

# Documento de PLANIFICACIÓN



**Sistema de Administración y  
Gestión de RESTaurantes**



Gaspar Muñoz Soria  
Samuel Guirado Navarro  
Carlos Moreno Muñoz

V1.3

## ÍNDICE DE CONTENIDO

Apartado de control de versiones.....	4
1. Estilo arquitectónico.....	5
2. Diagrama de clases de diseño.....	6
3. Diagrama de paquetes estructurales.....	8
4. Diagrama de componentes.....	10
<i>Controlador Impresora.....</i>	<i>12</i>
<i>Gestión de base de datos.....</i>	<i>13</i>
<i>Controlador principal.....</i>	<i>14</i>
<i>Gestión Carta.....</i>	<i>15</i>
<i>Gestión Incidencias.....</i>	<i>16</i>
<i>Gestión Pedidos Proveedor.....</i>	<i>17</i>
<i>Gestión Productos.....</i>	<i>18</i>
5. Diagramas de colaboración de diseño.....	19
<i>compruebaElementosInvalidados.....</i>	<i>19</i>
<i>eliminaElemento.....</i>	<i>20</i>
<i>eliminaProducto.....</i>	<i>21</i>
<i>modificaProducto.....</i>	<i>22</i>
<i>modificaElemento.....</i>	<i>23</i>
<i>imprimeListaProductosaPedir.....</i>	<i>24</i>
<i>notificaRecepcionPedidoProveedor.....</i>	<i>25</i>
<i>nuevaIncidencia.....</i>	<i>26</i>
<i>nuevoElemento.....</i>	<i>27</i>
<i>nuevoProducto.....</i>	<i>28</i>
<i>obtieneElementosConProducto.....</i>	<i>29</i>
<i>obtieneBebidas.....</i>	<i>30</i>
<i>obtieneIngredientes.....</i>	<i>31</i>
<i>obtieneProductosBajoMinimos.....</i>	<i>32</i>
<i>obtieneProductosSeccion.....</i>	<i>33</i>
<i>obtieneSecciones.....</i>	<i>34</i>
6. Diagrama relacional.....	35

7. Diseño de la interfaz de usuario.....	36
<i>Consideraciones generales de nuestra interfaz.....</i>	<i>36</i>
<i>Interfaz para el cocinero jefe.....</i>	<i>38</i>
<i>Menú principal.....</i>	<i>38</i>
<i>Interfaz para la gestión de ingredientes.....</i>	<i>39</i>
<i>Añadir nuevo ingrediente.....</i>	<i>40</i>
<i>Modificar ingrediente.....</i>	<i>41</i>
<i>Eliminar ingrediente.....</i>	<i>43</i>
<i>Notificar incidencia con ingrediente.....</i>	<i>44</i>
<i>Interfaz para la gestión de la carta.....</i>	<i>46</i>
<i>Añadir nuevo elemento a carta.....</i>	<i>47</i>
<i>Modificar elemento de carta.....</i>	<i>50</i>
<i>Eliminar elemento de carta.....</i>	<i>53</i>
<i>Imprimir lista de productos a pedir.....</i>	<i>54</i>
<i>Notificar recepción de productos.....</i>	<i>55</i>
<i>Interfaz para el metre.....</i>	<i>56</i>
<i>Menú principal.....</i>	<i>56</i>
<i>Añadir nueva bebida.....</i>	<i>57</i>
<i>Modificar bebida.....</i>	<i>58</i>
<i>Eliminar bebida.....</i>	<i>60</i>
<i>Notificar incidencia con bebida.....</i>	<i>61</i>
Apéndice 1.0.....	63
Apéndice 1.1.....	64
Apéndice 1.2.....	65
Apéndice 1.3.....	66

## 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>
1.0	29-03-2010	Versión inicial del documento.
1.1	31-03-2010	<ul style="list-style-type: none"><li>• Cambios en operaciones de componentes.</li><li>• Cambios en diagramas de colaboración de diseño</li></ul>
1.2	05-04-2010	Se han producido cambios en algunos diagramas de colaboración y de componentes motivados por errores encontrados por el equipo de implementación
1.3	18-04-2010	Revisión del documento de diseño final de la 1ª iteración

## 1. ESTILO ARQUITECTÓNICO

---

Para el diseño del Sistema de Administración y Gestión de REStaurantes hemos optado por un estilo como el **MVC** o **Modelo-Vista-Controlador**.

Primero veremos los inconvenientes de algunos estilos arquitectónicos y luego las razones de los estilos que utilizaremos..

Utilizar una arquitectura multicapa no es lo más correcto dado que la aplicación no requiere de un nivel muy profundo de comunicaciones y por tanto se reduce el nivel de acoplamiento necesario en este estilo arquitectónico, y no requiere descomponer el subsistema en diversas capas organizadas jerárquicamente.

Un estilo par a par no tiene sentido que sea utilizado para el diseño de esta aplicación ya que está perfectamente definido cuales son los clientes y cuales son los servicios requeridos, no existiendo posibilidad de intercambios entre ellos.

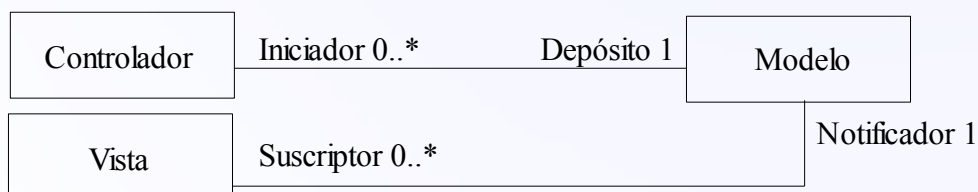
Otra opción sería realizar un diseño a partir de una arquitectura de repositorio, en el cual todos los datos están contenidos en un espacio compartido. Pero dicho estilo es útil si gestiona una gran cantidad de datos y los datos que manejará nuestro sistema de gestión de un restaurante, a priori, no son de dimensiones excesivamente grandes, por tanto obtendríamos un nivel mayor de acoplamiento que con otros modelos.

A continuación detallaremos las soluciones adoptadas.

Utilizaremos una arquitectura basada en capas cerradas que separe la aplicación en tres niveles: interfaz de usuario, lógica de aplicación y servicios. Esto mejora la estructura del sistema y fomenta la flexibilidad en cuanto a sustitución de servicios o interfaz de usuario.

Por otra parte se usará una arquitectura MVC (Modelo-Vista-Controlador) para desacoplar la lógica de la aplicación del interfaz de usuario. Esto facilita el diseño de distintas interfaces de usuarios adaptados a los distintos terminales, una para los clientes, otra para la cocina y otra para la barra.

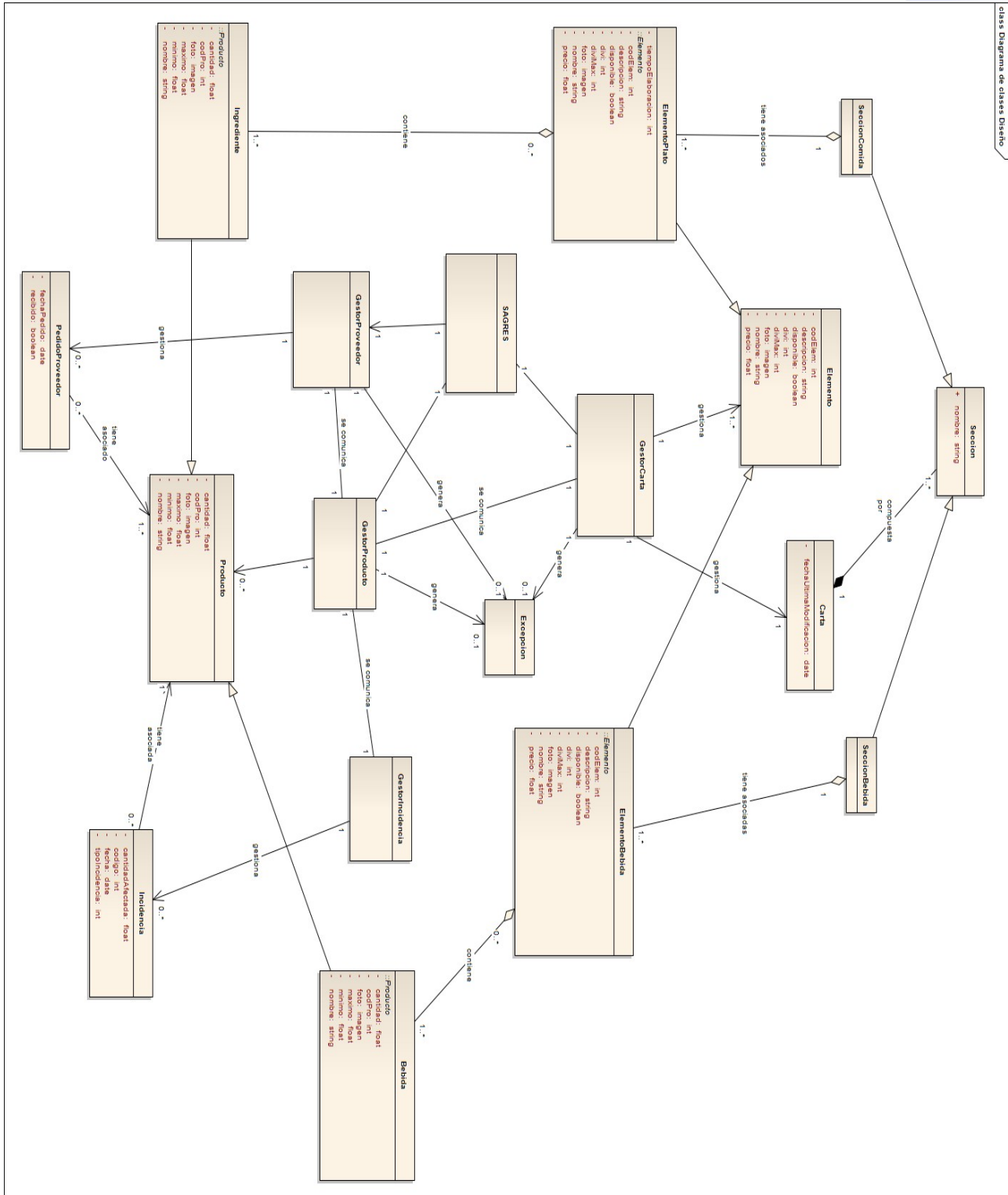
Con esto obtenemos un nivel alto de cohesión y un nivel bajo de acoplamiento separando los datos de todo lo demás, ya que cambios en nuestro diagrama de clases no repercutirán en cambios en la interfaz de usuario y viceversa.





