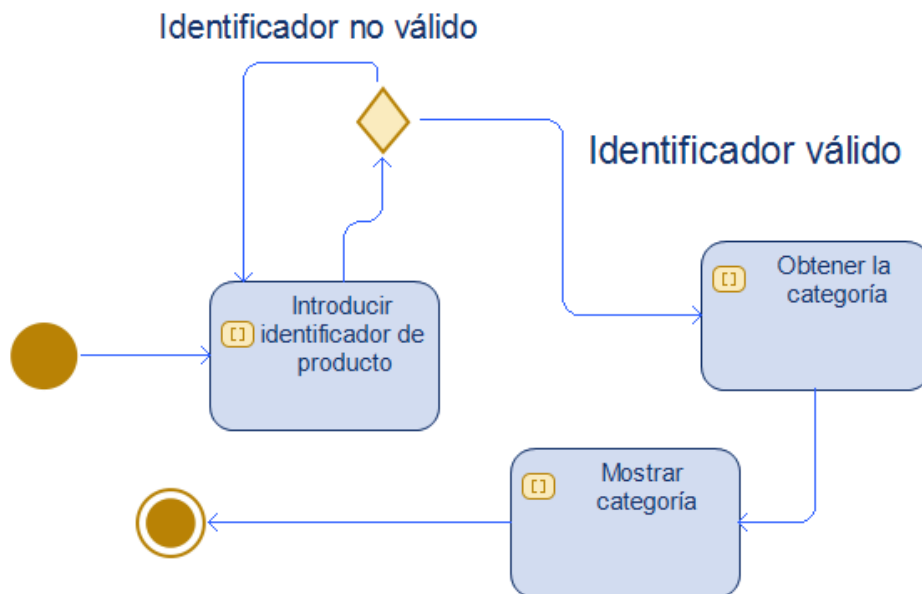
3.2.6 CATEGORÍAS

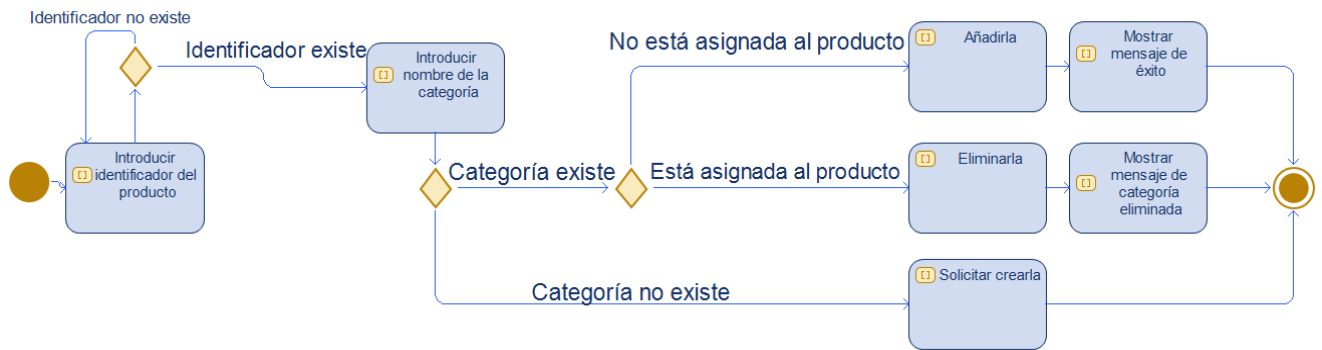


CU-CAT-01		Creación de categoría	
Subsistema	Gestión de Categorías		
Descripción	Crear una nueva categoría para asignar productos		
Actores	Actor principal: Administrador y empleado Actores secundarios: Base de Datos		
Descrito por	Jon		
Prioridad	Alta		
Estabilidad	Media		
Precondiciones	No debe existir una categoría con el mismo nombre en la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir el nombre de la nueva categoría	
	2	Verificar que la categoría no existe en la base de datos	
	3	Añadir la nueva categoría a la base de datos	
	4	Mostrar mensaje de confirmación	
Postcondiciones			
Éxito:	La categoría se crea y queda registrada en la base de datos		
Fallo:	Mensaje indicando que ya existe una categoría con ese nombre		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Si ya existe una categoría con el mismo nombre, solicitar un nuevo nombre	
Requisitos no funcionales	Asegurar consistencia de los datos en la base de datos		
Comentarios			

