

# Documento de **MODELADO DE REQUISITOS**



**Sistema de Administración y  
Gestión de RESTaurantes**



Samuel Guirado Navarro  
Carlos Moreno Muñoz  
Gaspar Muñoz Soria

V1.0

# ÍNDICE DE CONTENIDO

Apartado de control de versiones.....	4
1. Modelo funcional.....	5
<i>Identificación de actores.....</i>	<i>5</i>
Cocinero Jefe.....	5
Metre.....	5
<i>Identificación de los requerimientos funcionales.....</i>	<i>6</i>
<i>Descripción de los casos de uso.....</i>	<i>7</i>
Añadir elemento a la carta.....	7
Eliminar elemento a la carta.....	9
Modificar elemento de la carta.....	10
Consultar elementos de la carta.....	11
Imprimir lista de productos a pedir.....	12
Notificar incidencia con Ingrediente.....	13
Notificar incidencia con bebida.....	14
Notificar recepción de pedido del proveedor.....	15
Añadir ingrediente.....	16
Eliminar ingrediente.....	17
Modificar ingrediente.....	18
Añadir bebida.....	19
Eliminar bebida.....	20
Modificar bebida.....	21
Consultar lista de productos.....	22
Consultar lista de ingredientes.....	23
Consultar lista de bebidas.....	24
<i>Elaboración del Diagrama de casos de uso.....</i>	<i>25</i>
2. Subsistemas funcionales.....	26
<i>Identificación de los subsistemas funcionales.....</i>	<i>26</i>
<i>Elaboración del diagrama de paquetes para subsistemas funcionales.....</i>	<i>27</i>
Diagrama de casos de uso global con división en subsistemas.....	27
Subsistema de gestión de carta.....	28
Subsistema de gestión de productos.....	28
Subsistema de gestión de ingredientes.....	29
Subsistema de gestión de bebidas.....	29

<b>3. Requisitos no funcionales.....</b>	<b>30</b>
<i>Identificación y descripción de los requisitos no funcionales del sistema.....</i>	<i>30</i>
Facilidad de uso.....	30
Fiabilidad.....	30
Rendimiento.....	30
Soporte.....	31
Implementación.....	31
Interfaz.....	31
Operaciones.....	31
Legales.....	31
<b>4. Operaciones del sistema.....</b>	<b>32</b>
<i>Elaboración de diagramas de secuencia del sistema .....</i>	<i>32</i>
Añadir elemento a la carta.....	32
Eliminar elemento de la carta.....	33
Modificar elemento de la carta.....	34
Consultar elementos de la carta.....	35
Imprimir lista de productos a pedir.....	35
Notificar recepción de pedido del proveedor.....	36
Consultar lista de productos.....	36
Añadir ingrediente.....	37
Eliminar ingrediente.....	38
Modificar ingrediente.....	39
Consultar lista de ingredientes.....	40
Notificar incidencia con ingrediente.....	41
Añadir bebida.....	42
Eliminar bebida.....	43
Modificar bebida.....	44
Consultar lista de bebidas.....	45
Notificar incidencia con bebida.....	46
<i>Identificación de las operaciones del sistema .....</i>	<i>47</i>
<b>Apéndice 1.0.....</b>	<b>54</b>

## **APARTADO DE CONTROL DE VERSIONES**

Todas las versiones están especificadas a fondo en el apartado de “Apéndices”, al final de este documento, cada apéndice se corresponde en nombre con su número de versión. Por ejemplo, el “Apéndice 0.1” se corresponde con la versión v0.1. Para ver los cambios realizados sobre cada versión, hay que ir deshaciendo los cambios desde el final.

<i>Versión</i>	<i>Fecha</i>	<i>Descripción</i>
V1.0	15/03/10	Documento inicial

A stylized logo for 'Sagres' in a cursive script. Above the word is a decorative swirl that loops around the 'S' and extends upwards.

## 1. MODELO FUNCIONAL

---

### Identificación de actores

#### *Cocinero Jefe*

Es el encargado de gestionar la carta del restaurante. Entre sus tareas están las de añadir, modificar y eliminar elementos de la carta. También es el encargado de gestionar el stock de ingredientes del restaurante. Otra de sus responsabilidades será la de notificar incidencias que ocurran con ingredientes o bebidas que utilice en la elaboración de los platos del restaurante. También deberá informar de cuando llega un pedido del proveedor.

#### *Metre*

Es el encargado de gestionar todo lo referente al stock de bebidas del restaurante. Al igual que el cocinero jefe, éste también deberá notificar de cualquier incidencia relacionada con alguna bebida.

Sagres



## Identificación de los requerimientos funcionales

Estos son los requerimientos funciones obtenidos tras la reunión con el cliente asociados a los subsistemas estudiados en esta primera iteración:

**RF1.** El sistema contempla a tres tipos de usuario: cliente, camarero jefe y metre.

**RF2.** El cocinero jefe podrá añadir nuevos elementos a la carta aportando el nombre del nuevo elemento, la descripción y el precio de éste, así como los ingredientes y el gasto de elaboración asociados y una foto del mismo finalizado. Cada elemento deberá tener asignada una y solo una sección de la carta, las cuales serán: entradas, carnes, pescados, bebidas y postres. Además, en el caso de los elementos que no sean bebidas, el cocinero deberá especificar si estos se pueden partir en raciones o no, indicando el número de raciones en caso positivo.

**RF3.** El cocinero jefe ha de poder realizar pequeñas modificaciones en los datos (salvo en los ingredientes) de cualquier elemento de la carta, siempre y cuando el restaurante permanezca cerrado.

**RF4.** El cocinero jefe podrá también eliminar cualquier elemento de la carta. Para evitar la posibilidad de que el cocinero elimine un elemento mientras un cliente lo selecciona, esta operación solo podrá llevarse a cabo cuando el restaurante este cerrado.

**RF5.** El sistema debe ofrecer la posibilidad de que con solo pulsar un botón, se le pueda notificar que ha llegado el pedido del proveedor, actualizándose las cantidades de los productos pedidos automáticamente.

**RF6.** El cocinero jefe podrá añadir nuevos ingredientes al stock de productos través del sistema. Los datos que deberá proporcionar de cada ingrediente son el nombre, la cantidad actual de la que se dispondrá, el límite mínimo de cantidad que puede haber en stock, el límite máximo y una foto del mismo.

**RF7.** El cocinero jefe podrá modificar los datos de cualquier ingrediente.

**RF8.** El cocinero jefe también podrá eliminar ingredientes del sistema.

**RF9.** El sistema deberá llevar un control de la pérdidas que se produzcan en el restaurante debido a incidencias con productos y bebidas. Para ello, el sistema ofrecerá la posibilidad de poder notificar una incidencia ya sea con un ingrediente o con una bebida. Se deberá indicar tan solo la cantidad de producto afectado.

**RF10.** El metre será el encargado de añadir, modificar y eliminar las bebidas del stock de productos.

**RF11.** Los datos necesarios para añadir una bebida serán los mismos que para un ingrediente. Sin embargo, en este caso también será necesario añadir la cantidad de líquido que cabe en el envase en el que se encuentra dicha bebida.

**RF12.** El sistema deberá elaborar una lista de necesidades con las cantidades requeridas de cada uno de los productos que se deben comprar, a partir de los productos consumidos y del mínimo que debe haber de existencias. Además, el cocinero podrá imprimir esta lista cuando desee para que se pueda realizar el pedido.

**RF13.** El sistema tendrá la obligación de notificar al usuario cuando se produzca el hecho en el que la cantidad actual en stock de un producto supere su límite mínimo.

## Descripción de los casos de uso

### Añadir elemento a la carta

<b>Nombre del caso:</b>	Añadir elemento a la carta
<b>Resumen:</b>	Se añadirá un nuevo elemento a la carta del restaurante
<b>Dependencias:</b>	Incluye al caso de uso "Consultar lista de Productos"
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"> <li>Se han añadido uno o más elementos nuevos a la carta</li> </ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"> <li>El usuario selecciona la opción "Añadir elemento a la carta" en el terminal táctil.</li> <li>Se inicia el caso de uso "Consultar lista de Productos"</li> <li>Mientras el usuario desee añadir elementos a la carta: <ol style="list-style-type: none"> <li>El usuario rellena los datos correspondientes al elemento: <ol style="list-style-type: none"> <li>Introduce el nombre</li> <li>Introduce la descripción</li> <li>Introduce la foto</li> <li>Introduce el precio</li> <li>Introduce el tiempo de elaboración</li> <li>Selecciona la sección <ol style="list-style-type: none"> <li>Si la sección asignada no es la de bebidas <ol style="list-style-type: none"> <li>El usuario especifica si el plato se puede dividir en raciones</li> </ol> </li> </ol> </li> </ol> </li> <li>Mientras el usuario quiera añadir productos al elemento: <ol style="list-style-type: none"> <li>El usuario selecciona un producto de la lista</li> <li>El usuario introduce la cantidad deseada de dicho producto</li> </ol> </li> <li>El usuario confirma los datos</li> <li>El sistema comprueba que los datos introducidos sean correctos</li> <li>Los datos introducidos son correctos. El sistema genera un código asociado al elemento y registra todos los datos</li> <li>El sistema confirma que se ha añadido un nuevo elemento a la carta satisfactoriamente</li> <li>El sistema pregunta al usuario si desea seguir añadiendo más elementos a la carta</li> </ol> </li> <li>El usuario finaliza la operación</li> </ol>
<b>Cursos</b>	<ol style="list-style-type: none"> <li>El usuario se ha equivocado al seleccionar un producto <ol style="list-style-type: none"> <li>El usuario elimina el producto deseado</li> </ol> </li> </ol>

<i>alternativos:</i>	<p>3.2.1. El usuario se ha equivocado al introducir la cantidad de un ingrediente</p> <p>3.2.1.1. El usuario selecciona el producto de la lista</p> <p>3.2.1.2. El usuario introduce la nueva cantidad</p> <p>3.5. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1</p> <p>*. El usuario cancela la operación</p>
<i>Observaciones:</i>	
<i>Requisitos no funcionales específicos:</i>	<ul style="list-style-type: none"> <li>- Las secciones asignables a un elemento de la carta serán: <b>Entrantes, Carnes, Pescados, Bebidas o Postres</b></li> <li>- Las cantidades de los ingredientes asignados a los platos se establecerán en gramos en caso de ser sólidos y mililitros en caso de ser líquidos</li> <li>- <b>La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb</b></li> </ul>



Sagres



## Eliminar elemento a la carta

<b>Nombre del caso:</b>	Eliminar elemento de la carta
<b>Resumen:</b>	Se eliminarán uno o más elementos de la carta del restaurante
<b>Dependencias:</b>	Incluye al caso de uso "Consultar elementos de la carta"
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se han eliminado uno o más elementos de la carta</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Eliminar elemento de la carta" en el terminal táctil.</li><li>2. Se inicia el caso de uso "Consultar elementos de la carta"</li><li>3. Mientras el usuario quiera eliminar elementos de la carta:<ol style="list-style-type: none"><li>3.1 El usuario selecciona el elemento de la carta que desea eliminar</li><li>3.2 El sistema muestra la información asociada al elemento y solicita la confirmación de la operación</li><li>3.3 El usuario confirma la operación</li><li>3.4 El sistema confirma que el elemento ha sido eliminado de la carta satisfactoriamente</li><li>3.5 El sistema pregunta al usuario si desea eliminar más elementos de la carta</li><li>3.6 Si el usuario desea eliminar más elementos de la carta<ol style="list-style-type: none"><li>3.6.1 Se inicia el caso de uso "Consultar elementos de la carta"</li></ol></li></ol></li><li>4. El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<ol style="list-style-type: none"><li>3.3. El usuario no confirma la operación. Se vuelve al paso 3</li></ol> <p>*. El usuario cancela la operación</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales:</b>	

## Modificar elemento de la carta

<b>Nombre del caso:</b>	Modificar elemento de la carta
<b>Resumen:</b>	Se modificarán uno o más elementos de la carta del restaurante
<b>Dependencias:</b>	Incluye al caso de uso “Consultar elementos de la carta”
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>Se han modificado uno o más elementos de la carta</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>El usuario selecciona la opción "Modificar elemento de la carta" en el terminal táctil.</li><li>Se inicia el caso de uso “Consultar elementos de la carta”</li><li>Mientras el usuario necesite modificar elementos:<ol style="list-style-type: none"><li>El usuario selecciona un elemento de la lista</li><li>El sistema muestra la información asociada a ese elemento</li><li>Mientras el usuario desee modificar datos del elementos:<ol style="list-style-type: none"><li>El usuario modifica el dato deseado</li></ol></li><li>El usuario finaliza la modificación de los datos</li><li>El sistema solicita la confirmación de la modificación de los datos</li><li>El usuario confirma la operación</li><li>El sistema comprueba que los datos introducidos son correctos.</li><li>Los datos introducidos son correctos. El sistema registra los nuevos cambios y confirma que el elemento ha sido modificado con éxito</li><li>El sistema pregunta al usuario si desea modificar otro elemento</li></ol></li><li>El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<ol style="list-style-type: none"><li><ol style="list-style-type: none"><li>El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.2</li></ol></li><li>*. El usuario cancela la operación</li></ol>
<b>Observaciones:</b>	
<b>Requisitos no funcionales:</b>	

### Consultar elementos de la carta

<b>Nombre del caso:</b>	Consultar elementos de la carta
<b>Resumen:</b>	Se obtendrá una lista con los elementos que forman la carta del restaurante
<b>Dependencias:</b>	Es incluido por los casos de uso “Modificar elemento de la carta” y “Eliminar elemento de la carta”
<b>Actores:</b>	Cocinero jefe, Metre y Cliente
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha generado una lista con los elementos que forman la carta del restaurante</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El caso de uso se inicia cuando el usuario desea conocer cuales son los elementos que componen la carta</li><li>2. El sistema proporciona una lista detallada de todos los elementos incluidos en la carta</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

### Imprimir lista de productos a pedir

<b>Nombre del caso:</b>	Imprimir lista de productos a pedir
<b>Resumen:</b>	Se imprimirá una lista de los productos que están bajo mínimos en stock, para poder hacer un pedido al proveedor
<b>Dependencias:</b>	
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha imprimido la lista de productos a pedir</li><li>• Se ha guardado una copia de la lista en el sistema</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Imprimir lista de productos a pedir" en el terminal táctil.</li><li>2. El sistema obtiene la lista de productos que están bajo mínimos en stock</li><li>3. El sistema imprime dicha lista</li><li>4. El sistema guarda una copia de la lista</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	La lista de productos estará dividida en dos partes: por un lado los ingredientes y por otro lado las bebidas

## Notificar incidencia con Ingrediente

<b>Nombre del caso:</b>	Notificar incidencia con Ingrediente
<b>Resumen:</b>	Se notifica que ha ocurrido un problema con algún ingrediente
<b>Dependencias:</b>	Incluye el caso de uso “Consultar lista de Ingredientes”
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha registrado una incidencia con un ingrediente</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Notificar incidencia con Ingrediente" en el terminal táctil.</li><li>2. Se inicia el caso de uso “Consultar lista de Ingredientes”</li><li>3. Mientras el usuario desee notificar incidencias:<ol style="list-style-type: none"><li>3.1. El usuario selecciona el ingrediente sobre el que desea notificar la incidencia</li><li>3.2. El usuario introduce la cantidad de producto afectado y especifica el motivo de la incidencia</li><li>3.3. El sistema comprueba que los datos introducidos son correctos.</li><li>3.4. Los datos introducidos son correctos. El sistema genera un código asociado a la incidencia y registra la incidencia</li><li>3.5. El sistema resta del stock la cantidad del ingrediente especificado por el usuario</li><li>3.6. Si la cantidad actual en stock del producto afectado se encuentra por debajo de su mínimo<ol style="list-style-type: none"><li>3.6.1 El sistema informa de que la cantidad ese producto en stock se encuentra por debajo de su mínimo</li></ol></li><li>3.7. El sistema informa de que la incidencia se ha registrado correctamente</li><li>3.8. El sistema pregunta al usuario si desea notificar alguna incidencia más</li></ol></li><li>4. El usuario finaliza las notificaciones</li></ol>
<b>Cursos alternativos:</b>	<ol style="list-style-type: none"><li>3.4 El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1</li></ol> <p>*. El usuario cancela la operación</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales:</b>	