## 2. DIAGRAMA DE CLASES DE DISEÑO

Ahora veremos un diagrama estático que describe la estructura de SAGRES mostrando sus clases, atributos y las relaciones entre ellos, es decir, orientado al diseño de la aplicación. Contiene las definiciones de las entidades del software en vez de conceptos del mundo real.



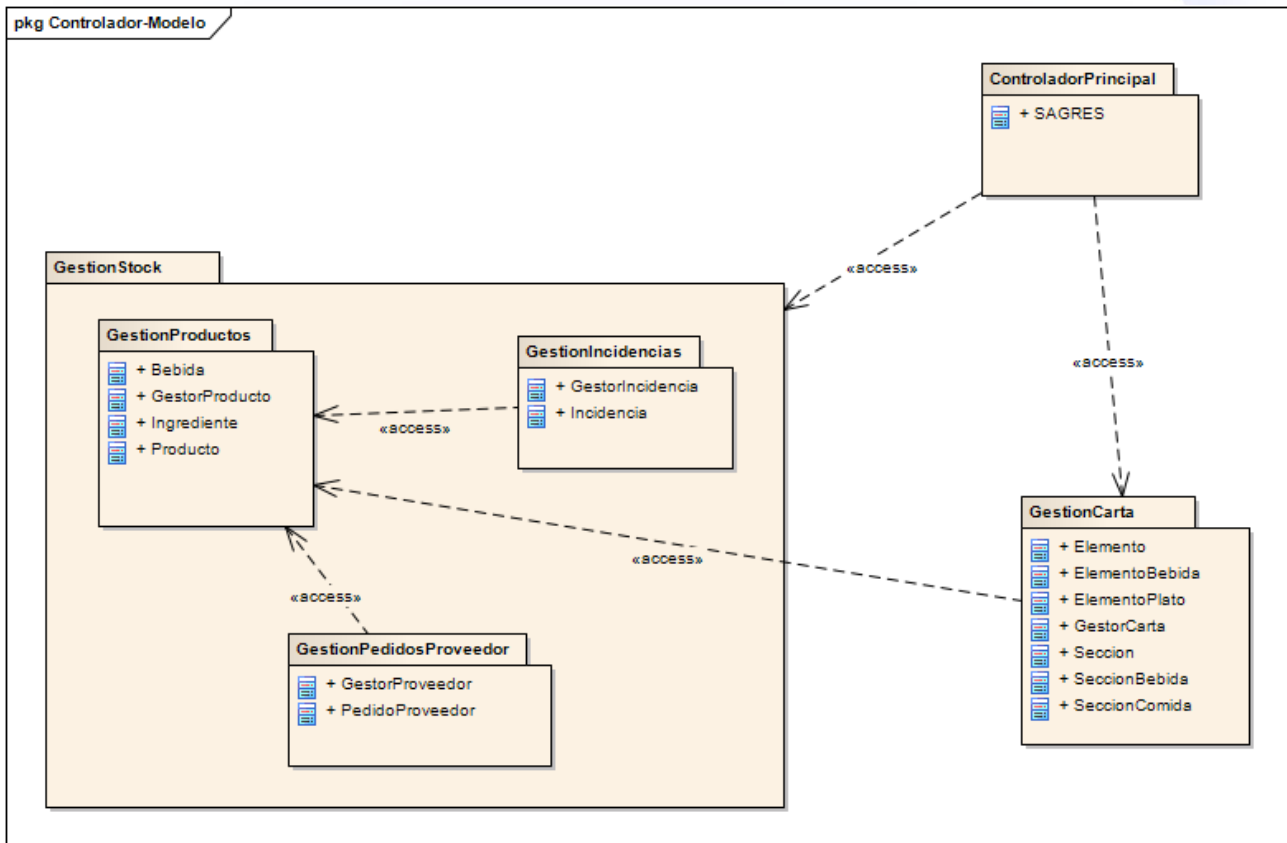
Vamos a comentar las cosas más importantes del diagrama de clases orientado al diseño.

1. El sistema como veremos en las siguientes páginas se divide en diferentes paquetes estructurales. Para la funcionalidad de cada uno de ellos se ha definido un controlador. Estos controladores son 4:
  - GestorProducto controlará todo lo relativo al stock de productos registrados en el sistema.
  - GestorCarta: controlará todo lo relativo a la gestión de la carta del restaurante.
  - GestorProveedor: controlará todo lo relativo a pedidos a proveedores.
  - GestorIncidencia: controlará todo lo relativo a las distintas incidencias que pueden sufrir los productos del stock.
2. Además, como se puede observar, se ha definido una clase controladora principal denominada SAGRES que servirá como interacción entre la capa de la interfaz de usuario y cada uno de los gestores dentro de la capa de nuestro modelado estático.
3. Vemos como tenemos tres generalizaciones en las que utilizaremos la herencia, que tanto nos facilita el trabajo a la hora de diseñar. Tendremos una clase Producto de la que heredan las clases Ingrediente y Bebida. Por otro lado tendremos la clase Elemento, de la que heredan las clases ElementoPlato y ElementoBebida, y finalmente la clase Sección de la que heredan Sección Comida y Sección Bebida.

Sagres

### 3. DIAGRAMA DE PAQUETES ESTRUCTURALES

Una vez realizado el diagrama de clases orientado al diseño, tenemos que identificar las agrupaciones lógicas en torno a nuestro sistema, quedando de una manera global así, una vez eliminadas las dependencias cíclicas que se daban en un diagrama inicial.