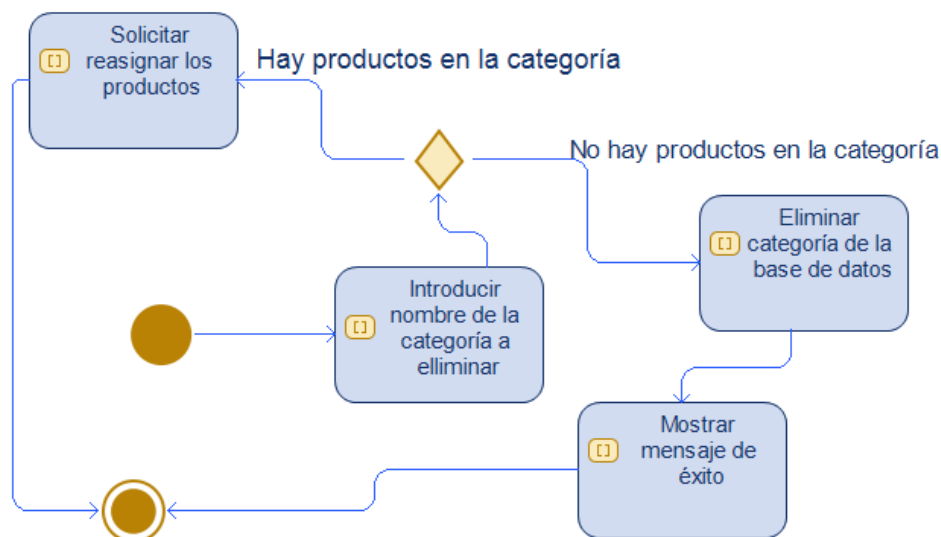


CU-CAT-02		Consulta categoría de un producto	
Subsistema	Gestión de Categorías		
Descripción	Consultar a qué categoría pertenece un producto específico		
Actores	Actor principal: Administrador y empleado Actores secundarios: Base de Datos		
Descrito por	Jon		
Prioridad	Media		
Estabilidad	Alta		
Precondiciones	El producto debe estar registrado en la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir el identificador del producto	
	2	Verificar que el producto existe en la base de datos	
	3	Obtener la categoría asignada al producto	
	4	Mostrar la categoría al usuario	
Postcondiciones			
Éxito:	Se muestra la categoría del producto		
Fallo:	Mensaje indicando que el producto no está registrado		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Solicitar otro identificador si el producto no existe o está mal introducido	
Requisitos no funcionales	Evitar sobrecarga de la base de datos en consultas frecuentes		
Comentarios			