## Notificar incidencia con bebida

<b>Nombre del caso:</b>	Notificar incidencia con Bebida
<b>Resumen:</b>	Se notifica que ha ocurrido un problema con alguna bebida
<b>Dependencias:</b>	Incluye el caso de uso "Consultar lista de Bebidas"
<b>Actores:</b>	Cocinero jefe, Metre
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>Se ha registrado una incidencia con una bebida</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>El usuario selecciona la opción "Notificar incidencia con Bebida" en el terminal táctil.</li><li>Se inicia el caso de uso "Consultar lista de Bebidas"</li><li>Mientras el usuario desee notificar incidencias:<ol style="list-style-type: none"><li>El usuario selecciona la bebida sobre la que desea notificar la incidencia</li><li>El usuario introduce la cantidad de producto afectado y especifica el motivo de la incidencia</li><li>El sistema comprueba que los datos introducidos son correctos.</li><li>Los datos introducidos son correctos. El sistema genera un código asociado a la incidencia y registra la incidencia</li><li>El sistema resta del stock la cantidad del ingrediente especificado por el usuario</li><li>Si la cantidad actual en stock del producto afectado se encuentra por debajo de su mínimo<ol style="list-style-type: none"><li>El sistema informa de que la cantidad ese producto en stock se encuentra por debajo de su mínimo</li></ol></li><li>El sistema informa de que la incidencia se ha registrado correctamente</li><li>El sistema pregunta al usuario si desea notificar alguna incidencia más</li></ol></li><li>El usuario finaliza las notificaciones</li></ol>
<b>Cursos alternativos:</b>	<ol style="list-style-type: none"><li><ol style="list-style-type: none"><li>El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1</li></ol></li><li>*. El usuario cancela la operación</li></ol>
<b>Observaciones:</b>	El cocinero podrá notificar incidencias de bebidas puesto que puede utilizarlas como ingredientes
<b>Requisitos no funcionales:</b>	

### Notificar recepción de pedido del proveedor

<b>Nombre del caso:</b>	Notificar recepción de pedido del proveedor
<b>Resumen:</b>	Se notifica la llegada del pedido y se actualizan las cantidades de los productos en stock que han sido pedidos
<b>Dependencias:</b>	
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha modificado la cantidad actual en stock de uno o más productos</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Notificar recepción de pedido del proveedor" en el terminal táctil.</li><li>2. El sistema obtiene los datos del pedido</li><li>3. El sistema actualiza las cantidades de los productos especificados en el pedido</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	El sistema utiliza la copia que posee del pedido para actualizar las cantidades de los productos correspondientes
<b>Requisitos no funcionales:</b>	

## Añadir ingrediente

<b>Nombre del caso:</b>	Añadir ingrediente
<b>Resumen:</b>	Se añaden uno o más ingredientes
<b>Dependencias:</b>	
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha registrado un nuevo producto en el sistema</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Añadir ingrediente" en el terminal táctil.</li><li>2. Mientras que el usuario quiera añadir ingredientes:<ol style="list-style-type: none"><li>2.1. El usuario introduce los datos correspondientes al nuevo ingrediente:<ol style="list-style-type: none"><li>2.1.1 Introduce el nombre</li><li>2.1.2 Introduce el límite mínimo en stock.</li><li>2.1.3 Introduce el máximo en stock.</li><li>2.1.4 Introduce la cantidad actual en stock</li><li>2.1.5 Introduce una foto.</li></ol></li><li>2.2. El usuario confirma los datos.</li><li>2.3. El sistema comprueba si los datos introducidos son correctos</li><li>2.4. Los datos introducidos son correctos. El sistema genera un código de producto, registra todos los datos y confirma que ha sido añadido satisfactoriamente.</li><li>2.5. El sistema pregunta al usuario si desea seguir añadiendo productos</li></ol></li><li>3. El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>2.4b El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	<ul style="list-style-type: none"><li>- Tanto los límites como la cantidad actual se deberán dar en gramos.</li><li>- La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb</li></ul>

## Eliminar ingrediente

<b>Nombre del caso:</b>	Eliminar ingrediente
<b>Resumen:</b>	Se eliminarán uno o más ingredientes.
<b>Dependencias:</b>	Incluye al caso de uso "Consultar lista de ingredientes"
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se han eliminado uno o más productos del sistema.</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Eliminar producto" en el terminal táctil.</li><li>2. Se inicia el caso de uso "Consultar lista de ingredientes"</li><li>3. Mientras que el usuario quiera eliminar ingredientes:<ol style="list-style-type: none"><li>3.1. El usuario selecciona el ingrediente que quiere eliminar.</li><li>3.2. El sistema solicita la confirmación de la operación.</li><li>3.3. El usuario confirma la operación.</li><li>3.4. El sistema elimina el ingrediente e informa de ello al usuario.</li><li>3.5. El sistema pregunta al usuario si desea eliminar otro elemento</li><li>3.6. Si el usuario quiere eliminar otro elemento<ol style="list-style-type: none"><li>3.6.1 Se inicia el caso de uso "Consultar lista de Productos"</li></ol></li></ol></li><li>4. El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>3.3b El usuario no confirma la operación. Se vuelve al paso 2.</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales:</b>	

## Modificar ingrediente

<b>Nombre del caso:</b>	Modificar ingrediente
<b>Resumen:</b>	Se modificarán los parámetros deseados de uno o más ingredientes
<b>Dependencias:</b>	Incluye al caso de uso “Consultar lista de ingredientes”
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>Se ha registrado uno o más cambios en los detalles de un ingrediente.</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>El usuario selecciona la opción "Modificar ingrediente" en el terminal táctil.</li><li>Se inicia el caso de uso “Consultar lista de ingredientes”</li><li>Mientras el usuario quiera modificar ingredientes:<ol style="list-style-type: none"><li>El usuario selecciona el ingrediente que quiere modificar.</li><li>El sistema muestra los datos asociados a ese ingrediente.</li><li>Mientras el usuario quiera modificar parámetros:<ol style="list-style-type: none"><li>El usuario modifica el parámetro deseado.</li></ol></li><li>El sistema solicita al usuario que confirme los cambios</li><li>El usuario confirma los cambios.</li><li>El sistema comprueba si los datos introducidos son correctos</li><li>Los datos introducidos son correctos. El sistema registra los cambios e informa que se han realizado satisfactoriamente.</li><li>El sistema pregunta al usuario si desea modificar otro ingrediente</li><li>Si el usuario solicita modificar otro ingrediente<ol style="list-style-type: none"><li>Se inicia el caso de uso “Consultar lista de ingrediente”</li></ol></li></ol></li><li>El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>3.5b El usuario no confirma los cambios. Se vuelve al paso 2</p> <p>3.6b El sistema informa de que los datos introducidos son incorrectos. Se vuelve al paso 2.</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	



## Añadir bebida

<b>Nombre del caso:</b>	Añadir bebida
<b>Resumen:</b>	Se añaden una o más bebidas
<b>Dependencias:</b>	
<b>Actores:</b>	Metre
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>• Se ha registrado una nueva bebida en el sistema</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción "Añadir bebida" en el terminal táctil.</li><li>2. Mientras que el usuario quiera añadir bebidas:<ol style="list-style-type: none"><li>2.1. El usuario introduce los datos correspondientes a la bebida:<ol style="list-style-type: none"><li>2.1.1 Introduce el nombre</li><li>2.1.2 Introduce el límite mínimo en stock.</li><li>2.1.3 Introduce el máximo en stock.</li><li>2.1.4 Introduce la cantidad actual en stock</li><li>2.1.5 Introduce una foto.</li><li>2.1.6 El usuario deberá introducir la cantidad por envase.</li></ol></li><li>2.2. El usuario confirma los datos.</li><li>2.3. El sistema comprueba si los datos introducidos son correctos</li><li>2.4. Los datos introducidos son correctos. El sistema genera un código de producto y registra todos los datos.</li><li>2.5. El sistema confirma que se ha añadido la bebida satisfactoriamente.</li><li>2.6. El sistema pregunta al usuario si desea seguir añadiendo bebidas.</li></ol></li><li>3. El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>2.4b El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	<ul style="list-style-type: none"><li>- Los límites mínimo y máximo vendrán en número de unidades de envase, y la cantidad por envase en mililitros.</li><li>- La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb</li></ul>

## Eliminar bebida

<b>Nombre del caso:</b>	Eliminar bebida
<b>Resumen:</b>	Se eliminarán una o más bebidas.
<b>Dependencias:</b>	Incluye al caso de uso "Consultar lista de bebidas"
<b>Actores:</b>	Metre.
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>Se han eliminado uno o más bebidas del sistema.</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>El usuario selecciona la opción "Eliminar bebida" en el terminal táctil.</li><li>Se inicia el caso de uso "Consultar lista de bebidas"</li><li>Mientras que el usuario quiera eliminar bebidas:<ol style="list-style-type: none"><li>El usuario selecciona la bebida que quiere eliminar.</li><li>El sistema solicita la confirmación de la operación.</li><li>El usuario confirma la operación.</li><li>El sistema elimina la bebida informa de ello al usuario.</li><li>El sistema pregunta al usuario si desea eliminar otro elemento</li><li>Si el usuario quiere eliminar otro elemento<ol style="list-style-type: none"><li>Se inicia el caso de uso "Consultar lista de bebidas"</li></ol></li></ol></li><li>El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>3.3b El usuario no confirma la operación. Se vuelve al paso 2.</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

## Modificar bebida

<b>Nombre del caso:</b>	Modificar bebida
<b>Resumen:</b>	Se modificarán los parámetros deseados de una o más bebidas
<b>Dependencias:</b>	Incluye al caso de uso "Consultar lista de bebidas"
<b>Actores:</b>	Metre
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	<ul style="list-style-type: none"><li>Se ha registrado uno o más cambios en los detalles de una bebida.</li></ul>
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>El usuario selecciona la opción "Modificar bebida" en el terminal táctil.</li><li>Se inicia el caso de uso "Consultar lista de bebidas"</li><li>Mientras el usuario quiera modificar bebidas:<ol style="list-style-type: none"><li>El usuario selecciona la bebida que quiere modificar.</li><li>El sistema muestra los datos asociados a esa bebida.</li><li>Mientras el usuario quiera modificar parámetros:<ol style="list-style-type: none"><li>El usuario modifica el parámetro deseado.</li></ol></li><li>El sistema solicita al usuario que confirme los cambios</li><li>El usuario confirma los cambios.</li><li>El sistema comprueba si los datos introducidos son correctos</li><li>Los datos introducidos son correctos. El sistema registra los cambios e informa que se han realizado satisfactoriamente.</li><li>El sistema pregunta al usuario si desea modificar otra bebida</li><li>Si el usuario solicita modificar otra bebida<ol style="list-style-type: none"><li>Se inicia el caso de uso "Consultar lista de bebidas"</li></ol></li></ol></li><li>El usuario finaliza la operación</li></ol>
<b>Cursos alternativos:</b>	<p>3.5b El usuario no confirma los cambios. Se vuelve al paso 3.2</p> <p>3.7b El sistema informa de que los datos introducidos son incorrectos. Se vuelve al paso 3.2.</p> <p>*. El usuario cancela la operación.</p>
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

### Consultar lista de productos

<b>Nombre del caso:</b>	Consultar lista de Productos
<b>Resumen:</b>	Se obtiene una lista con los productos registrados en el sistema.
<b>Dependencias:</b>	Es incluido por los casos de uso “Añadir Elemento a la Carta”. Incluye a los casos de uso “Consultar lista de bebidas” y “Consultar lista de ingredientes”
<b>Actores:</b>	Cocinero Jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El caso de uso se inicia cuando el usuario desea ver la lista de productos registrados en el sistema</li><li>2. Se inicia el caso de uso 'Consultar lista de bebidas'</li><li>3. Se inicia el caso de uso 'Consultar lista de ingredientes'</li><li>4. El sistema combina las dos listas y muestra una lista completa de todos los productos.</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

### Consultar lista de ingredientes

<b>Nombre del caso:</b>	Consultar lista de ingredientes
<b>Resumen:</b>	Se obtiene una lista con los productos registrados en el sistema dependiendo del tipo de usuario que la solicite
<b>Dependencias:</b>	Es incluido por los casos de uso “Eliminar ingrediente”, “Modificar ingrediente”, “Notificar incidencia con ingrediente”, “Añadir Elemento a la Carta” y “Consultar lista de productos”
<b>Actores:</b>	Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El caso de uso se inicia cuando el usuario desea ver la lista de productos registrados en el sistema</li><li>2. El sistema muestra una lista completa de todos los ingredientes registrados</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

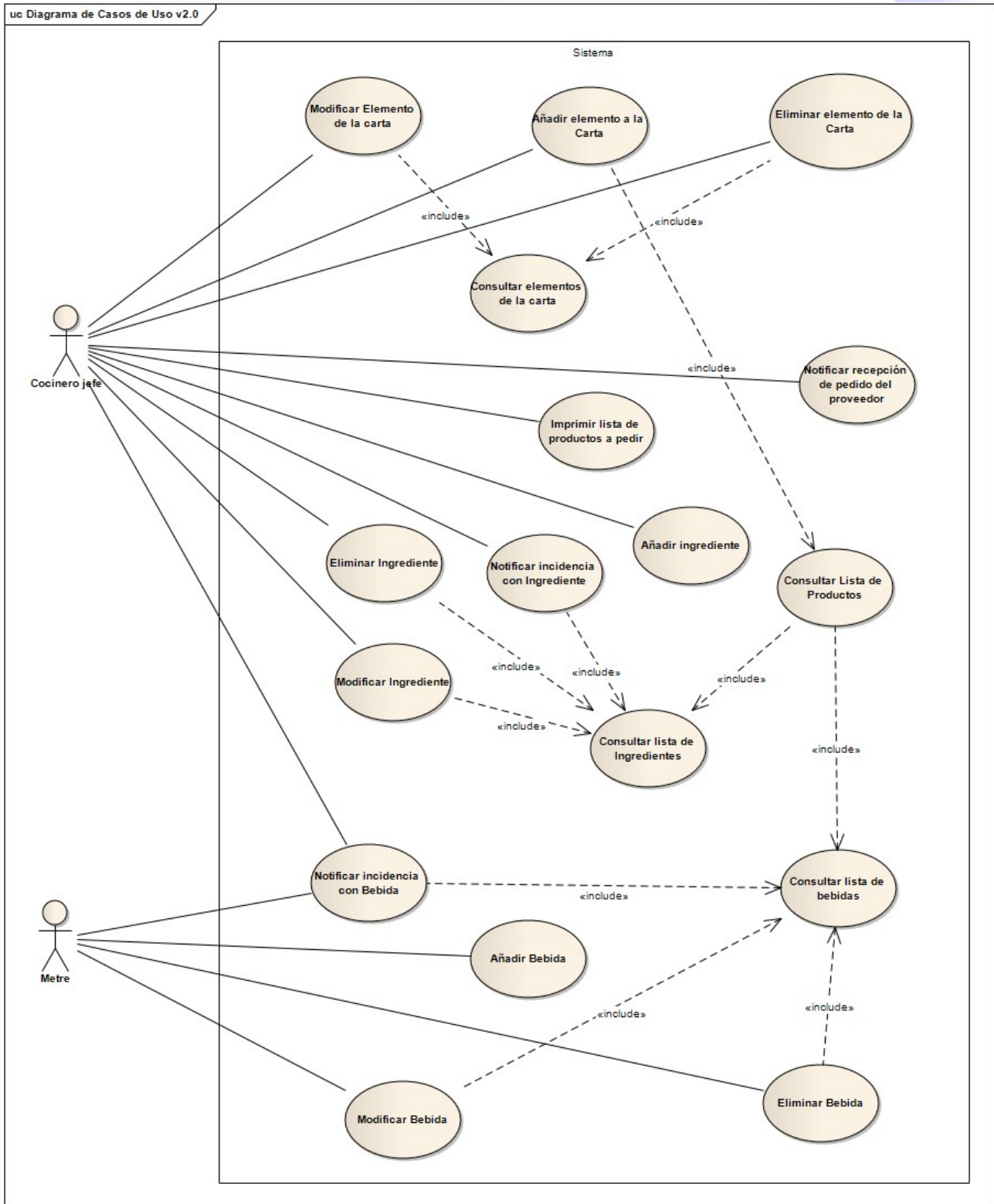


### Consultar lista de bebidas

<b>Nombre del caso:</b>	Consultar lista de bebidas
<b>Resumen:</b>	Se obtiene una lista con las bebidas registradas
<b>Dependencias:</b>	Es incluido por los casos de uso “Eliminar bebidas”, “Modificar bebidas”, “Notificar incidencia con bebida”, “Añadir Elemento a la Carta” y “Consultar lista de productos”
<b>Actores:</b>	Metre, Cocinero jefe
<b>Precondiciones:</b>	
<b>Postcondiciones:</b>	
<b>Curso normal:</b>	<ol style="list-style-type: none"><li>1. El caso de uso se inicia cuando el usuario desea ver la lista de bebidas registradas en el sistema</li><li>2. El sistema muestra una lista completa de todas las bebidas registradas</li></ol>
<b>Cursos alternativos:</b>	
<b>Observaciones:</b>	
<b>Requisitos no funcionales específicos:</b>	

## Elaboración del Diagrama de casos de uso

Los casos de uso anteriormente especificados, los podemos representar en el siguiente diagrama global de casos de uso, para concluir la elaboración del modelo funcional de nuestro sistema.