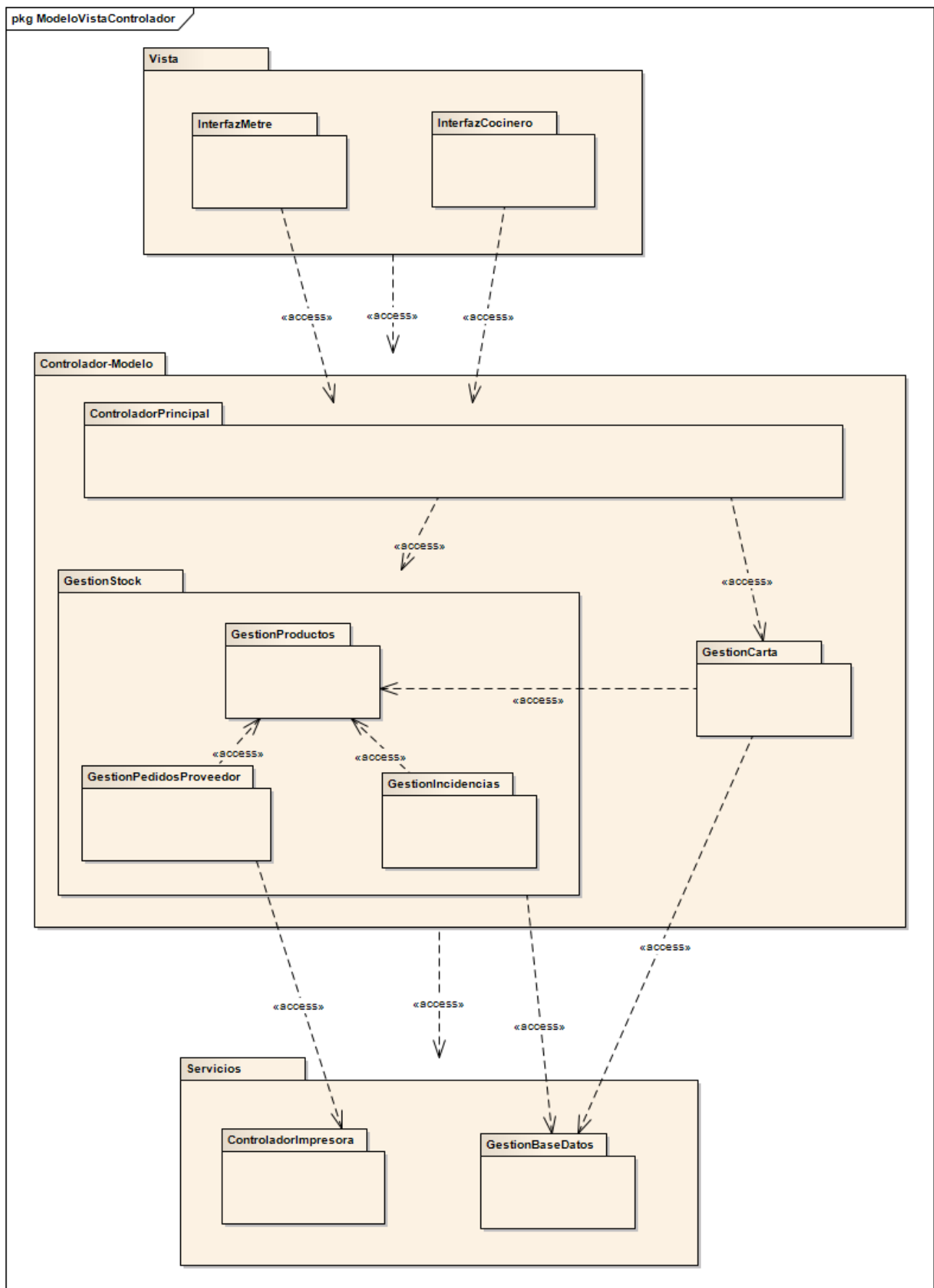
Observamos como los paquetes están relacionados entre sí, controlando funcionalidades 'independientes' dentro de nuestro sistema y dependiendo todos del principal, al que hemos llamado SAGRES, como nuestra aplicación.

Sin embargo, no queda claro, como esta representación se vería afectada por el estilo arquitectónico que utilizaremos, es decir, el Modelo-Vista-Controlador y su lógica dividida en tres capas.

Para este aspecto, hemos diseñado un diagrama que refleja claramente como el anterior diagrama de paquetes queda encuadrado en nuestra lógica, interactuando tanto con la interfaz de usuario como con los servicios de los que disponemos.

Este diagrama muestra perfectamente la cohesión de los paquetes localizados en cada una de las capas, así como el bajo acoplamiento entre estas últimas. Se establece una dependencia de las capas superiores hacia las capas inferiores, no siendo conscientes las capas inferiores de qué paquetes las utiliza.





## 4. DIAGRAMA DE COMPONENTES

---

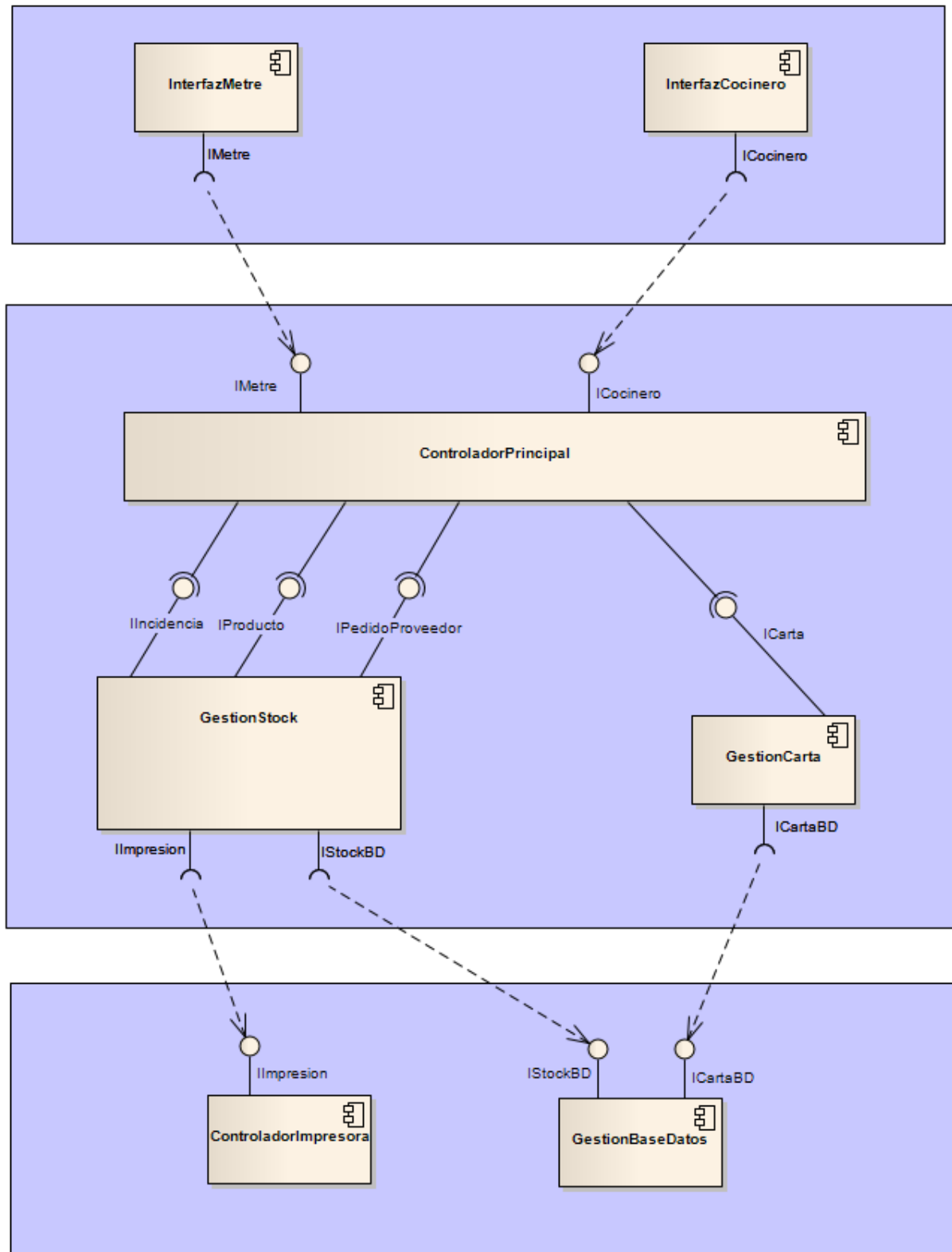
El diagrama de paquetes está estructurado en agrupaciones lógicas, pero necesitamos definir subsistemas, es decir, cerrar partes del sistema en componentes.

La principal característica de este diagrama son las interfaces que hemos añadido para cerrar las partes reutilizables del sistema. Cada interfaz contiene una colección de operaciones y se utiliza para especificar los servicios de una clase o de un componente.