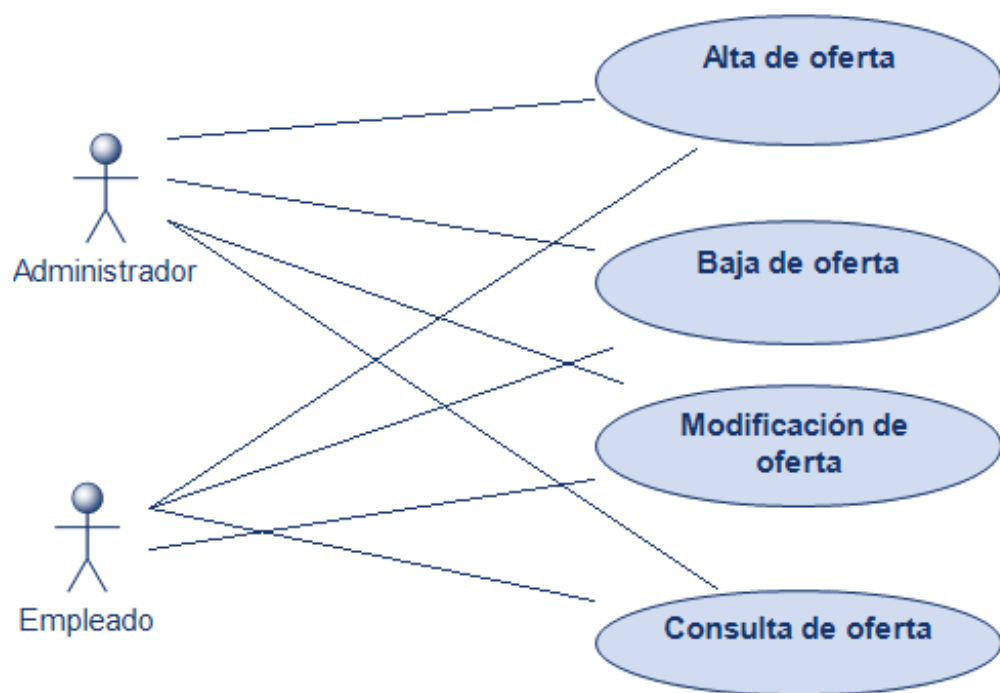


CU-CAT-03 Modificación de categoría		
Subsistema	Gestión de Categorías	
Descripción	Cambiar la categoría asignada a un producto	
Actores	Actor principal: Administrador y empleado Actores secundarios: Base de Datos	
Descrito por	Jon	
Prioridad	Alta	
Estabilidad	Media	
Precondiciones	El producto y las categorías deben estar registradas en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	Introducir el identificador del producto
	2	Verificar que el producto existe en la base de datos
	3	Solicitar el nombre de la categoría
	4	Verificar que la categoría existe
	5	Verificar que la categoría no está asignada al producto
	6	Añadir la categoría al producto
	7	Mostrar mensaje de éxito
Postcondiciones		
Éxito:	Se actualiza la categoría del producto en la base de datos	
Fallo:	Mensaje indicando que el producto o la categoría no existen	
Flujo de eventos alternativo	Paso	Acción
	2.a	Si el producto no existe solicitará otro identificador
	4.a	Si la categoría no existe solicitará crearla
	6.a	Si la categoría está asignada la eliminará del producto y mostrará mensaje de categoría eliminada
Requisitos no funcionales	Registro de cambios para auditoría.	
Comentarios		

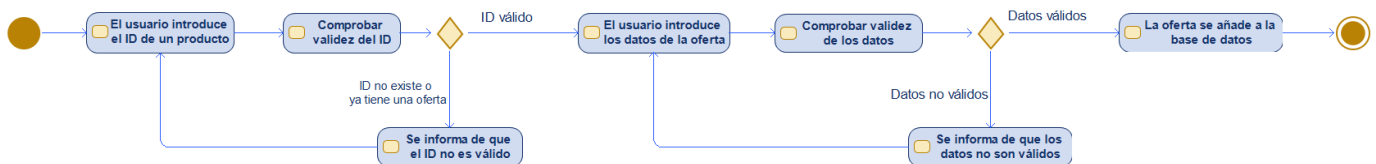


CU-CAT-04		Eliminación de categoría	
Subsistema	Gestión de Categorías		
Descripción	Eliminar una categoría de la base de datos		
Actores	Actor principal: Administrador y empleados Actores secundarios: Base de Datos		
Descrito por	Jon		
Prioridad	Media		
Estabilidad	Media		
Precondiciones	No debe haber productos asignados a la categoría que se desea eliminar		
Flujo de eventos principal	Paso	Acción	
	1	Introducir nombre de la categoría a eliminar	
	2	Verificar que no hay productos asociados	
	3	Eliminar la categoría de la base de datos	
	4	Mostrar mensaje de confirmación	
Postcondiciones			
Éxito:	La categoría se elimina de la base de datos		
Fallo:	Mensaje indicando que hay productos asociados a la categoría		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Si hay productos asociados, solicitar reasignarlos antes de proceder	
Requisitos no funcionales	Garantizar que no se pierdan datos relacionados con productos		
Comentarios			