## 2. SUBSISTEMAS FUNCIONALES

---

### **Identificación de los subsistemas funcionales**

Para gestionar y optimizar todas las funcionalidades del sistema expuestas en el apartado de “Requerimientos Funcionales”, se han dividido estas en dos subsistemas.

Si analizamos el diagrama de casos de usos global, podemos ver estas dos grandes agrupaciones o subsistemas con claridad. La primera de ellas se corresponde a los casos de uso referentes a la carta de nuestro sistema, y la segunda concierne a la gestión de los productos que se usarán en el día a día del restaurante.

Este último, a su vez, podemos fraccionarlo en dos nuevos subsistemas hijos, a los que llamaremos “subsistema de gestión de bebidas” y “subsistema de gestión de ingredientes”. El primero alude al trabajo propio del metre, que será el encargado de la administración de las bebidas del stock. El último atañe a la tarea del jefe de cocina, que además de gestionar la carta también deberá encargarse de la administración de los ingredientes.

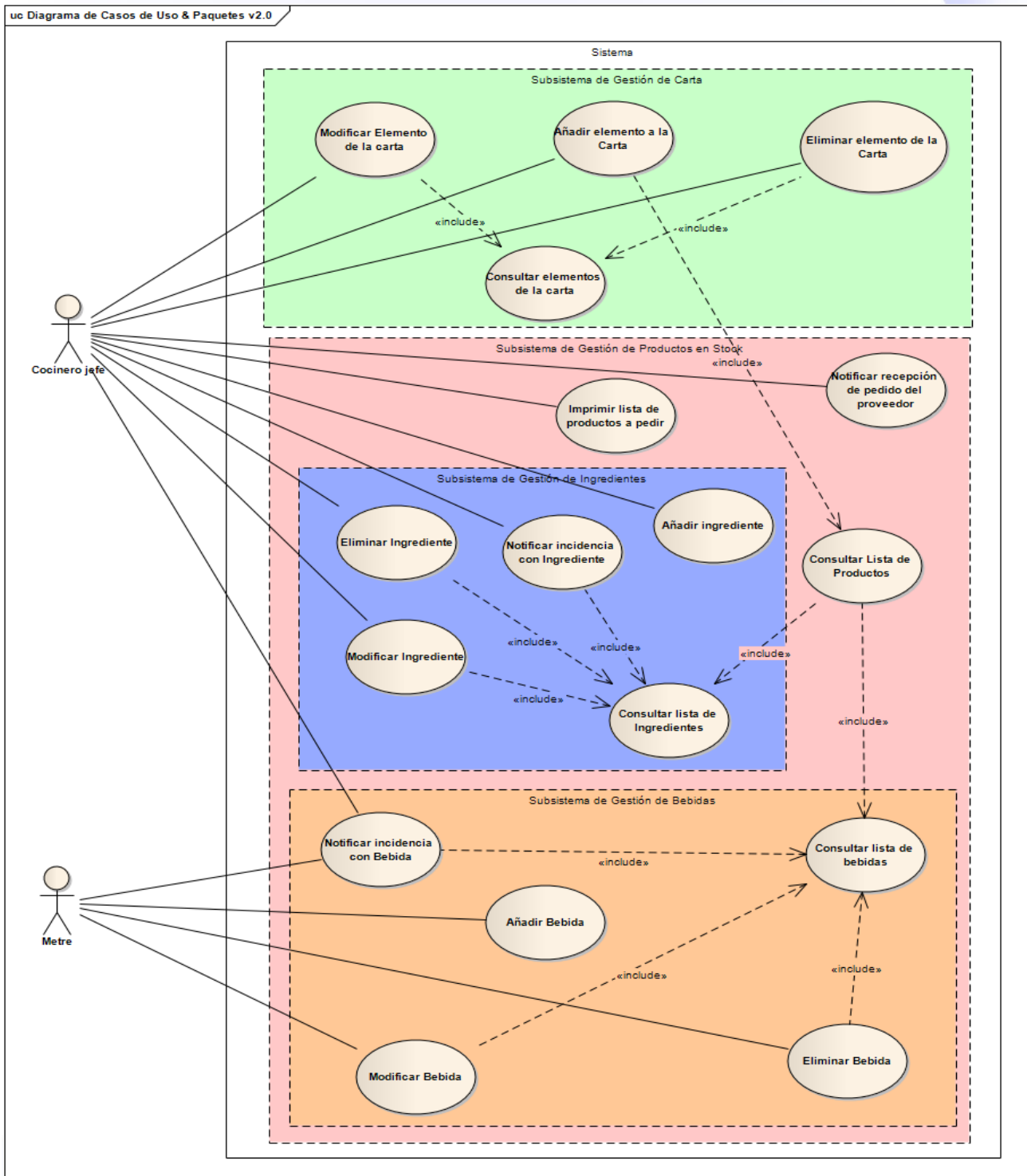
Con esta división de subsistemas y el específico reparto de tareas entre cocinero jefe y metre nos ahorramos tener que implementar un sistema autenticación, ya que debido al escaso número de tipos de usuarios (tan solo 3) sería una pérdida de tiempo.

Sagres

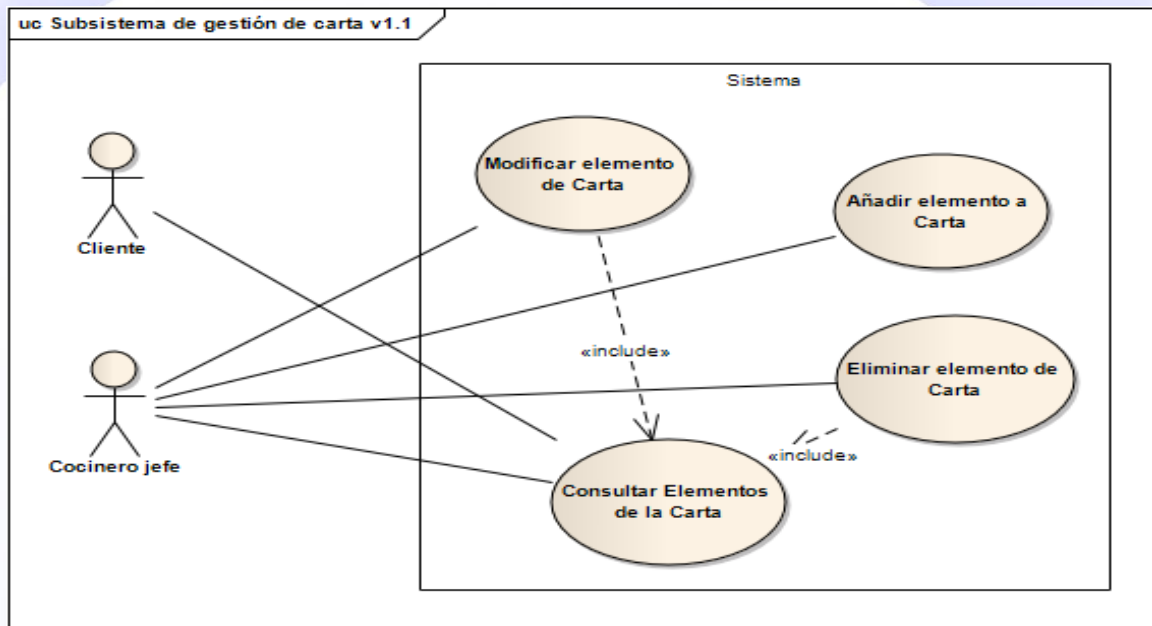
## Elaboración del diagrama de paquetes para subsistemas funcionales

Una vez identificados nuestros subsistemas, los plasmaremos gráficamente en los diagramas de paquetes.

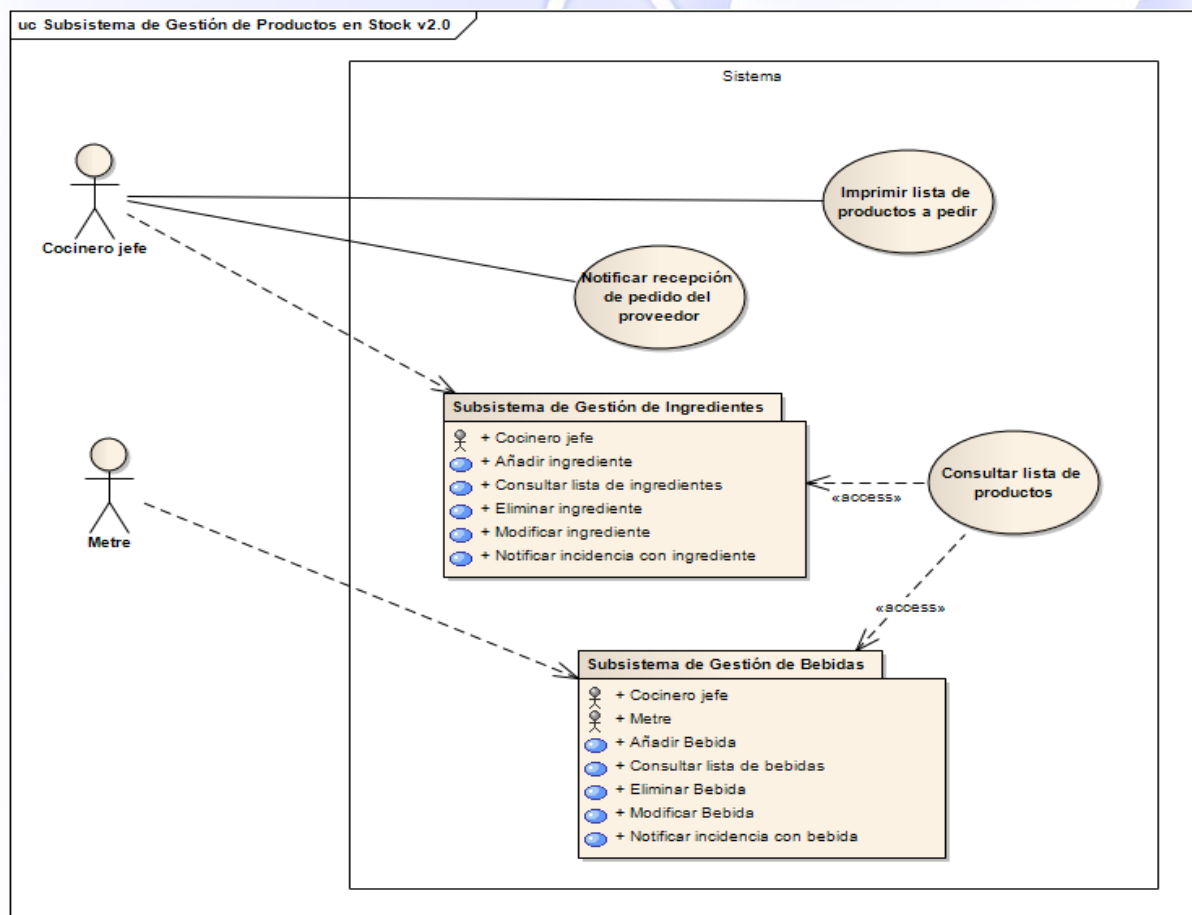
### Diagrama de casos de uso global con división en subsistemas