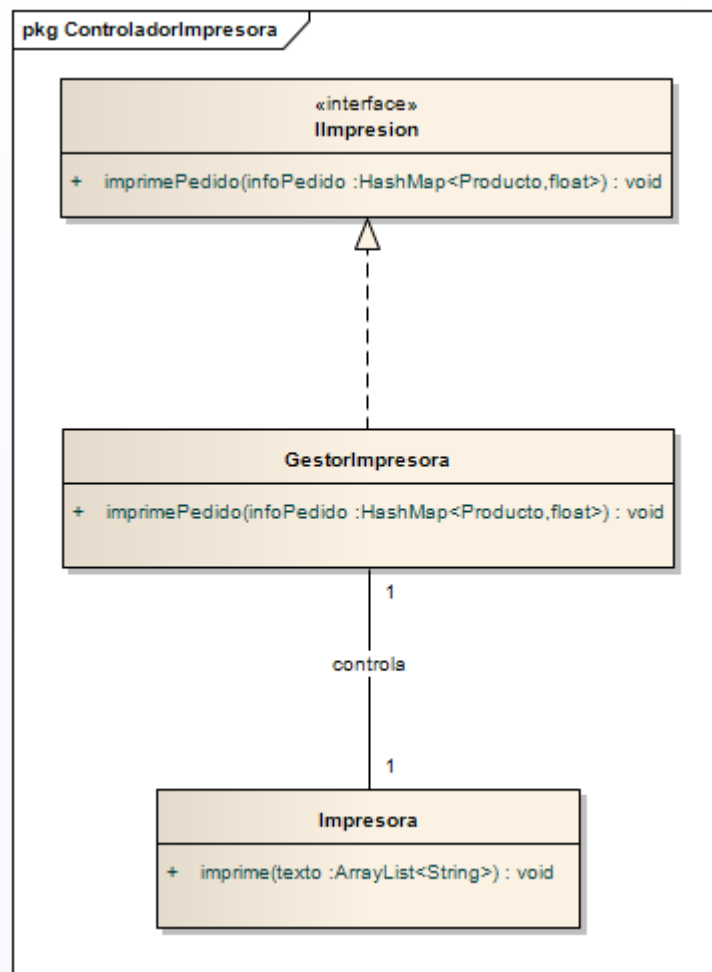
La ventaja de usar interfaces es que se rompe la dependencia directa entre componentes. Un componente que usa una interfaz puede funcionar adecuadamente independientemente del componente que la realiza.

Primero veremos la arquitectura del sistema en base a componentes, y a continuación veremos cada uno de los subsistemas principales, con las interfaces que hemos definido. Todas las operaciones se pueden consultar con más facilidad en nuestro proyecto Enterprise(1a\_Iteracion.eap), esto es así ya que debido a la gran dimensión de las imágenes algunas pueden no apreciarse correctamente.