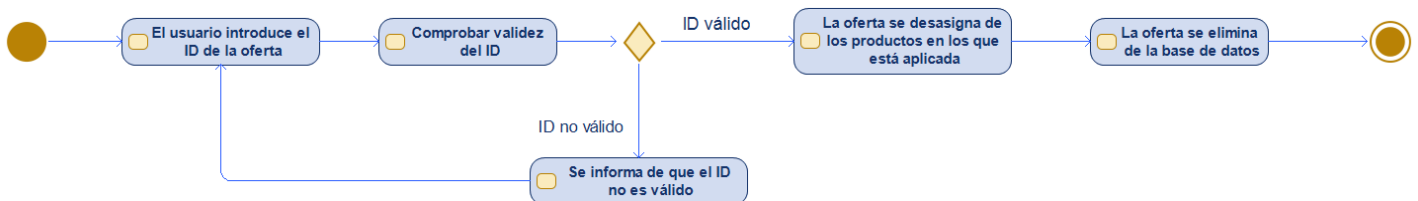
3.2.7 OFERTAS



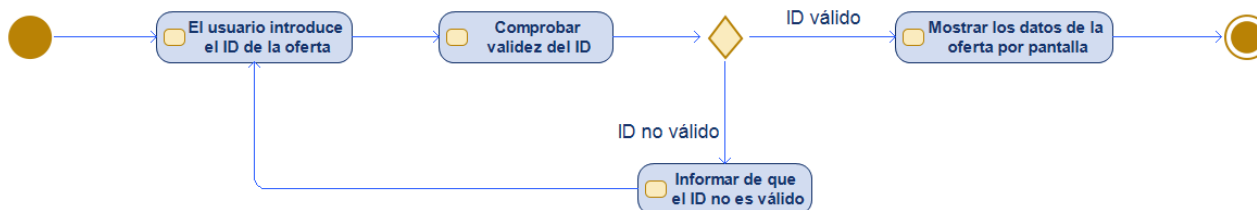
CU-3.5.7-01	Alta oferta	
Subsistema	Gestión de ofertas	
Descripción	Dar de alta una nueva oferta, introduciendo en nuestra base de datos su información y asignándola al producto correspondiente	
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador o empleado</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por	Pablo	
Prioridad	Baja	
Estabilidad	Media	
Precondiciones	No hay identificadores de productos repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	El usuario introduce el ID de un producto al que asignar la nueva oferta
	2	Comprobar que el ID es válido
	3	El usuario introduce los datos de la oferta
	4	Comprobar que los datos de la oferta sean válidos
	5	La oferta se añade a la base de datos
Postcondiciones		
Éxito: Nueva oferta en la base de datos		
Fallo: Enviar mensaje explicando la causa del fallo		
Flujo de eventos alternativo	Paso	Acción
	2.a	Si el ID no es válido, informar de ello y volver a pedir los datos
	4.a	Si los datos de la oferta no son válidos, volver a pedirlos
Requisitos no funcionales		
Comentarios		



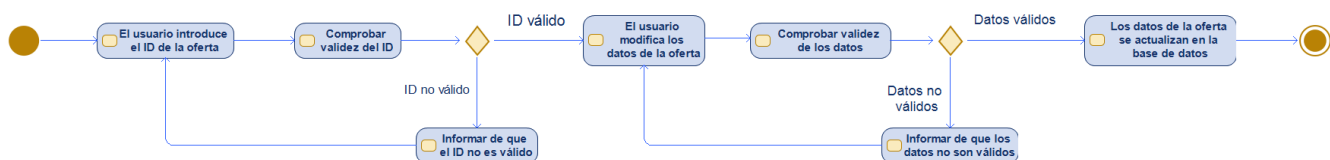
CU-3.5.7-02	Baja oferta	
Subsistema	Gestión de ofertas	
Descripción	Dar de baja una oferta, eliminando de nuestra base de datos su información y desasignándola de los productos correspondientes	
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador o empleado</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por	Pablo	
Prioridad	Baja	
Estabilidad	Media	
Precondiciones	No hay identificadores de ofertas repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	El usuario introduce el ID de la oferta que quiere eliminar
	2	Comprobar que el ID es válido
	3	La oferta se desasigna de los productos en los que está aplicada
	4	La oferta se elimina de la base de datos
Postcondiciones		
Éxito:	Se elimina la oferta de la base de datos	
Fallo:	Enviar mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción
	2.a	Si el ID no es válido, informar de ello y volver a pedir los datos
Requisitos no funcionales		
Comentarios		