## Subsistema de gestión de carta



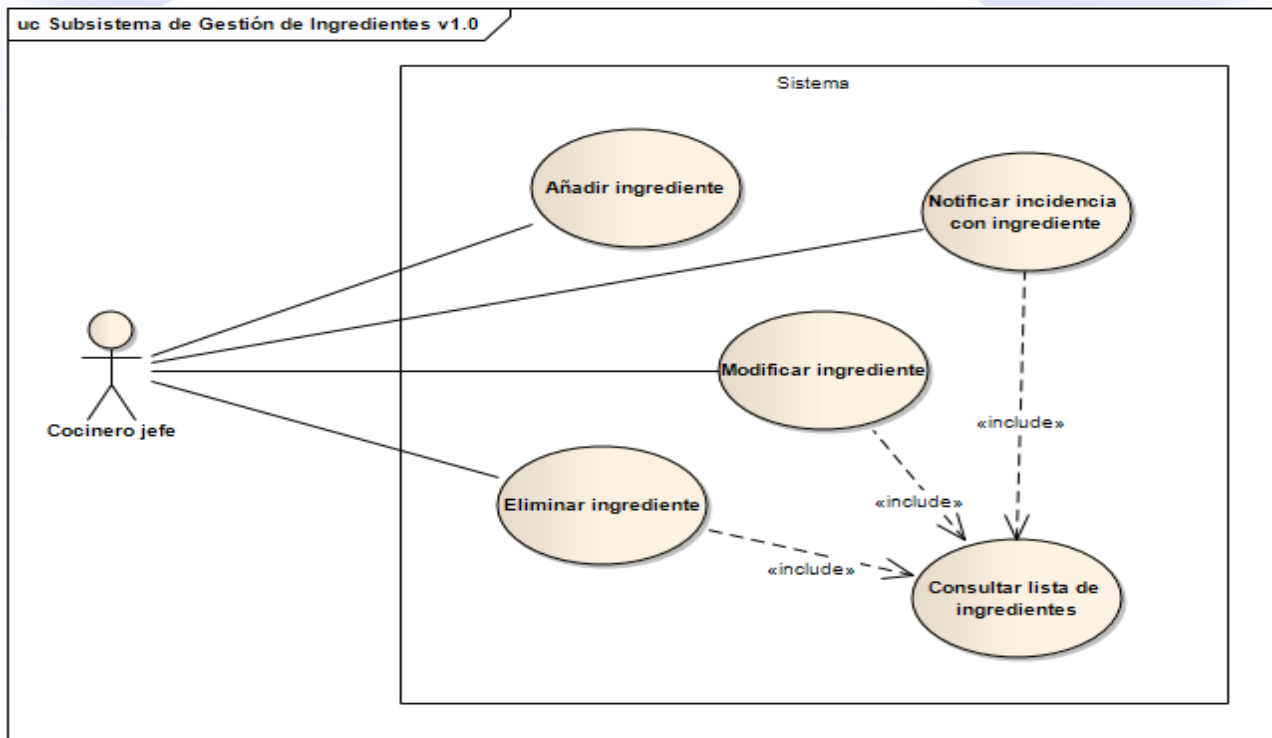
## Subsistema de gestión de productos



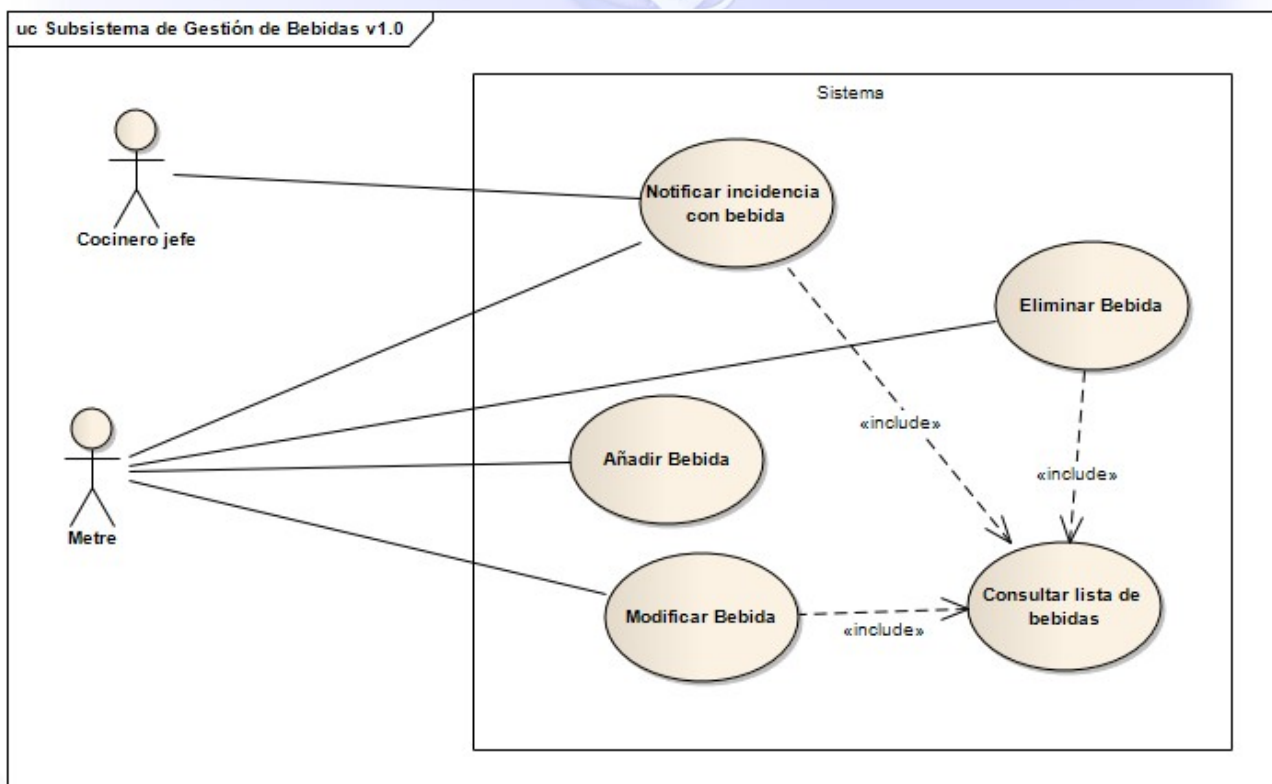
Este sería el diagrama de todo el subsistema de productos, pero como anteriormente comentamos se fracciona en dos nuevos, que son los que veremos a continuación.



## Subsistema de gestión de ingredientes



## Subsistema de gestión de bebidas



### 3. REQUISITOS NO FUNCIONALES

---

#### **Identificación y descripción de los requisitos no funcionales del sistema**

##### *Facilidad de uso*

- El sistema debe tener una interfaz de usuario amigable e intuitiva. Un usuario sin conocimientos informáticos deberá poder manejar la aplicación correctamente.
- Al ejecutarse sobre un terminal táctil, la interacción con el usuario se hará mediante un diseño fácil de utilizar y visible.
- El entorno deberá mostrar una breve información sobre cada una de las tareas (tips) que podemos realizar, preferiblemente al estar sobre una opción un determinado tiempo.
- El usuario dispondrá de un sencillo manual de uso sobre toda la funcionalidad del sistema.

##### *Fiabilidad*

- El sistema debe tener un grado alto de fiabilidad y robustez.
- Se debe prevenir y tratar cualquier error, mostrando un mensaje de información acerca de lo ocurrido, es decir, garantizamos la correcta captura de excepciones.
- El sistema deberá advertir ante posibles operaciones o acciones inválidas o erróneas que puedan provocar errores.
- Para prevenir de una caída del sistema y/o pérdidas de información, haremos copias de seguridad cada cierto tiempo de nuestros datos.
- Por ello, el número mayor de datos que podemos perder es el de los guardados desde la última copia de seguridad de nuestra base de datos.

##### *Rendimiento*

- No hay un tiempo de respuesta determinado hacia tareas en concreto pero al no requerir de cálculos u operaciones complejas deberá ser un tiempo eficiente y rápido.
- El tamaño de espacio ocupado en memoria masiva en el servidor de la base de datos irá en función de la cantidad de información almacenada en el sistema y no variará mucho respecto a otros sistemas que incluyan una base de datos.
- Se permitirá la ejecución de varios clientes concurrentemente y se garantizará la atención correcta y precisa de las peticiones de estos por parte de nuestro servidor.
- La peor situación aceptable para un usuario será la de que sus peticiones tarden un tiempo más de lo normal en ser atendidas.

## Soporte

- El producto final será soportado en cualquier equipo con la máquina virtual de java instalada y donde pueda correr una versión compatible del gestor de la base de datos.
- Deberá ser fácilmente actualizable. Las tareas de mantenimiento, tales como actualizaciones a nuevos entornos hardware, serán resueltas por los programadores.

## Implementación

- La plataforma hardware consistirá en un terminal táctil con conexión permanente a la red del servidor en el caso de los terminales de las mesas, cocina y bar; y a Internet en el caso de los terminales de las habitaciones.
- Cada cliente web deberá contar con conexión de red al servidor (generalmente Internet u lan que soporte conectividad tcp/ip).
- El lenguaje de programación empleado será java y en la parte del cliente web utilizaremos php + mysql.
- Para la implementación del código de la aplicación se utilizará el IDE gratuito NetBeans, ya que dispone de una opción que nos permitirá utilizar subversión, quedando los archivos de código del proyecto distribuidos en una misma localización en Internet.
- Para dicho subversión, se utilizará la aplicación web de uso gratuita Google Code para almacenar los archivos y SVN Tortoise para gestionar estos últimos (también de uso gratuito).

## Interfaz

- El sistema no interactuará con otro sistema externo. Los datos importados serán introducidos por un usuario mediante los menús gráficos aportados por el sistema y de forma táctil o bien provendrán de los equipos clientes vía red local o Internet. Los datos se exportarán desde la aplicación al cliente también web por red local o Internet.

## Operaciones

- El sistema al iniciarse dispondrá de la información contenida en su base de datos.
- Los administradores interaccionarán con el sistema cuando surja algún cambio imprevisto que el sistema no sea capaz de detectar.

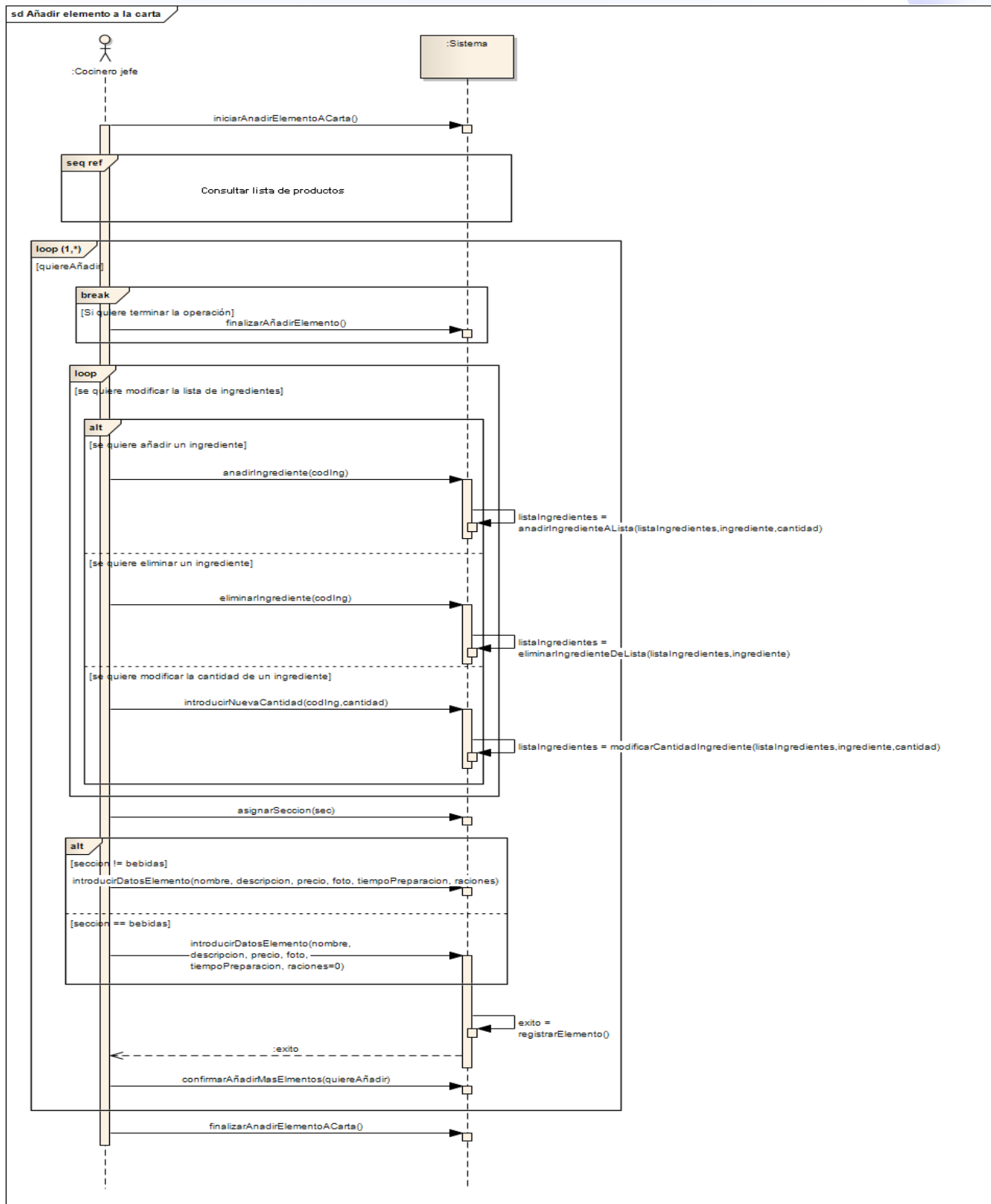
## Legales

- El sistema debe cumplir las disposiciones recogidas en la Ley Orgánica de Datos Personales y en el Reglamento de medidas de seguridad.
- Al usar software gratuito para el desarrollo del proyecto no será necesaria la compra de licencias de ningún tipo.