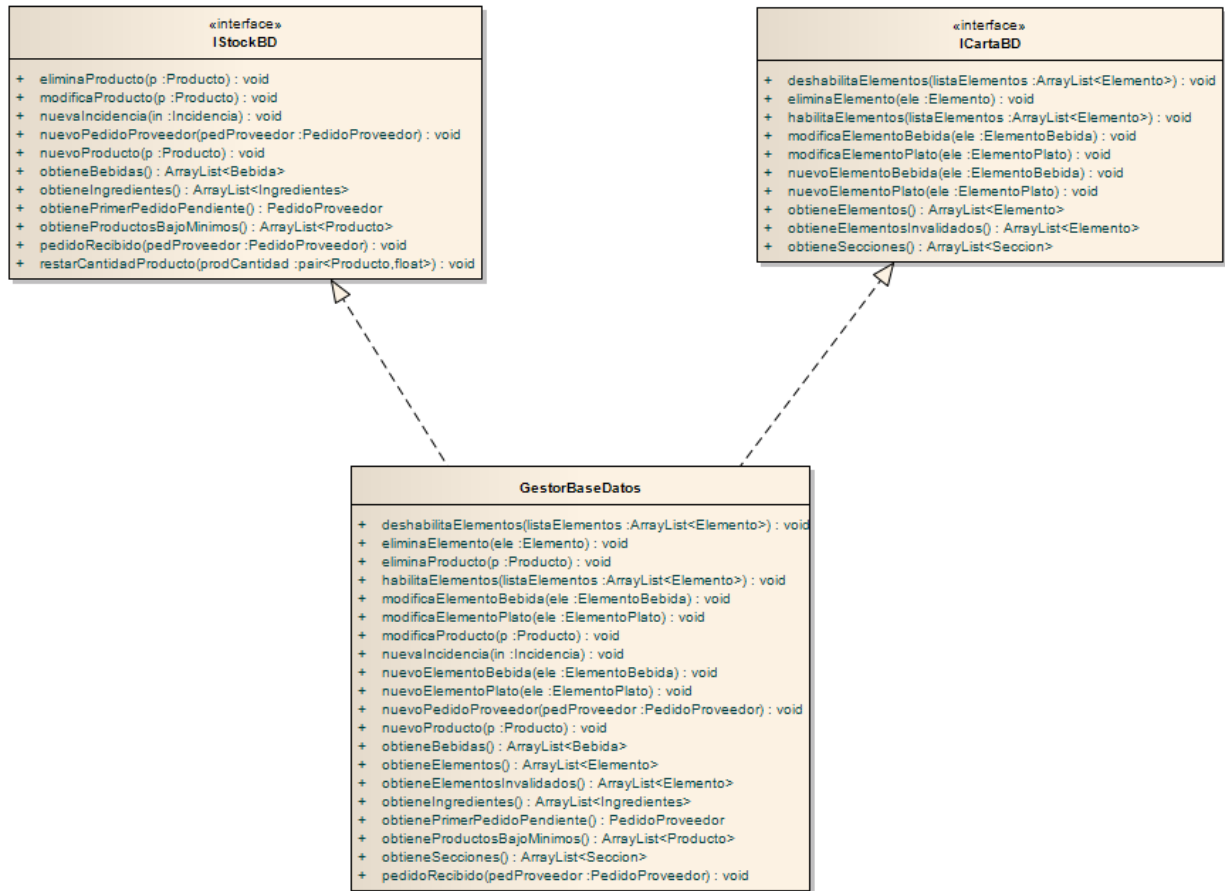


## Controlador Impresora

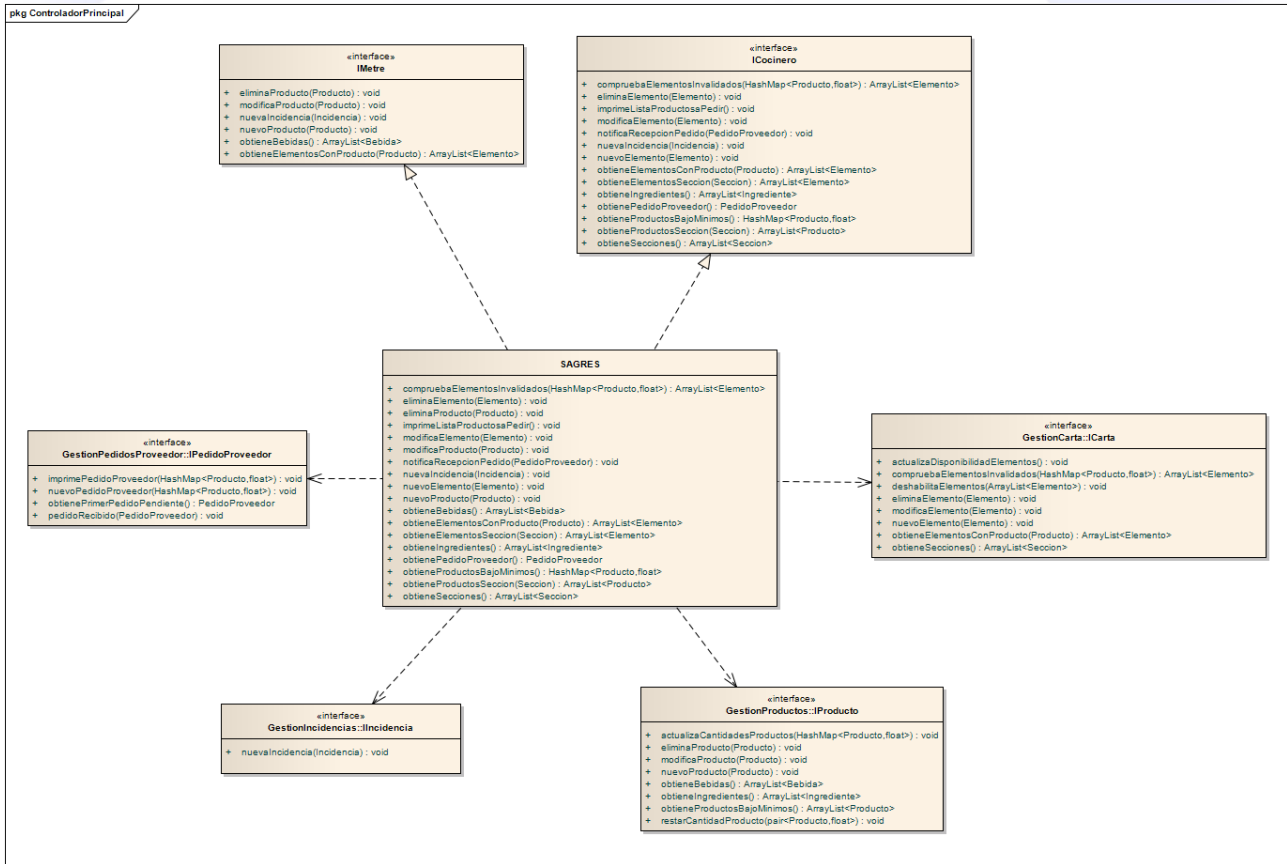


# Gestión de base de datos

pkg GestionBaseDatos



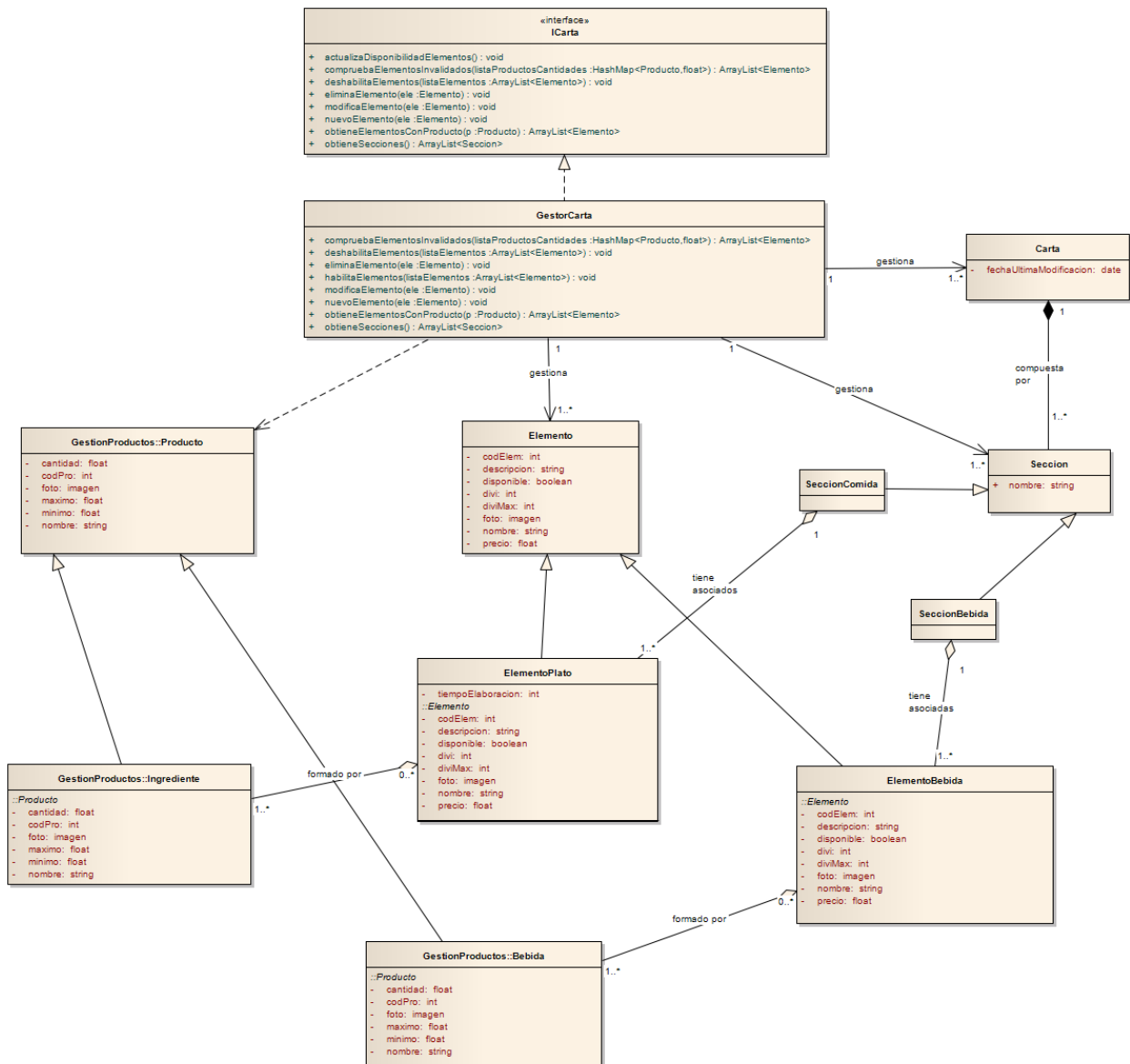
# Controlador principal



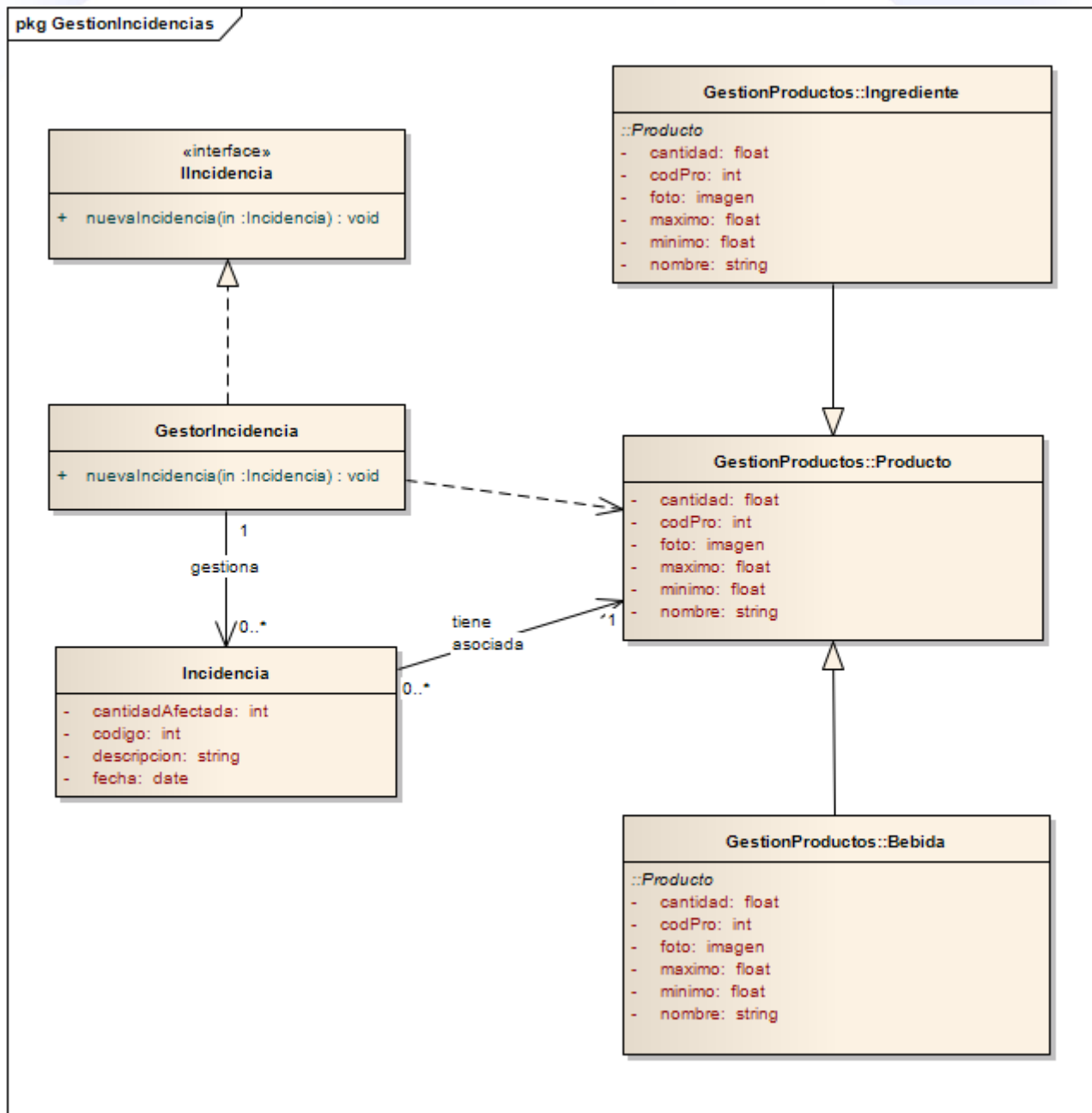