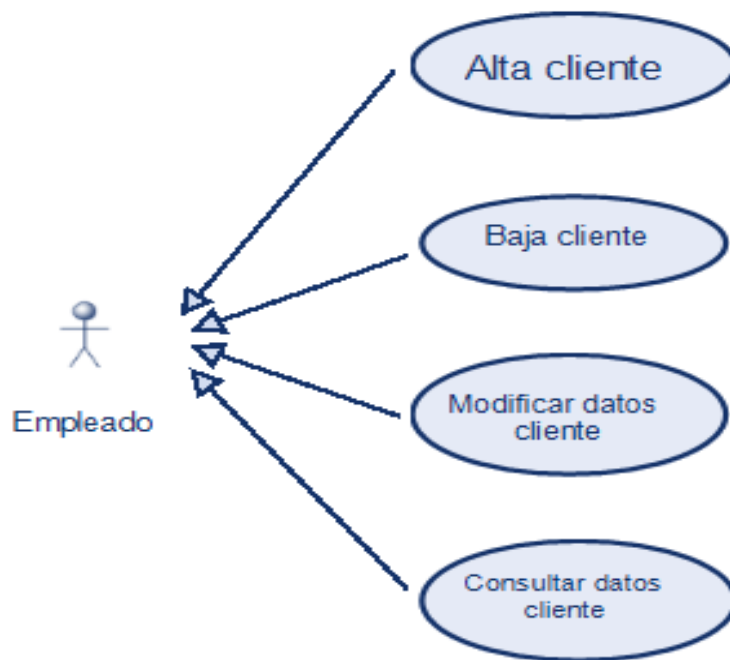
CU-3.5.7-03	Consulta oferta	
Subsistema	Gestión de ofertas	
Descripción	Consultar una de las ofertas existentes en la base de datos	
Actores	Actor principal que invoca el Caso de Uso: <i>Administrador o empleado</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por	Pablo	
Prioridad	Media	
Estabilidad	Alta	
Precondiciones	No hay identificadores de productos repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción
	1	El usuario introduce el ID de la oferta que quiere consultar
	2	Comprobar que el ID es válido
	3	Los datos de la oferta se muestran por pantalla
Postcondiciones		
Éxito:	Se muestra la oferta por pantalla	
Fallo:	Enviar mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción
	2.a	Si el ID no es válido, informar de ello y volver a pedir los datos
Requisitos no funcionales		
Comentarios		



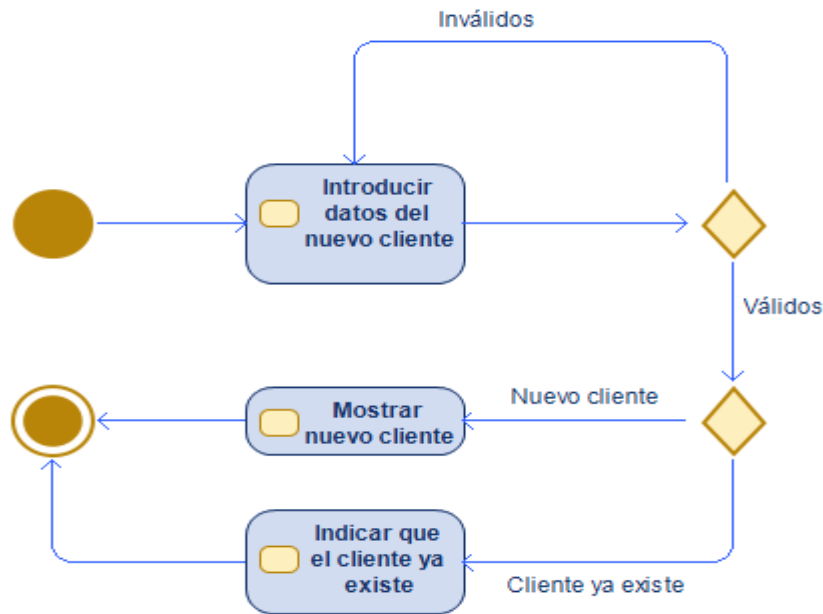
CU-3.5.7-04		Modificación oferta	
Subsistema		Gestión de ofertas	
Descripción		Modificar los datos de una de las ofertas existentes en la base de datos	
Actores		Actor principal que invoca el Caso de Uso: <i>Administrador o empleado</i> Actores secundarios: <i>Base de Datos</i>	
Descrito por		Pablo	
Prioridad		Baja	
Estabilidad		Media	
Precondiciones		No hay identificadores de productos repetidos en la base de datos	
Flujo de eventos principal	Paso	Acción	
	1	El usuario introduce el ID de la oferta que quiere modificar	
	2	Comprobar que el ID es válido	
	3	El usuario modifica los datos de la oferta	
	4	Comprobar que los datos de la oferta sean válidos	
	5	La oferta se actualiza en la base de datos	
Postcondiciones			
Éxito:		Oferta modificada en la base de datos	
Fallo:		Enviar mensaje explicando la causa del fallo	
Flujo de eventos alternativo	Paso	Acción	
	2.a	Si el ID no es válido, informar de ello y volver a pedir los datos	
	4.a	Si los datos de la oferta no son válidos, volver a pedirlos	
Requisitos no funcionales			
Comentarios			