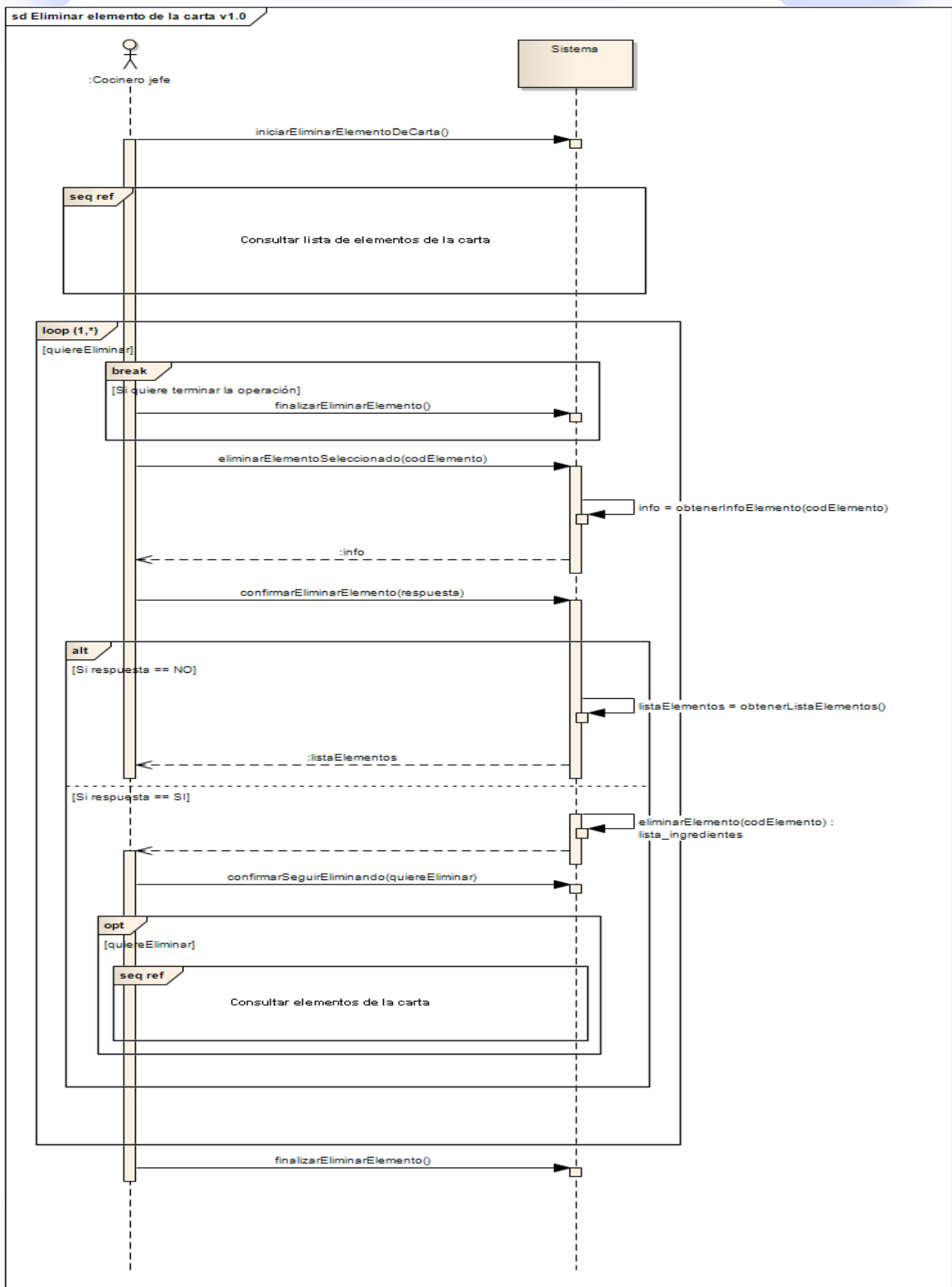
## 4. OPERACIONES DEL SISTEMA

### Elaboración de diagramas de secuencia del sistema

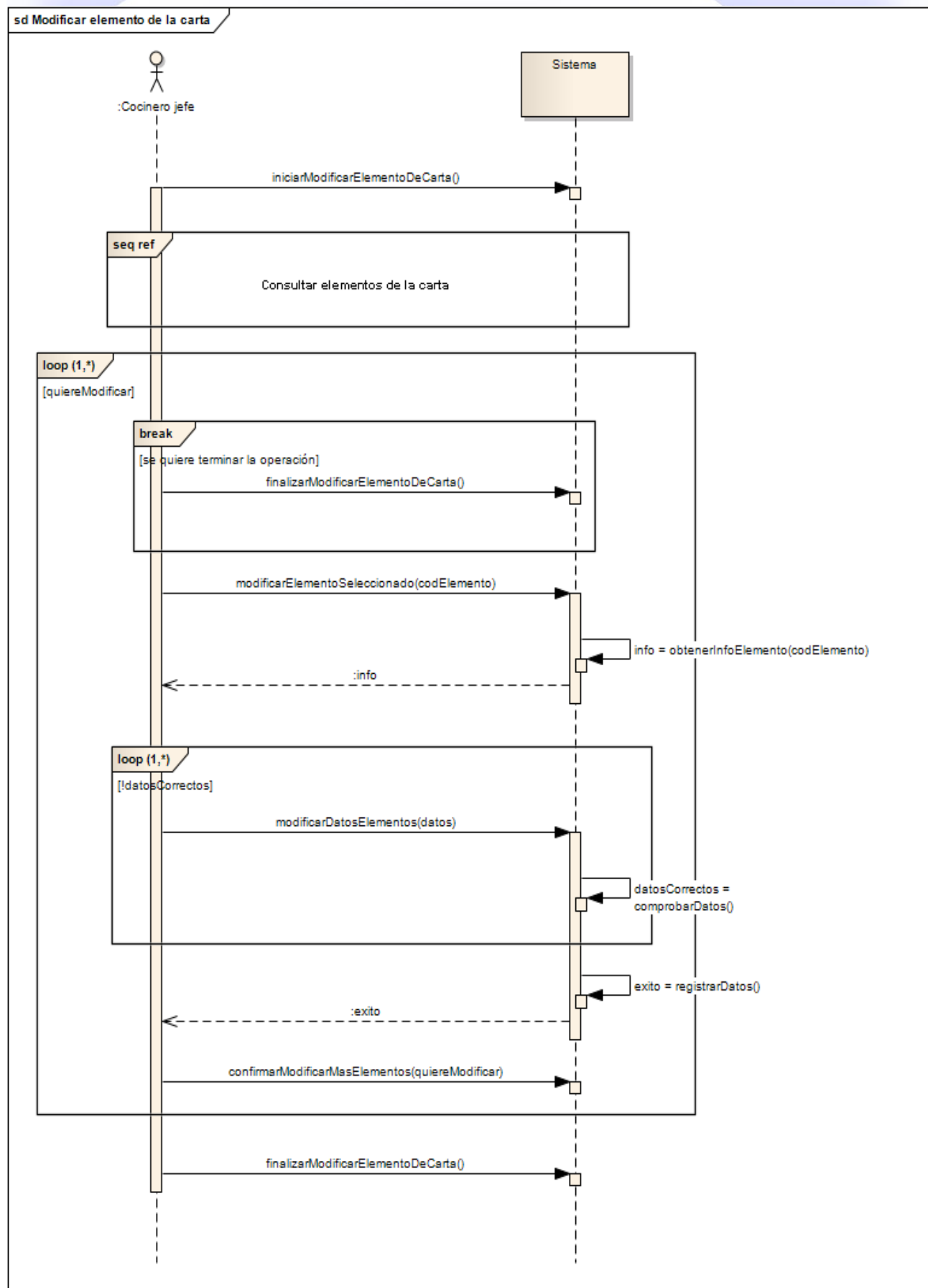
#### Añadir elemento a la carta



## Eliminar elemento de la carta

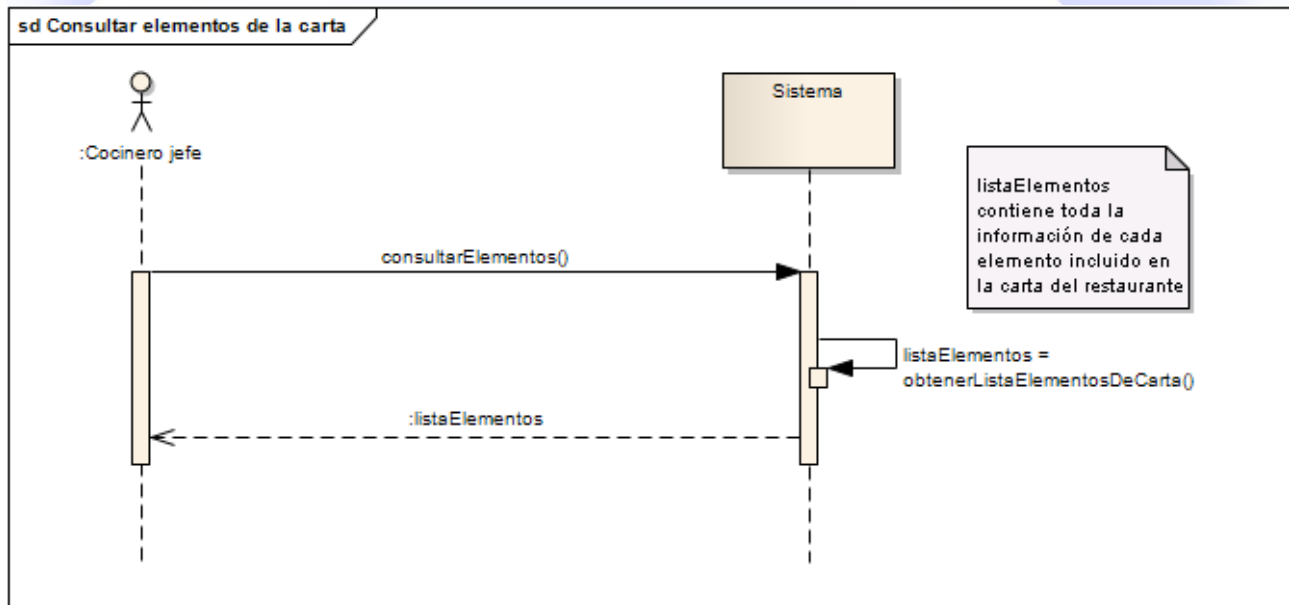


## Modificar elemento de la carta

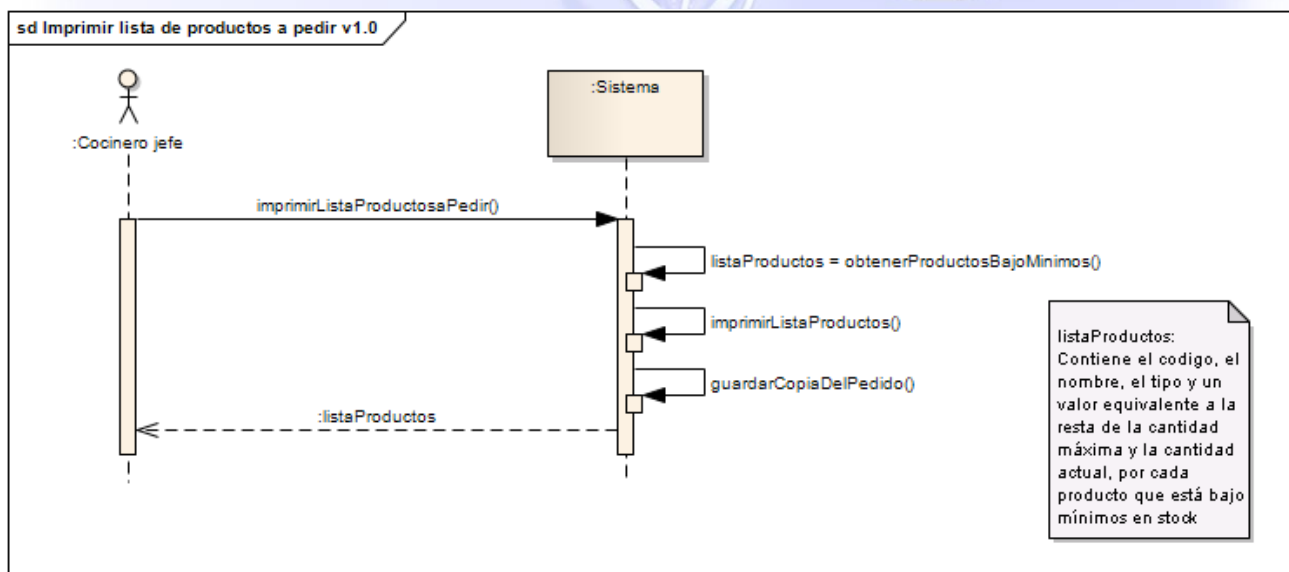




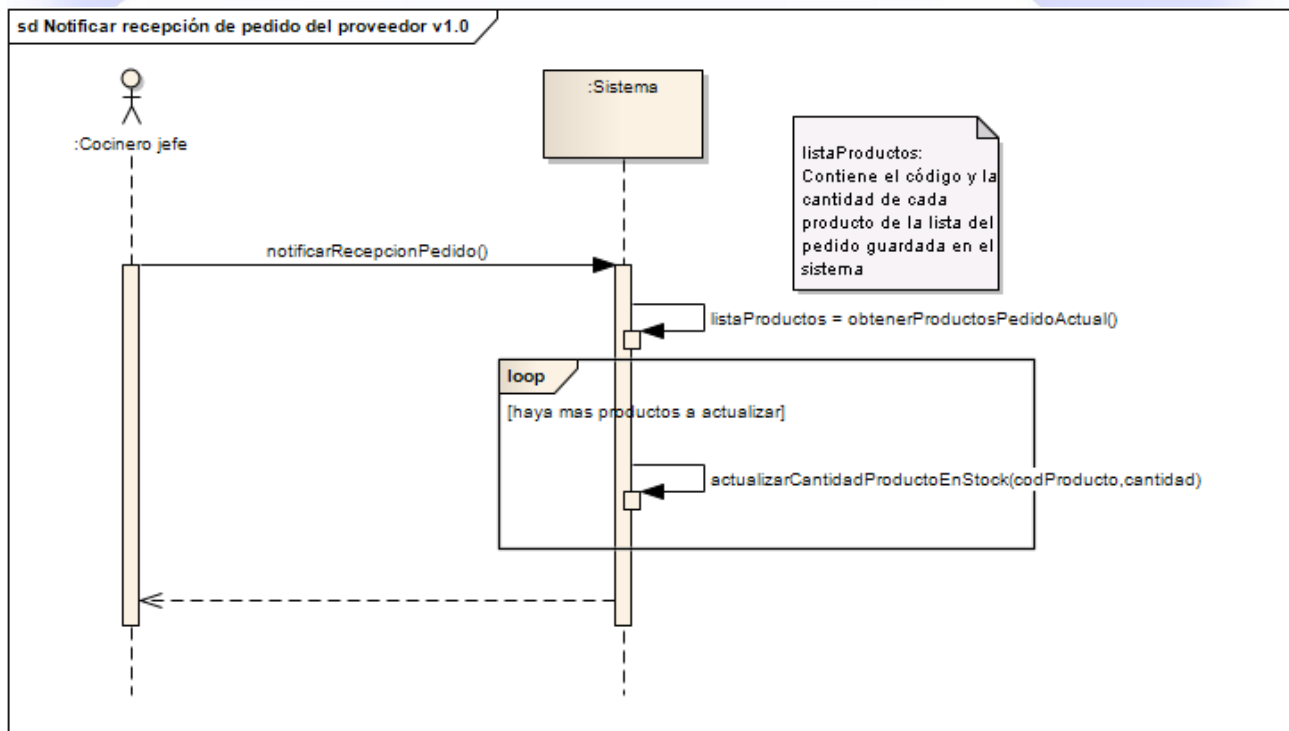
## Consultar elementos de la carta



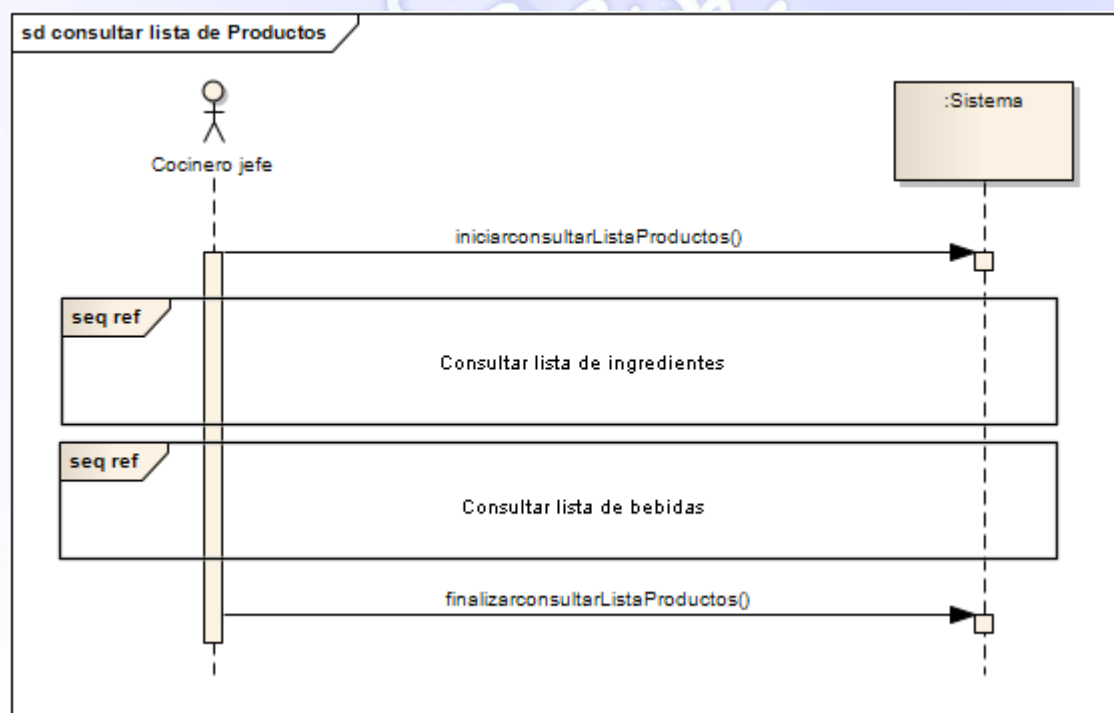
## Imprimir lista de productos a pedir



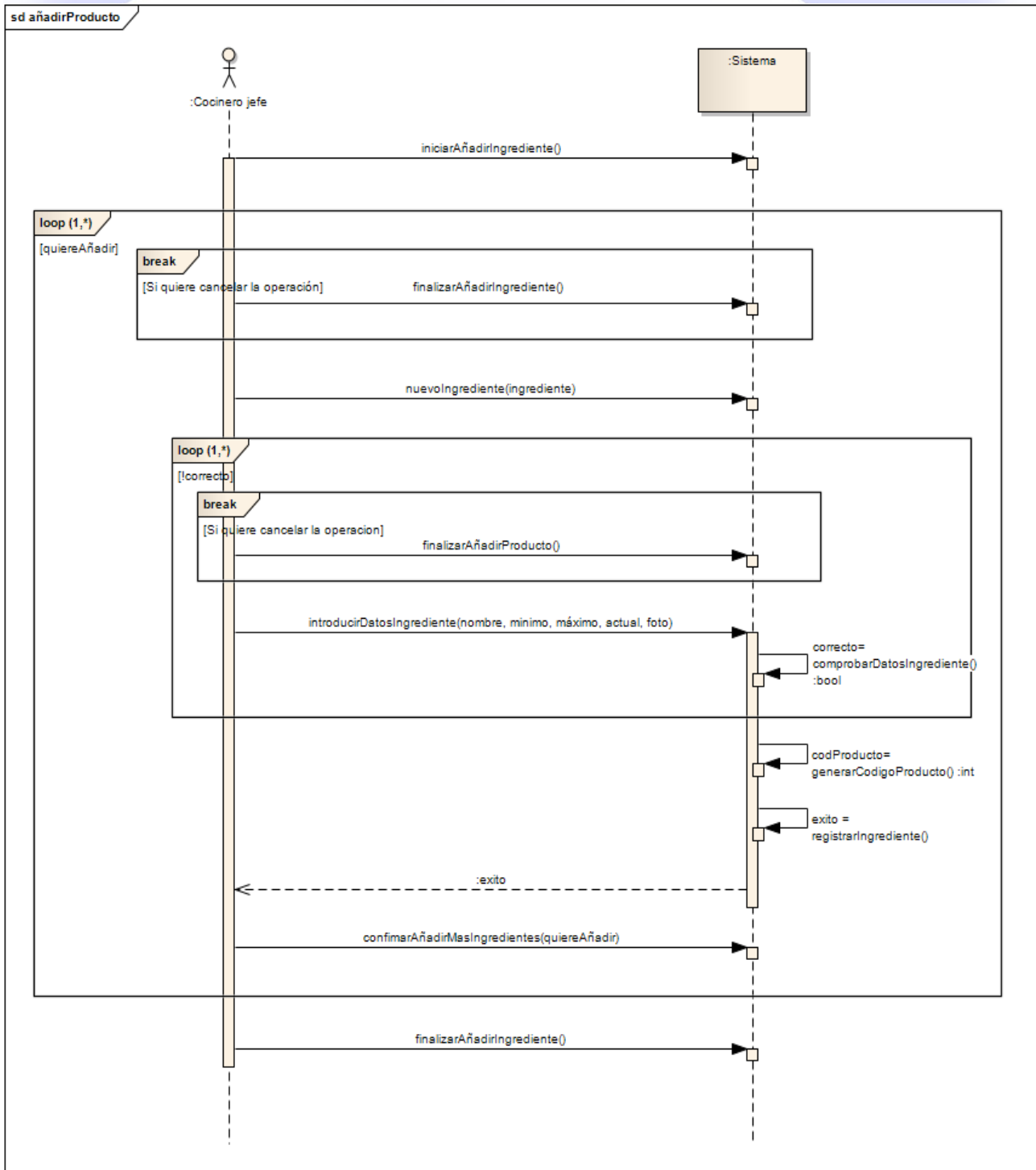
## Notificar recepción de pedido del proveedor