# Gestión Carta

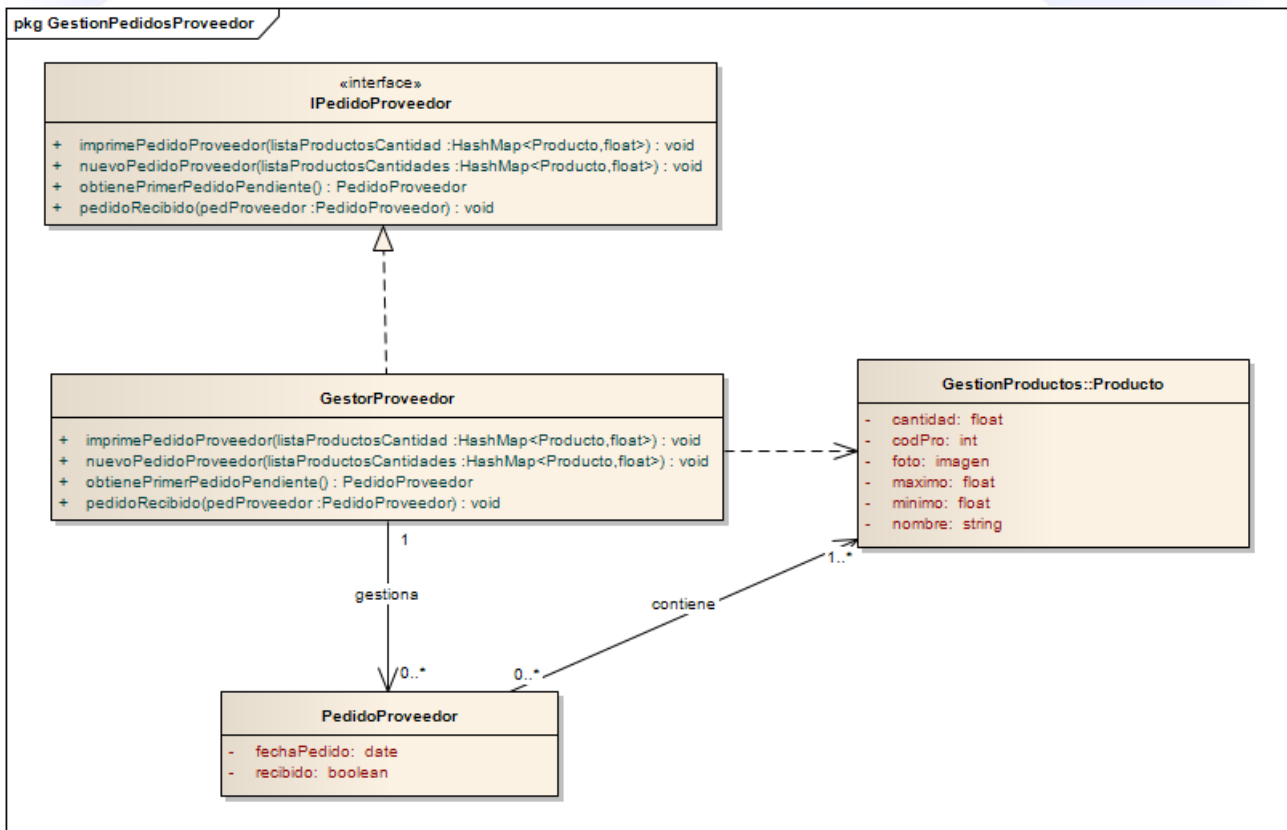
pkg GestionCarta



## Gestión Incidencias

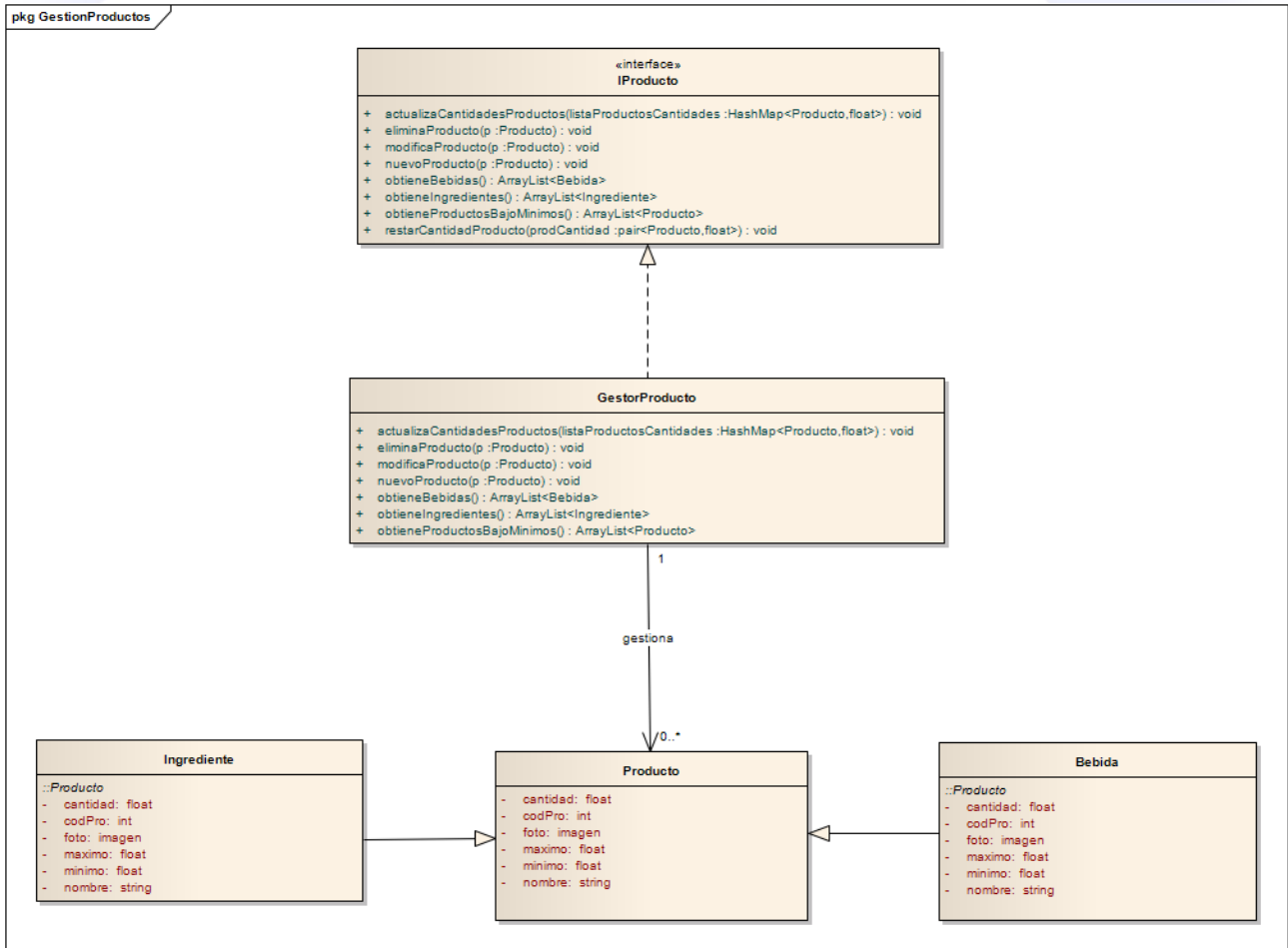


## Gestión Pedidos Proveedor



Sagres

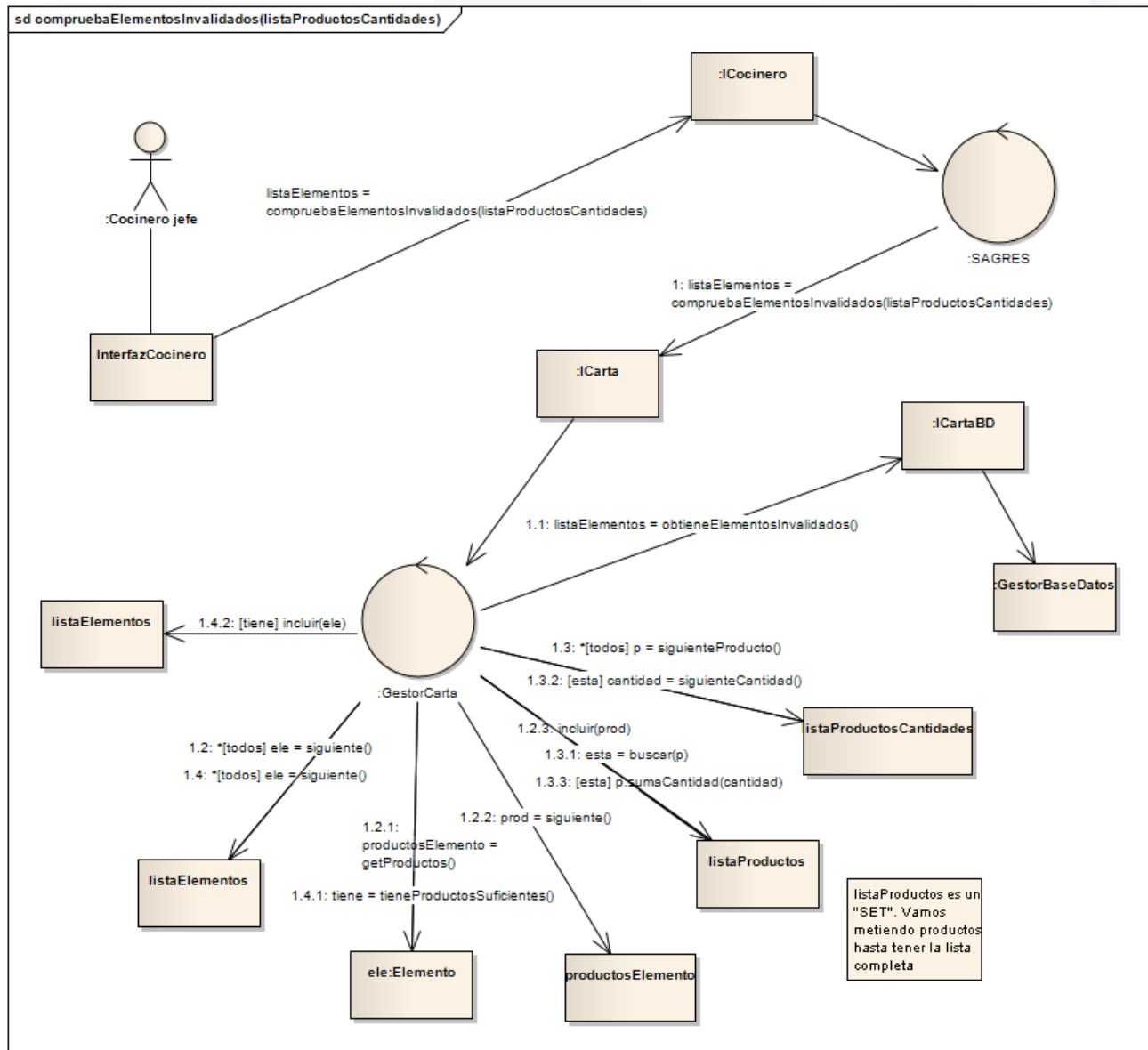
# Gestión Productos



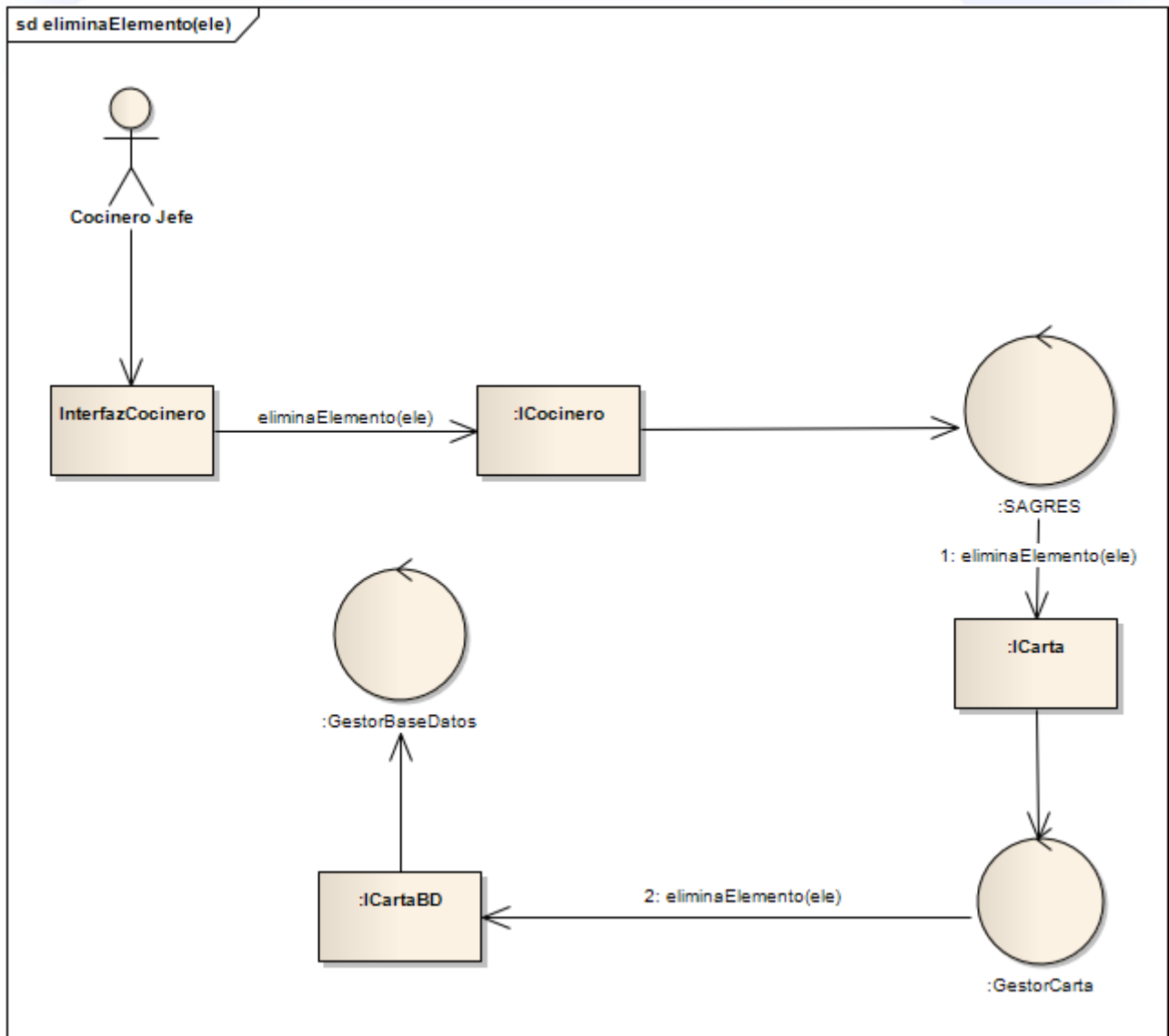
## 5. DIAGRAMAS DE COLABORACIÓN DE DISEÑO

Para cada una de las operaciones de las distintas interfaces tenemos un diagrama de colaboración de diseño que como su propio nombre indica, tratan de escenificar como se comunican unos componentes con otros en cada operación de las interfaces.