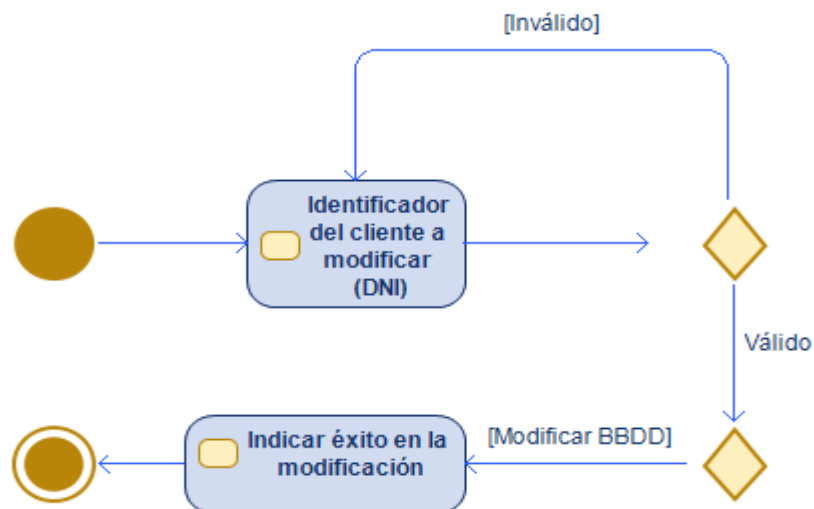
3.2.8 CLIENTES



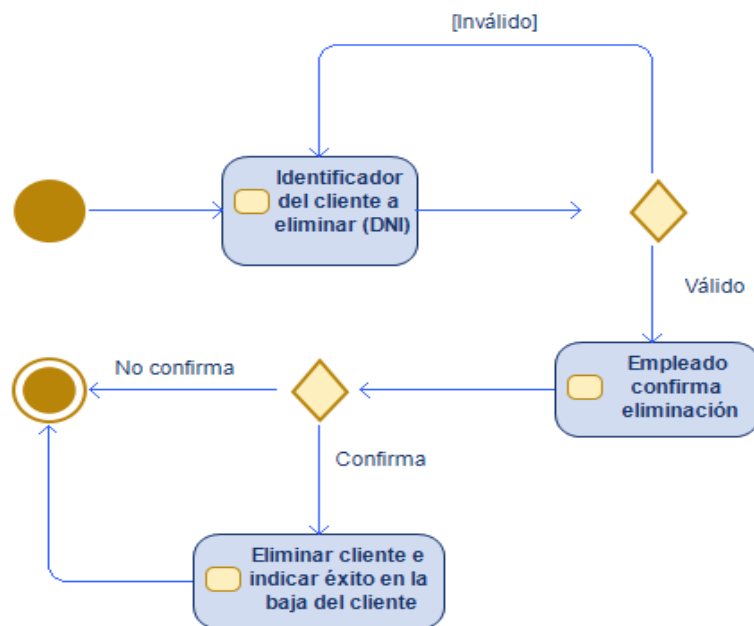
CU-CLI-01		Alta cliente	
Subsistema	Gestión de clientes		
Descripción	Dar de alta a un nuevo cliente		
Actores	Actor principal que invoca el Caso de Uso: Empleado Actores secundarios: Base de Datos		
Descrito por	Kei		
Prioridad	Media		
Estabilidad	Media		
Precondiciones	No hay DNI repetido en la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	Introducir nombre, correo, teléfono, dirección de entrega	
	2	Validar los datos ingresados	
	3	Verificar que no exista ya el cliente en la base de datos	
	4	Registrar al cliente en la base de datos	
	5	Mostrar al empleado los datos del cliente registrado	
Postcondiciones			
Éxito:	Nuevo cliente en la base de datos		
Fallo:	Enviar mensaje explicando la causa del fallo		
Flujo de eventos alternativo	Paso	Acción	
	2.a	Si los datos no son válidos, volver al paso 1 para corregirlos	
	2.b	Si el cliente ya existe, no introducir a la base de datos y comunicar al empleado	
Requisitos no funcionales	Consentimiento del cliente para almacenar sus datos		
Comentarios			