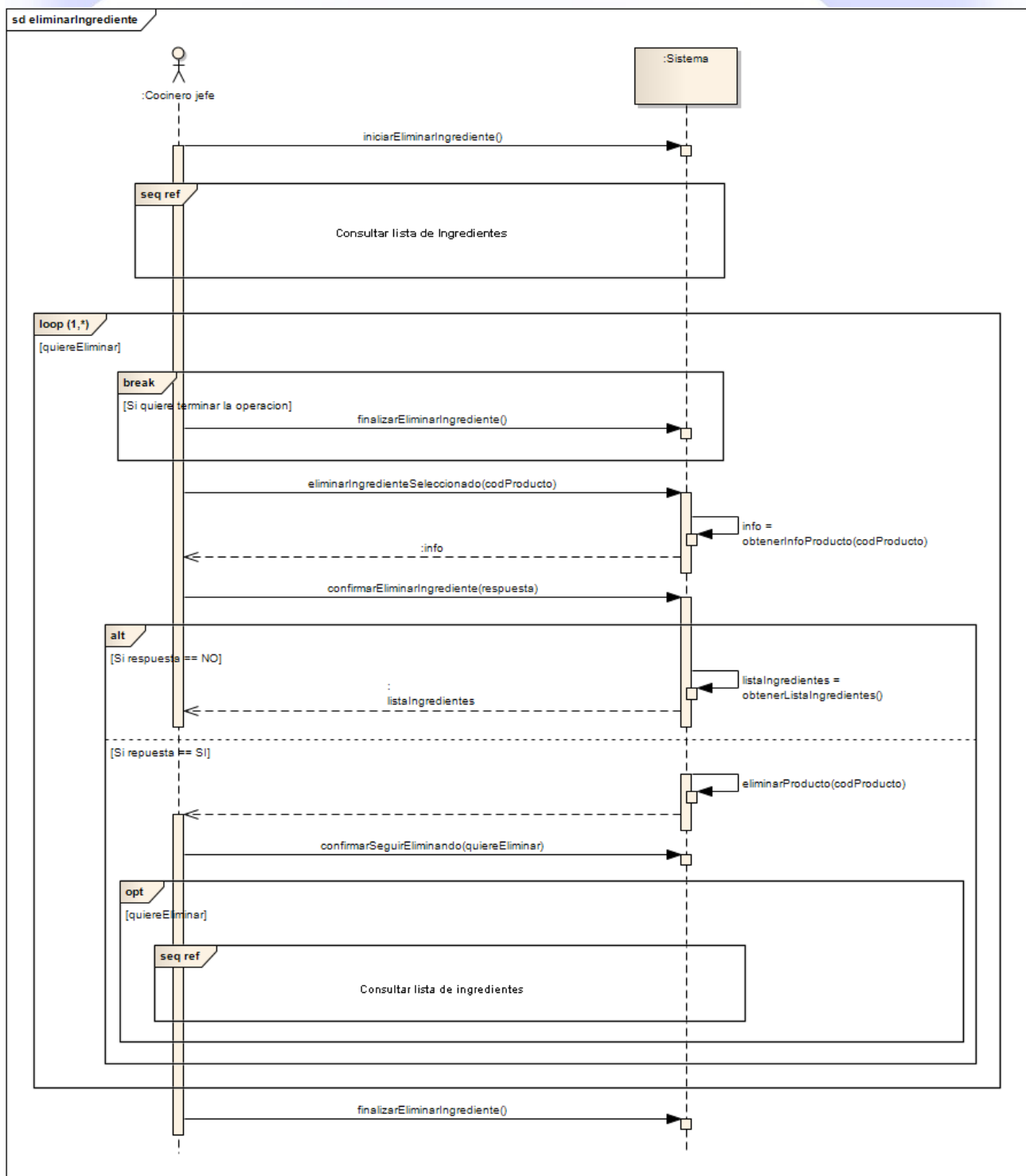
## Consultar lista de productos



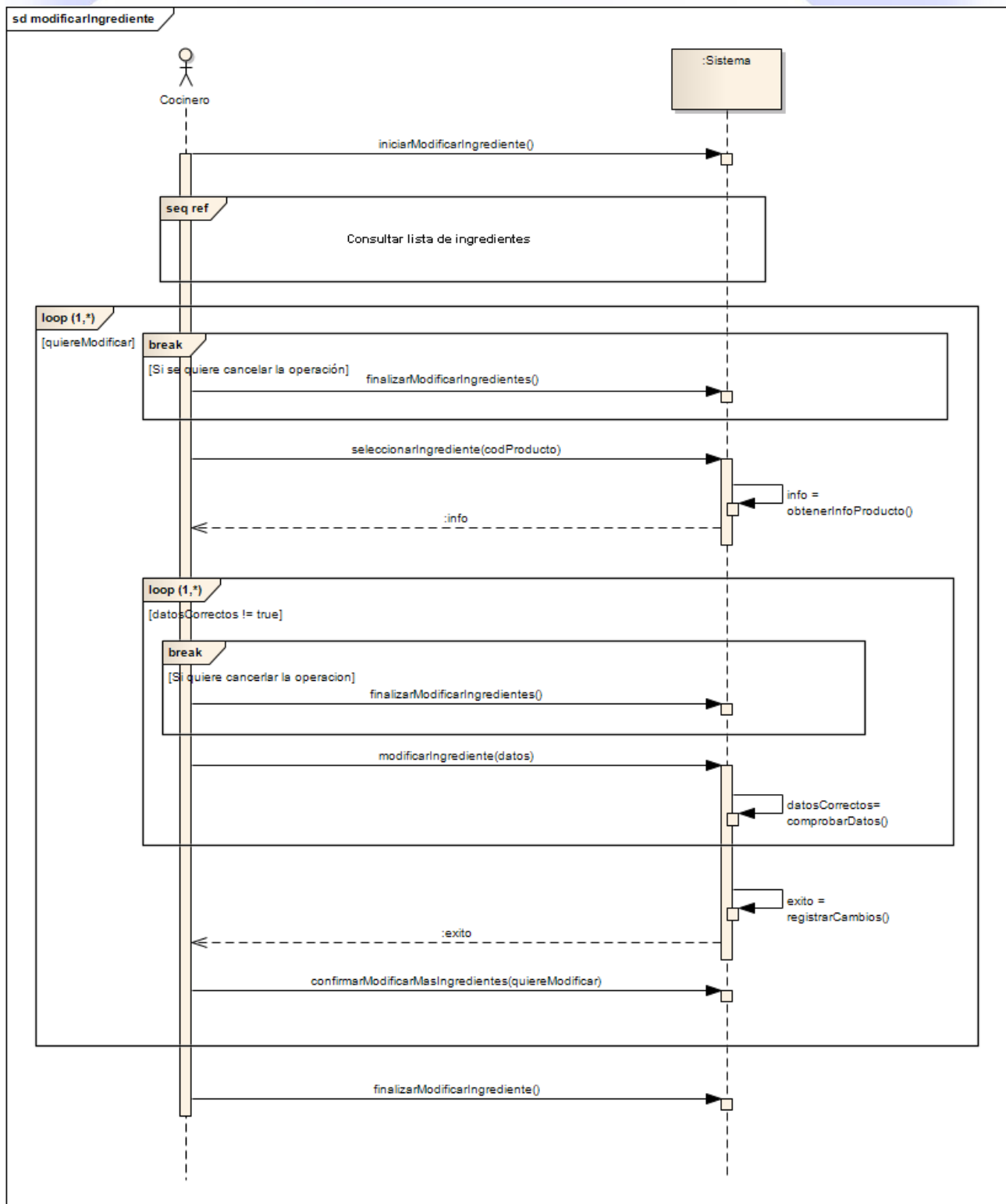
## Añadir ingrediente



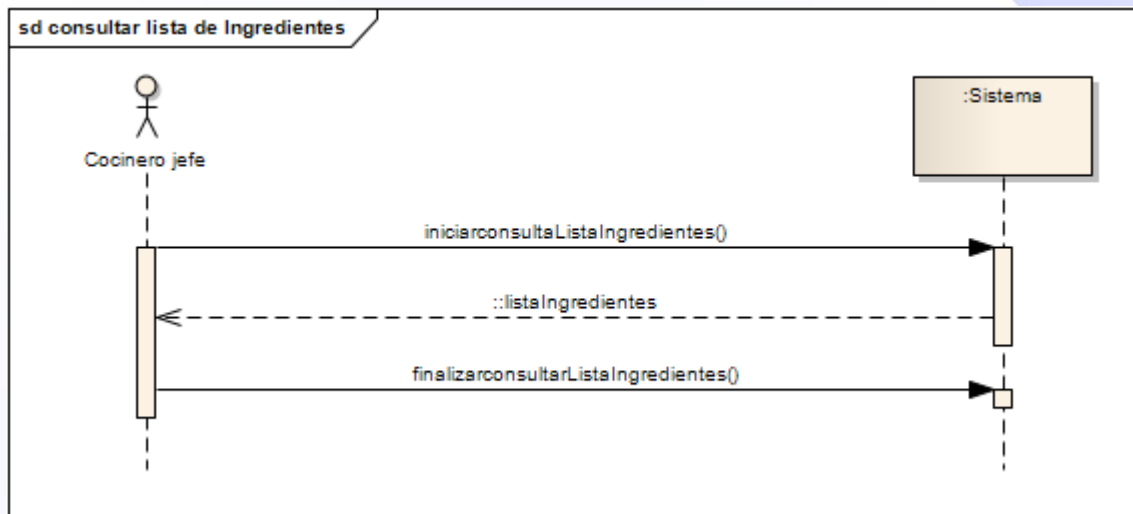
## Eliminar ingrediente



## Modificar ingrediente



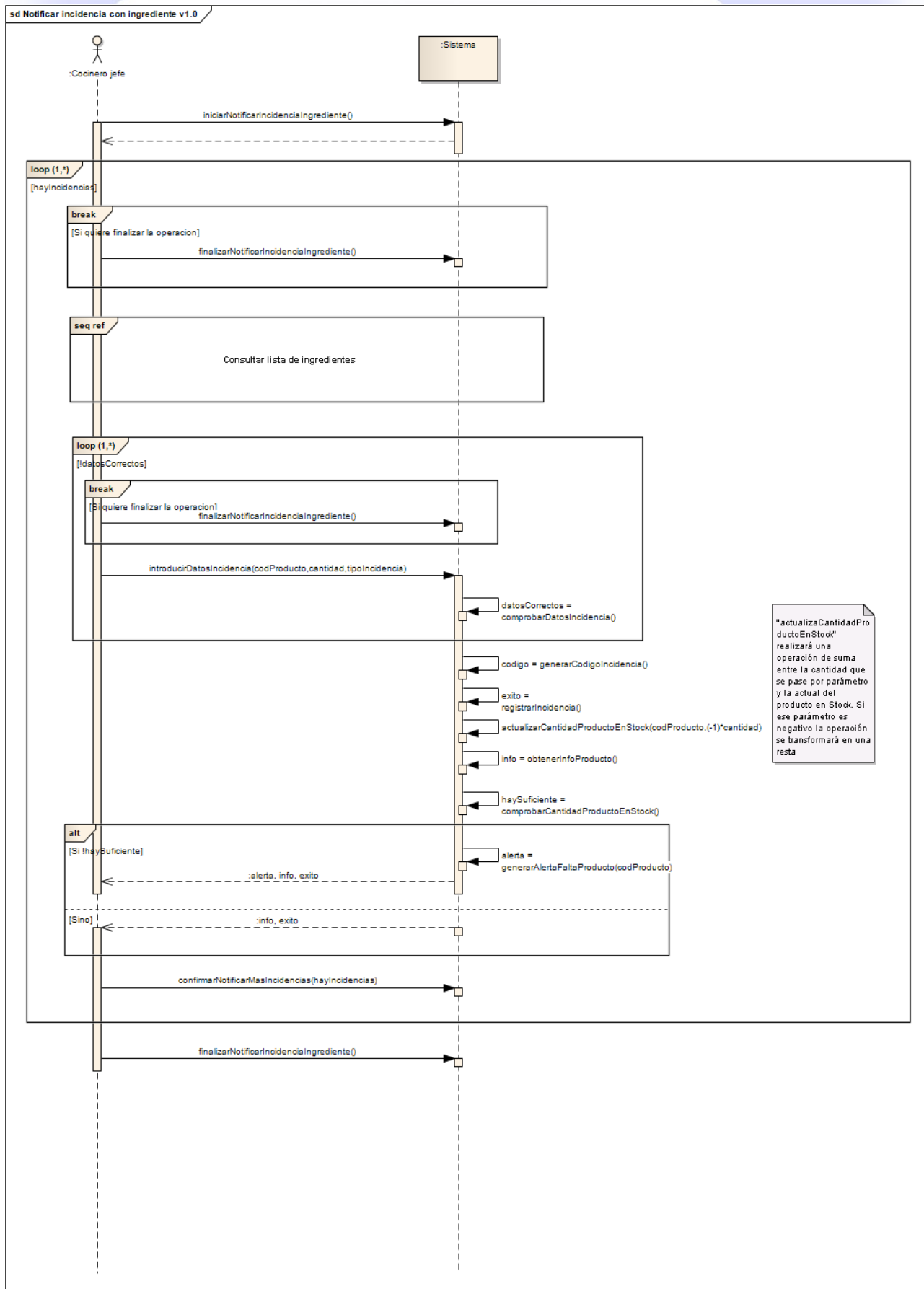
## Consultar lista de ingredientes



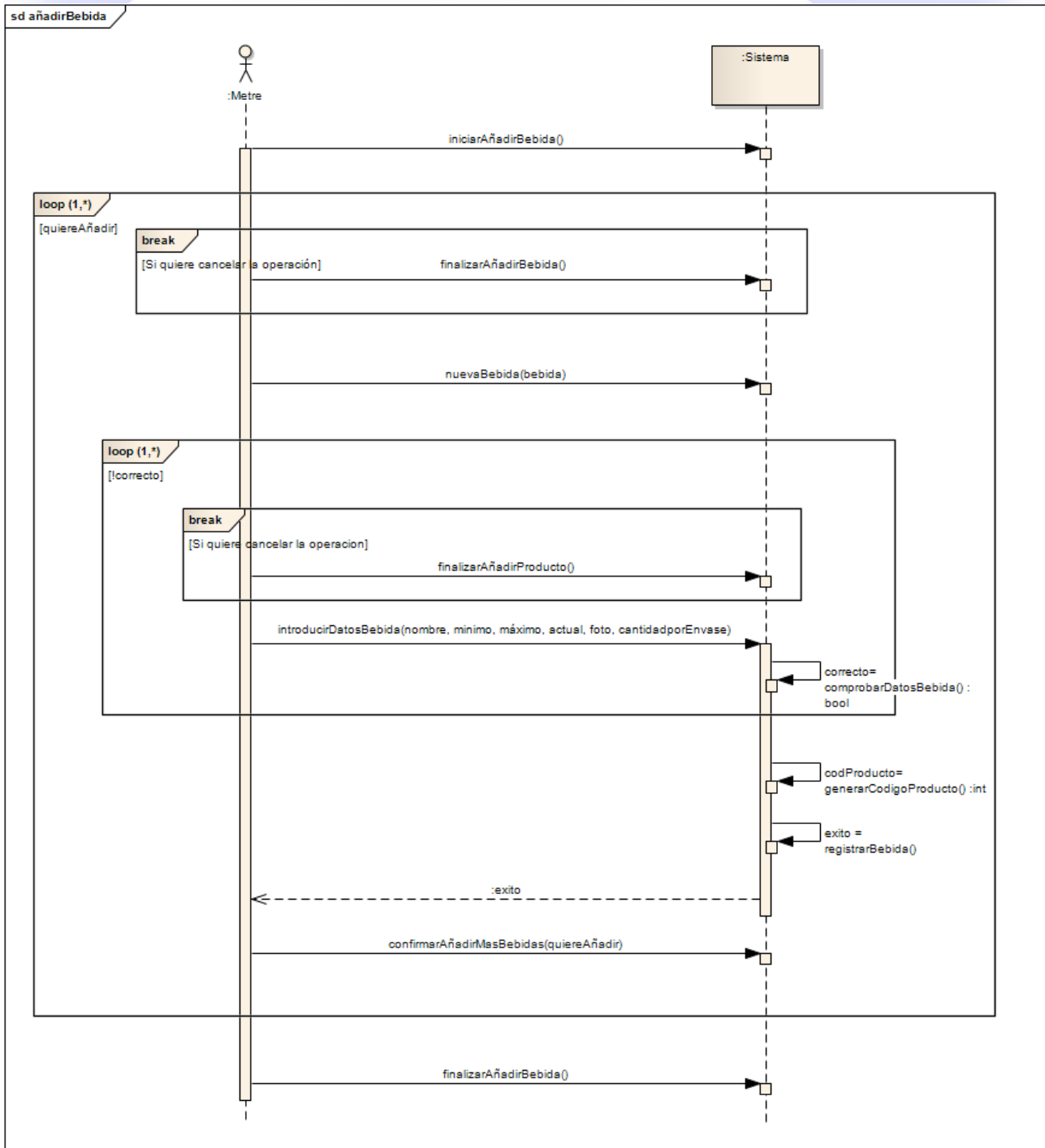
Sagres



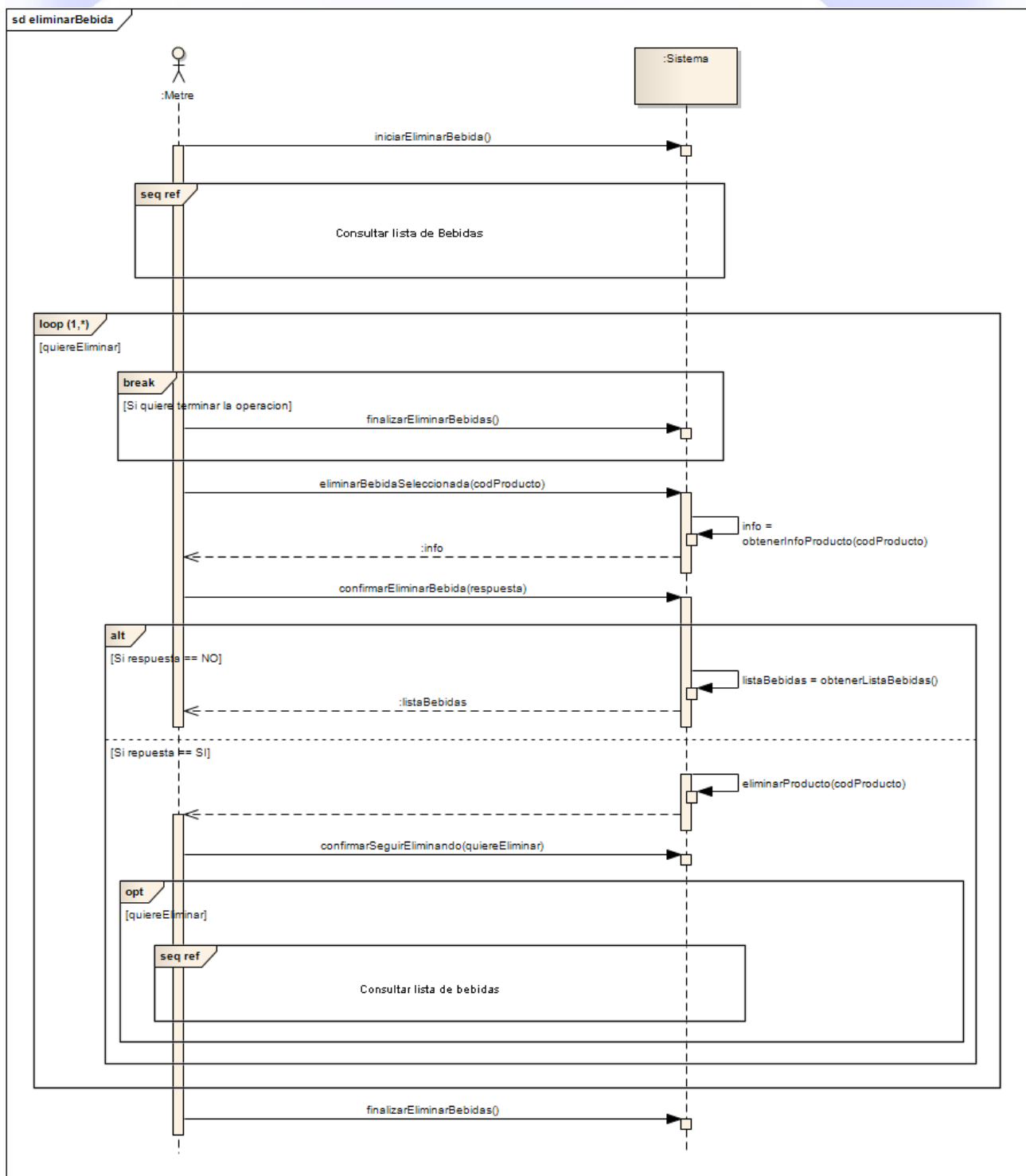
## Notificar incidencia con ingrediente



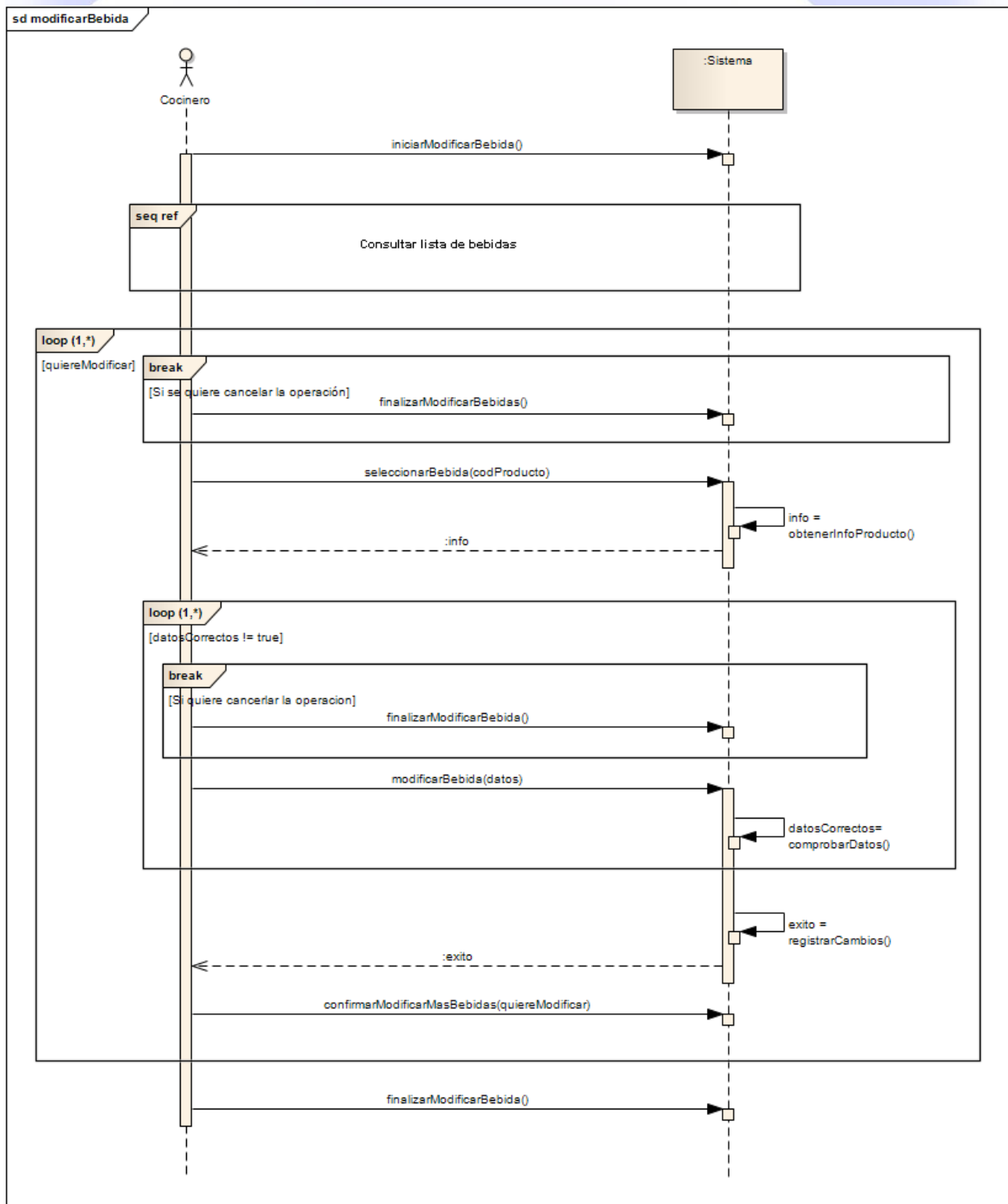
## Añadir bebida



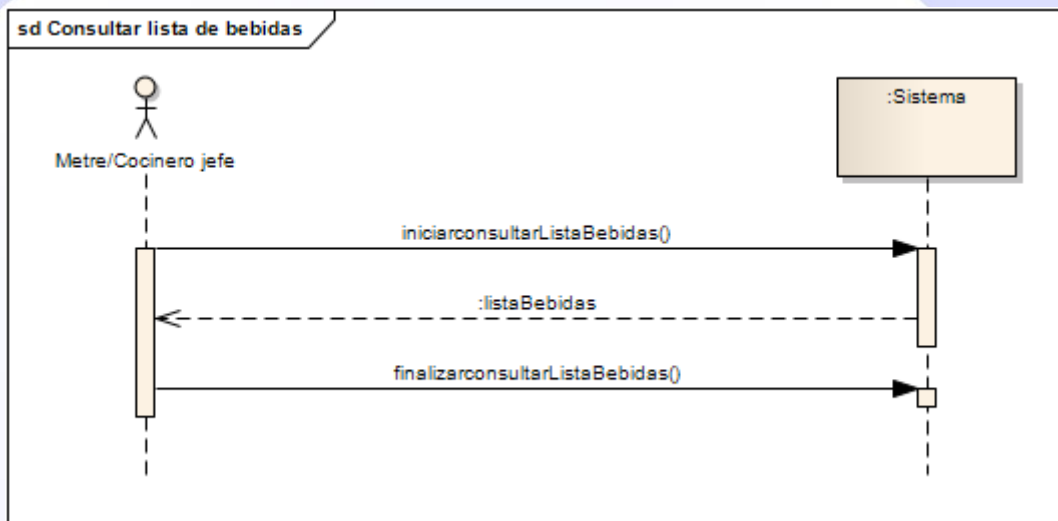
## Eliminar bebida



## Modificar bebida

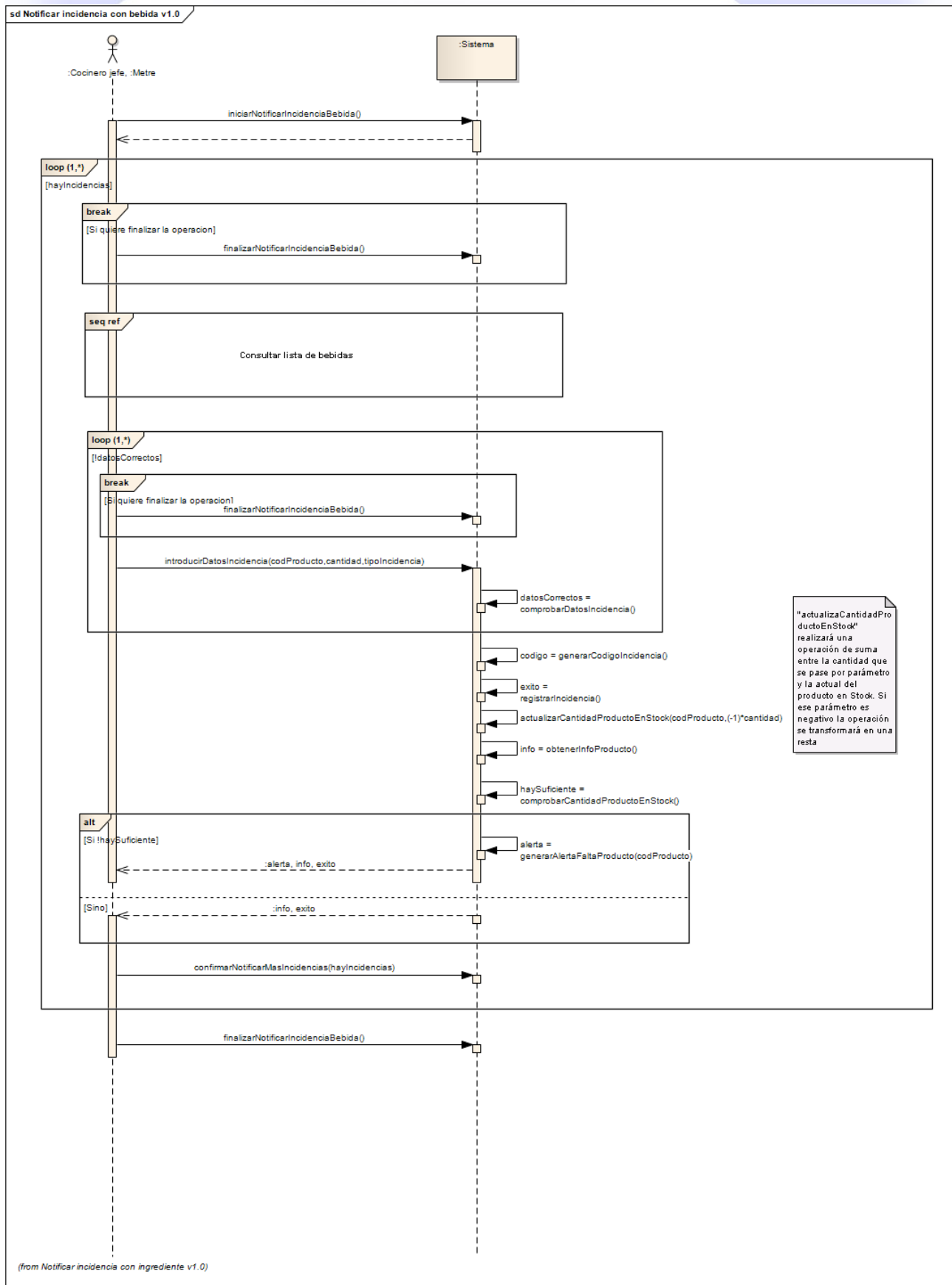


## Consultar lista de bebidas



Sagres

## Notificar incidencia con bebida





## Identificación de las operaciones del sistema

### Añadir elemento a la carta

- `iniciarAñadirElementoACarta()`: operación que inicia el caso de uso para añadir un nuevo elemento a la carta.
- `FinalizarAñadirElemento()`: operación con la que el usuario decide finalizar la operación de añadir un elemento
- `añadirIngrediente(codIng,cantidad)`: se introduce el código y la cantidad del ingrediente deseado en el sistema
- `añadirIngredienteAlista(listaIngredientes,codIng,cantidad)`: el sistema añade el código y la cantidad del ingrediente a la lista de ingredientes asociada al elemento de la carta que se va a añadir.
- `eliminarIngrediente(codIng)`: se introduce el código del ingrediente que se desea eliminar de la lista de ingredientes
- `eliminarIngredienteDeLista(listaIngredientes,codIng)`: el sistema elimina el ingrediente con código `codIng` de la lista de ingredientes asociados al elemento que se va a añadir
- `introducirNuevaCantidad(codIng,cantidad)`: se introduce el código del ingrediente del cual se desea modificar la cantidad, y la nueva cantidad.
- `modificarCantidadIngredienteDeLista(listaIngredientes,codIng,cantidad)`: el sistema modifica la cantidad del ingrediente con código `codIng` de la lista de ingredientes por el valor especificado en el parámetro `cantidad`.
- `asignarSeccion(sec)`: se introduce la sección que se le quiere asignar al nuevo elemento
- `introducirDatosElemento(nombre,descripcion,precio,foto,tiempoPreparacion,raciones)`: se introducen los datos correspondientes al nuevo elemento.
- `registrarElemento()`: el sistema registra el nuevo elemento.
- `confirmarAñadirMasElementos(quiereAñadir)`: el usuario responde al sistema acerca de si desea seguir añadiendo más elemento.

### Eliminar elemento de la carta

- `iniciarElementoDeCarta()`: operación que inicia el caso de uso para eliminar un elemento de la carta
- `finalizarEliminarElemento()`: operación con la que el usuario decide finalizar la operación de eliminar un elemento de la carta
- `eliminarElementoSeleccionado(codElemento)`: se introduce el código del elemento que se desea eliminar
- `obtenerInfoElemento(codElemento)`: el sistema obtiene toda la información asociada al elemento con código `codElemento`
- `confirmarEliminarElemento(respuesta)`: se introduce la confirmación o negación para eliminar el producto finalmente.

- `eliminarElemento(codElemento)`: el sistema elimina el elemento cuyo código es `codElemento`
- `confirmarSeguirEliminando(quiereEliminar)`: se introduce la respuesta de si se desea eliminar algún elemento mas de la carta

### **Modificar elemento de la carta**

- `iniciarModificarElementoDeCarta()`: operación que inicia el caso de uso para modificar un elemento de la carta
- `finalizarModificarElementoDeCarta()`: operación con la que el usuario decide finalizar la operación de modificar un elemento de la carta.
- `modificarElementoSeleccionado(codElemento)`: se introduce el código del elemento que se desea modificar.
- `modificarDatosElemento(datos)`: se introducen los nuevos datos del elemento. Estos datos son nombre, descripción, precio, foto, tiempo de preparación y el número de raciones en el que se puede dividir el plato.
- `ComprobarDatos()`: el sistema comprueba si los datos introducidos son correctos.
- `registrarDatos()`: el sistema registra los nuevos datos del elemento.
- `confirmarModificarMasElementos(quiereModificar)`: se introduce la respuesta de si se desea modificar algún elemento mas de la carta

### **Consultar elementos de la carta**

- `consultarElementos()`: operación que inicia el caso de uso para consultar los elementos de la carta registrados.
- `ObtenerListaElementosDeCarta()`: el sistema obtiene una lista con los elementos de la carta registrados

### **Imprimir lista de productos a pedir**

- `imprimirListaProductosAPedir()`: operación que inicia el caso de uso para imprimir la lista de productos a pedir
- `obtenerProductosBajoMinimos()`: el sistema obtiene una lista de los productos cuya cantidad actual en stock está por debajo de su límite
- `imprimirListaProductos()`: el sistema manda un mensaje a la impresora con la información que debe imprimir. En este caso imprimirá la lista de productos almacenada
- `guardarCopiaDelPedido()`: el sistema almacena una copia de la información del pedido