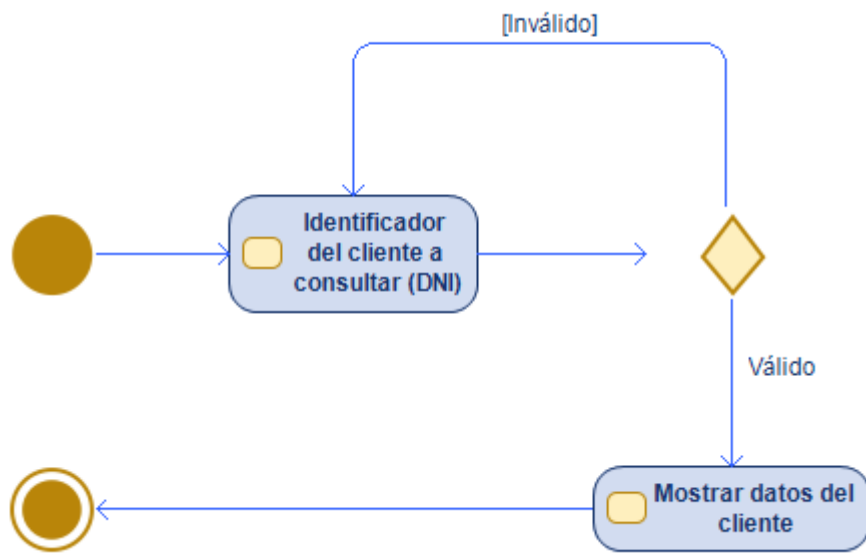
CU-CLI-02		Modificación cliente	
Subsistema	Gestión de clientes		
Descripción	Modificar los datos de un cliente existente		
Actores	Actor principal que invoca el Caso de Uso: Empleado Actores secundarios: Base de Datos		
Descrito por	Kei		
Prioridad	Media		
Estabilidad	Media		
Precondiciones	El cliente debe existir en la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	El empleado ingresa el identificador del cliente que desea modificar (DNI)	
	2	El sistema valida que el cliente existe en la base de datos	
	3	El empleado modifica los datos del cliente (nombre, correo, teléfono, dirección)	
	4	El sistema valida los nuevos datos ingresados	
	5	El sistema guarda los cambios en la base de datos y devuelve un mensaje de éxito	
Postcondiciones			
Éxito:	Los datos del cliente son modificados en la base de datos		
Fallo:	El sistema muestra un mensaje de error si la validación de datos falla		
Flujo de eventos alternativo	Paso		
	2.a	Si el cliente no existe, el sistema muestra un mensaje de error y solicita al empleado que ingrese un identificador válido	
Requisitos no funcionales	Los cambios deben realizarse de manera eficiente para evitar tiempos de espera largos		
Comentarios			



CU-CLI-03		Baja cliente	
Subsistema	Gestión de clientes		
Descripción	Eliminar a un cliente del sistema		
Actores	Actor principal que invoca el Caso de Uso: Empleado Actores secundarios: Base de Datos		
Descrito por	Kei		
Prioridad	Media		
Estabilidad	Media		
Precondiciones	El cliente debe existir en la base de datos		
Flujo de eventos principal	Paso	Acción	
	1	El empleado ingresa el identificador del cliente a eliminar	
	2	El sistema valida que el cliente existe en la base de datos	
	3	El empleado confirma la eliminación del cliente	
	4	El sistema elimina el cliente de la base de datos y devuelve un mensaje de éxito	
Postcondiciones	Éxito: El cliente es eliminado de la base de datos		
	Fallo: El sistema muestra un mensaje de error si el cliente no existe o si no se confirma la eliminación		
Flujo de eventos alternativo	Paso		
	2.a	Si el empleado no confirma la eliminación, el sistema no elimina al cliente y termina el proceso	
Requisitos no funcionales	La eliminación debe ser irreversible para evitar errores		
Comentarios			



CU-CLI-04		Consulta cliente	
Subsistema	Gestión de clientes		
Descripción	Dar de alta a un nuevo cliente		
Actores	Actor principal que invoca el Caso de Uso: Empleado Actores secundarios: Base de Datos		
Descrito por	Kei		
Prioridad	Media		
Estabilidad	Media		
Precondiciones	El cliente debe existir en la base de datos		
Flujo de eventos principal	Paso		
	1	El empleado ingresa el identificador del cliente	
	2	El sistema valida que el cliente existe en la base de datos	
	3	El sistema muestra los datos del cliente	
Postcondiciones			
Éxito:	Los datos del cliente son mostrados correctamente		
Fallo:	El sistema muestra un mensaje de error si el cliente no existe		
Flujo de eventos alternativo	Paso		
	2.a	Si el cliente no existe, el sistema muestra un mensaje de error y solicita al empleado que ingrese un identificador válido	
Requisitos no funcionales	La consulta debe ser rápida y eficiente, incluso si la base de datos es grande		
Comentarios			



3.3 Requisitos de rendimiento

Requisitos numéricos:

Número de terminales: hasta 50 terminales simultáneas.

Número de usuarios: hasta 200 usuarios.

Consultas por segundo: hasta 50 consultas por segundo

Actualizaciones por segundo: hasta 20 operaciones de guardado o modificado simultáneamente.

Tiempo de respuesta: Entre 2 y 5 segundos, depende de la operación y el tráfico en ese momento.

Requisitos no funcionales:

Consentimiento del empleado para almacenar sus datos, siguiendo las normas de la RGDP.

No modificar información inherente al trabajo de la empresa con la eliminación de los datos del empleado

No proceder con la devolución si el artículo está fuera de fecha, dañado o si es otro producto

Debe haber alguna necesidad de consultar la información correspondiente a una devolución pasada

Que haya algún motivo para querer eliminar los datos de una devolución ya efectuada

3.4 Requisitos lógicos de la base de datos

- **Pedidos de clientes:** ID (string), fecha de pedido (Date)
- **Ventas:** ID (string), fecha de entrega (Date), factura (int)
- **Productos:** ID (string), nombre (string), precio (int)
- **Clientes:** correo electrónico (string), nombre (string), teléfono (int), dirección (string)
- **Empleados:** DNI (string), nombre (string), teléfono (string), sueldo (int)
- **Compras:** ID (string), proveedor (string), factura (int)
- **Ofertas:** ID (string), descuento (int), fecha de finalización (Date)
- **Categorías de productos:** ID (string), nombre (string), descripción (string)

	Pedidos de clientes	Ventas	Productos	Clientes	Empleados	Compras	Ofertas	Categorías de productos
Pedidos de clientes		1	1...*	1				
Ventas	1		1...*	1	1			
Productos	*	1...*				1	1	1...*
Clientes	1...*	1...*						
Empleados		1...*				1...*		
Compras			1...*		1			
Ofertas			1...*					
Categorías de productos			1...*					

3.5 Restricciones de diseño

Para este proyecto hemos decidido seguir el estándar IEEE Std. 830-1998, ya que al no hacer uso de más especificaciones de requisitos aparte de esta SRS, no necesitamos un estándar más complejo. El software será desarrollado en el lenguaje Java, debido a que está enfocado en la programación orientada a objetos.

3.6 Atributos del sistema software

Fiabilidad:

El programa tendrá un alto nivel de fiabilidad, puesto que permite el acceso a la base de datos únicamente a los administradores y empleados autorizados.

Disponibilidad:

La disponibilidad de nuestro software será permanente al no requerir de acceso a internet para funcionar, pudiendo operar utilizando la información guardada en el sistema y actualizando la base de datos cuando se restablezca la conexión a la red.

Seguridad:

Gracias a nuestro grupo de desarrolladores, expertos en ciberseguridad, el software será completamente seguro, garantizando que nadie ajeno a la empresa pueda acceder al sistema.

Mantenibilidad:

Nuestro software tendrá un sistema de reporte de errores que da acceso a un panel de control el cual permite ver en tiempo real los fallos que se están produciendo en el programa para poder darles una solución lo más rápido posible.

El programa contará con actualizaciones de forma regular, las cuales se desarrollarán basándose en toda la información recogida en esos informes de errores y en la retroalimentación de nuestros clientes, ya sea para modificar o para incorporar funcionalidades al software que consideremos que pueden mejorar la experiencia de todos los usuarios.

Portabilidad:

En un principio el programa estará disponible únicamente para sistemas Windows, Linux y macOS, aunque uno de los objetivos del equipo de desarrollo es permitir que el software pueda ejecutarse en Android e iOS.

3.7 Otros requisitos

- Todos los módulos del software estarán vinculados de forma directa a la base de datos.
- Los administradores serán los únicos actores autorizados para acceder a todos los módulos del sistema.
- El desarrollo de cualquier nuevo módulo utilizará como base los módulos previamente presentes en el sistema.
- La interfaz de usuario estará diseñada para ser intuitiva y sencilla, para permitir que cualquier usuario pueda hacer uso del programa sin necesidad de tener ningún tipo de experiencia con éste.