### **Notificar recepción de pedido del proveedor**

- `notificarRecepcionPedido()`: operación que inicia el caso de uso para notificar al sistema que se ha recibido el pedido del proveedor

- `obtenerProductosPedidoActual()`: el sistema recupera la información asociada al pedido realizado
- `actualizarCantidadProductoEnStock(codProducto,cantidad)`: el sistema actualiza la cantidad del producto identificado por `codProducto`. La manera de actualizar la cantidad sera realizando una suma entre la cantidad actual del producto y la cantidad pasada por parámetro Si se pasara un valor de cantidad negativo por parámetro, esto operación de suma se transformaría en una resta.

### **Notificar incidencia con ingrediente**

- `iniciarNotificarIncidenciaIngrediente()`: operación que inicial el caso de uso para notificar una incidencia con un ingrediente
- `finalizarNotificarIncidenciaIngrediente()`: operación con la que el usuario decide finalizar la operación de notificar una incidencia con un ingrediente.
- `introducirDatosIncidencia(codProducto,cantidad,tipoIncidencia)`: se introducen los datos de la incidencia. En este caso los datos son el código de producto, la cantidad de producto implicado y el tipo de incidencia.
- `ComprobarDatosIncidencia()`: el sistema comprueba que los datos introducidos sean correctos.
- `GenerarCodigoIncidencia()`: el sistema genera un código para la incidencia
- `registrarIncidencia()`: el sistema registra la incidencia
- `comprobarCantidadProductoEnStock()`: el sistema comprueba si la cantidad del producto actual esta por debajo o por encima del limite
- `generarAlertaFaltaProducto(codProducto)`: el sistema genera un mensaje de alerta informando al usuario de que la cantidad actual en stock del producto identificado por `codProducto` ha superado su límite mínimo.
- `confirmarNotificarMasIncidencias(hayIncidencias)`: se introduce la respuesta de si se desea notificar alguna incidencia mas

### **Consultar lista de productos**

- `iniciarconsultaListaProductos()`: Operación que inicia el caso de uso para consultar la lista de productos.
- `FinalizarconsultarListaProductos()`: Operación que finaliza el caso de uso para consultar la lista de productos.

### **Añadir ingrediente**

- `iniciarAñadirIngrediente()`: Operación que inicia el caso de uso para añadir un nuevo ingrediente.
- `nuevoIngrediente(ingrediente)`: Selección del nuevo ingrediente.
- `introducirDatosIngrediente(nombre, mínimo, máximo, actual, foto)`: se introduce los datos necesarios para añadir un nuevo ingrediente, tales como nombre, mínimo, máximo, nivel actual y foto.

- **ComprobarDatosIngrediente():** Se comprueba que los datos que el usuario ha introducido son correctos, es decir, entran dentro de las limitaciones, son números 'lógicos'...
- **generarCódigoProducto():** El sistema asigna un código de producto al nuevo ingrediente. Será consecutivo al último que hay.
- **registrarIngrediente():** Se guarda toda la información correspondiente al nuevo ingrediente en el sistema.
- **confirmarAñadirMasIngredientes(quiereAñadir):** el usuario responde al sistema acerca de si desea seguir añadiendo nuevos ingredientes.
- **finalizarAñadirIngrediente():** Operación que finaliza el caso de uso de añadir un nuevo ingrediente.

### **Eliminar ingrediente**

- **iniciarEliminarIngredientes():** Operación que inicia el caso de uso para eliminar un ingrediente.
- **eliminarIngredienteSeleccionado(codProducto):** El usuario selecciona el ingrediente que quiere eliminar con código codProducto.
- **obtenerInfoProducto():** obtiene información del producto requerido y se lo devuelve al usuario.
- **confirmarEliminarIngrediente(respuesta):** el usuario confirma la eliminación del ingrediente seleccionado.
- **ObtenerListaIngredientes():** obtiene una lista de los ingredientes registrados en el sistema.
- **eliminarProducto(codProducto):** elimina definitivamente del sistema el producto con el código codProducto.
- **confirmarSeguirEliminando(quiereEliminar):** el usuario responde al sistema acerca de si desea seguir eliminando ingredientes.
- **finalizarEliminarIngrediente():** Operación que finaliza el caso de uso para eliminar un ingrediente.

### **Modificar ingrediente**

- **iniciarModificarIngrediente():** Operación que inicia el caso de uso para modificar un ingrediente.
- **seleccionarIngrediente(codProducto):** El usuario selecciona el ingrediente con el código codProducto.
- **obtenerInfoProducto():** obtiene información del producto requerido y se lo devuelve al usuario.
- **modificarIngrediente(datos):** el usuario modifica los datos del ingrediente. Datos son los datos una vez modificados, es decir, los datos nuevos.
- **comprobarDatos():** Se comprueba que los datos que el usuario ha introducido son correctos, es decir, entran dentro de las limitaciones, son números 'lógicos'...
- **registrarCambios():** El sistema guarda definitivamente los cambios en los datos del ingrediente.



- `confirmarModificarMasIngredientes(quiereModificar)`: el usuario responde al sistema acerca de si desea seguir modificando ingredientes.
- `finalizarModificarIngrediente()`: Operación que finaliza el caso de uso para modificar un ingrediente.

### **Consultar lista de ingredientes**

- `iniciarconsultaListaIngredientes()`: Operación que inicia el caso de uso para consultar la lista de ingredientes.
- `listaIngredientes`: Una lista de ingredientes que el sistema genera y muestra al usuario.
- `finalizarconsultarListaIngredientes()`: Operación que finaliza el caso de uso para consultar la lista de ingredientes.

### **Añadir bebida**

- `iniciarAñadirBebida()`: Operación que inicia el caso de uso para añadir una nueva bebida.
- `nuevaBebida(bebida)`: Selección de la nueva bebida.
- `introducirDatosBebida(nombre, minimo, máximo, actual, foto, cantidadPorEnvase)`: se introduce los datos necesarios para añadir un nuevo ingrediente, tales como nombre, mínimo, máximo, nivel actual, foto y cantidad por Envase.
- `comprobarDatosBebida()`: Se comprueba que los datos que el usuario ha introducido son correctos, es decir, entran dentro de las limitaciones, son números 'lógicos'...
- `generarCodigoProducto()`: El sistema asigna un código de producto a la nueva bebida. Será consecutivo al último que hay.
- `registrarBebida()`: Se guarda toda la información correspondiente a la nueva bebida en el sistema.
- `confirmarAñadirMasBebidas(quiereAñadir)`: el usuario responde al sistema acerca de si desea seguir añadiendo nuevas bebidas.
- `finalizarAñadirBebida()`: Operación que finaliza el caso de uso de añadir una nueva bebida.

### **Eliminar bebida**

- `iniciarEliminarBebida()`: Operación que inicia el caso de uso para eliminar una bebida.
- `eliminarBebidaSeleccionada(codProducto)`: El usuario selecciona la bebida con el código `codProducto`.
- `ObtenerInfoProducto()`: obtiene información del producto requerido y se lo devuelve al usuario.
- `confirmarEliminarBebida(respuesta)`: el usuario confirma la eliminación de la bebida seleccionada.
- `ObtenerListaBebidas()`: obtiene una lista de bebidas registrados en el sistema.
- `eliminarProducto(codProducto)`: elimina definitivamente del sistema el producto con el código `codProducto`.
- `confirmarSeguirEliminando(quiereEliminar)`: el usuario responde al sistema acerca de si desea seguir eliminando bebidas.

- finalizarconsultarListaBebidas(): Operación que finaliza el caso de uso para eliminar una bebida.

### **Modificar bebida**

- iniciarModificarBebida(): Operación que inicia el caso de uso para modificar una bebida.
- seleccionarBebida(codProducto): El usuario selecciona la bebida con el código codProducto.
- ObtenerInfoProducto(): obtiene información del producto requerido y se lo devuelve al usuario.
- modificarBebida(datos): el usuario modifica los datos de la bebida. Datos son los datos una vez modificados, es decir, los datos nuevos.
- comprobarDatos(): Se comprueba que los datos que el usuario ha introducido son correctos, es decir, entran dentro de las limitaciones, son números 'lógicos'...
- registrarCambios(): El sistema guarda definitivamente los cambios en los datos del ingrediente.
- confirmarModificarMasBebidas(quiereModificar): el usuario responde al sistema acerca de si desea seguir modificando nuevas bebidas.
- finalizarModificarBebida(): Operación que finaliza el caso de uso para modificar una bebida.

### **Consultar lista de bebidas**

- iniciarconsultaListaBebidas(): Operación que inicia el caso de uso para consultar la lista de bebidas.
- :listaBebidas: Una lista de bebidas que el sistema genera y muestra al usuario.
- finalizarconsultarListaBebidas(): Operación que finaliza el caso de uso para consultar la lista de bebidas.

### **Notificar incidencia con bebida**

- iniciarNotificarIncidenciaBebida(): Operación que inicia el caso de uso para notificar una incidencia con una bebida.
- introducirDatosIncidencia(codProducto, cantidad, tipoIncidencia): se introducen los datos de la incidencia. En este caso los datos son el código de producto, la cantidad de producto implicado y el tipo de incidencia.
- ComprobarDatosIncidencia(): el sistema comprueba que los datos introducidos sean correctos.
- GeneraCodigoIncidencia(): el sistema genera un código para la incidencia.
- RegistraIncidencia(): el sistema registra la incidencia.
- comprobarCantidadProductoEnStock(): el sistema comprueba si la cantidad del producto actual esta por debajo o por encima del limite.



- `generarAlertaFaltaDeProducto(codProducto)`: el sistema genera un mensaje de alerta informando al usuario de que la cantidad actual en stock del producto identificado por `codProducto` ha superado su límite mínimo.
- `confirmarNotificaMasIncidencias(hayIncidencias)`: se introduce la respuesta de si se desea notificar alguna incidencia más.
- `finalizarNotificarIncidenciaBebida()`: Operación que finaliza el caso de uso para notificar una incidencia con una bebida.



## APÉNDICE 1.0

Fecha	15/03/10
Descripción del problema	-
Impacto del problema	-
Soluciones adoptadas	<ul style="list-style-type: none"><li>• Se genero el documento de modelado de requisitos inicial</li></ul>
Anexos a la versión	

Sagres