### compruebaElementosInvalidados

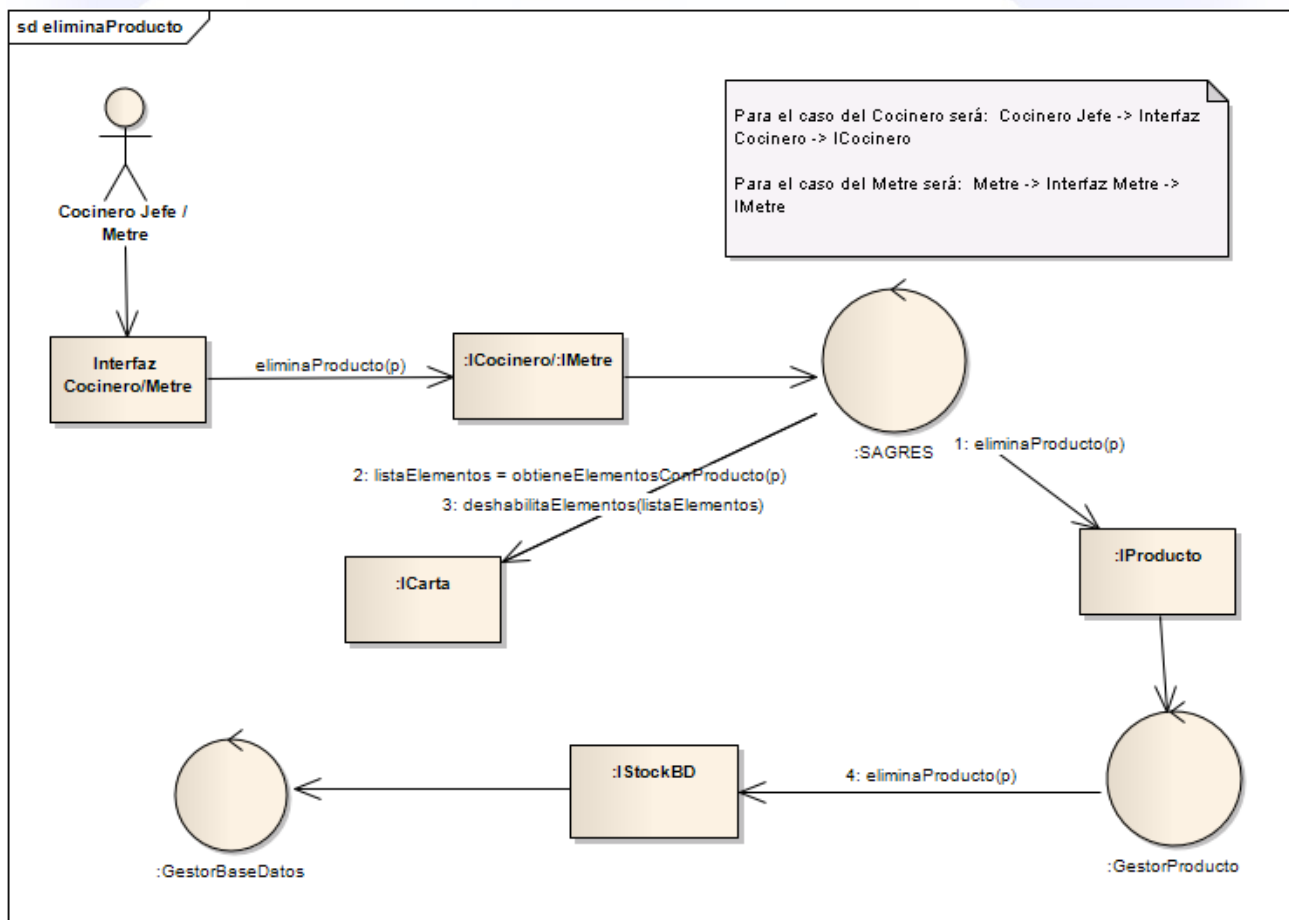


## eliminaElemento



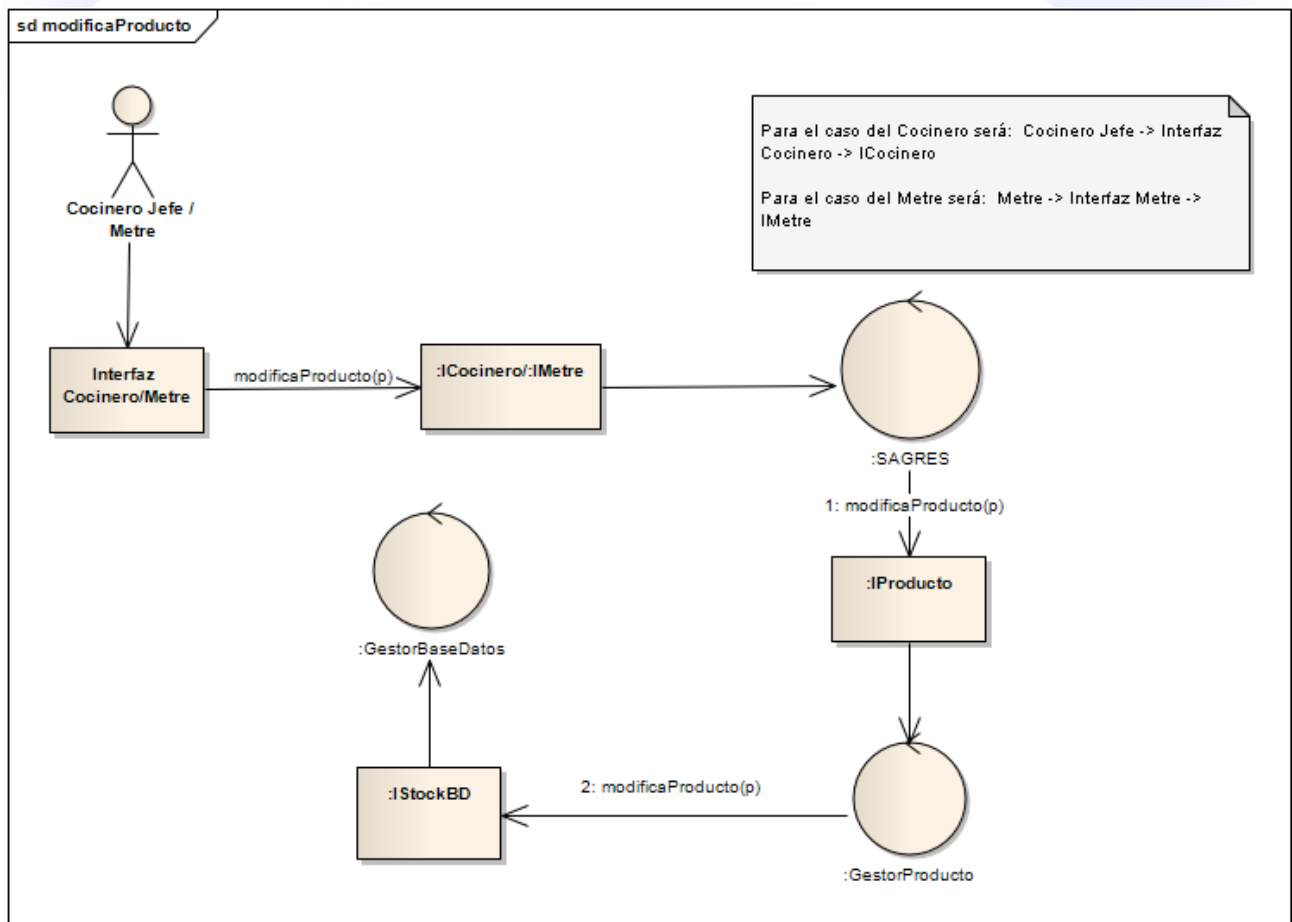


## eliminaProducto

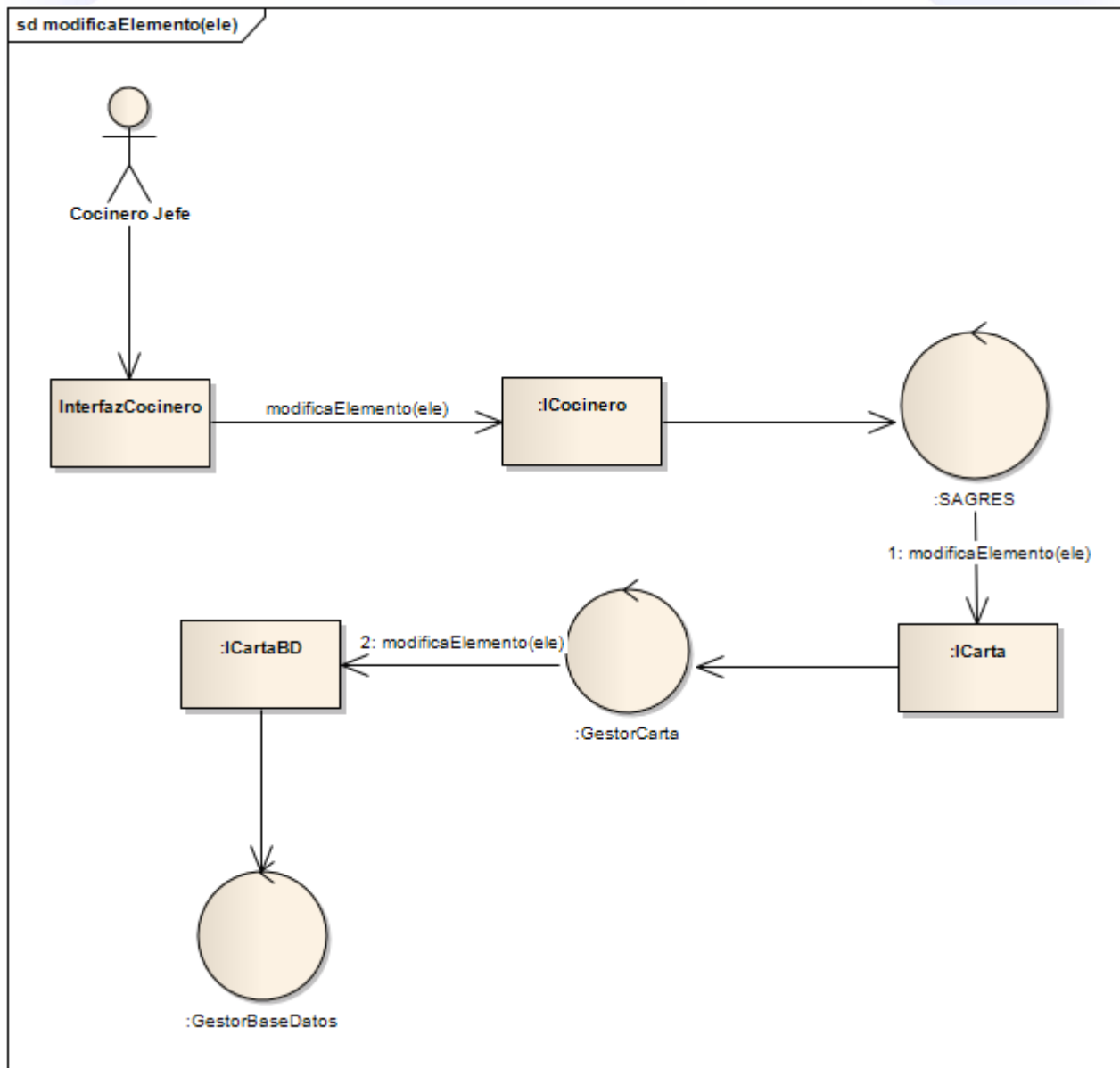


Sagres

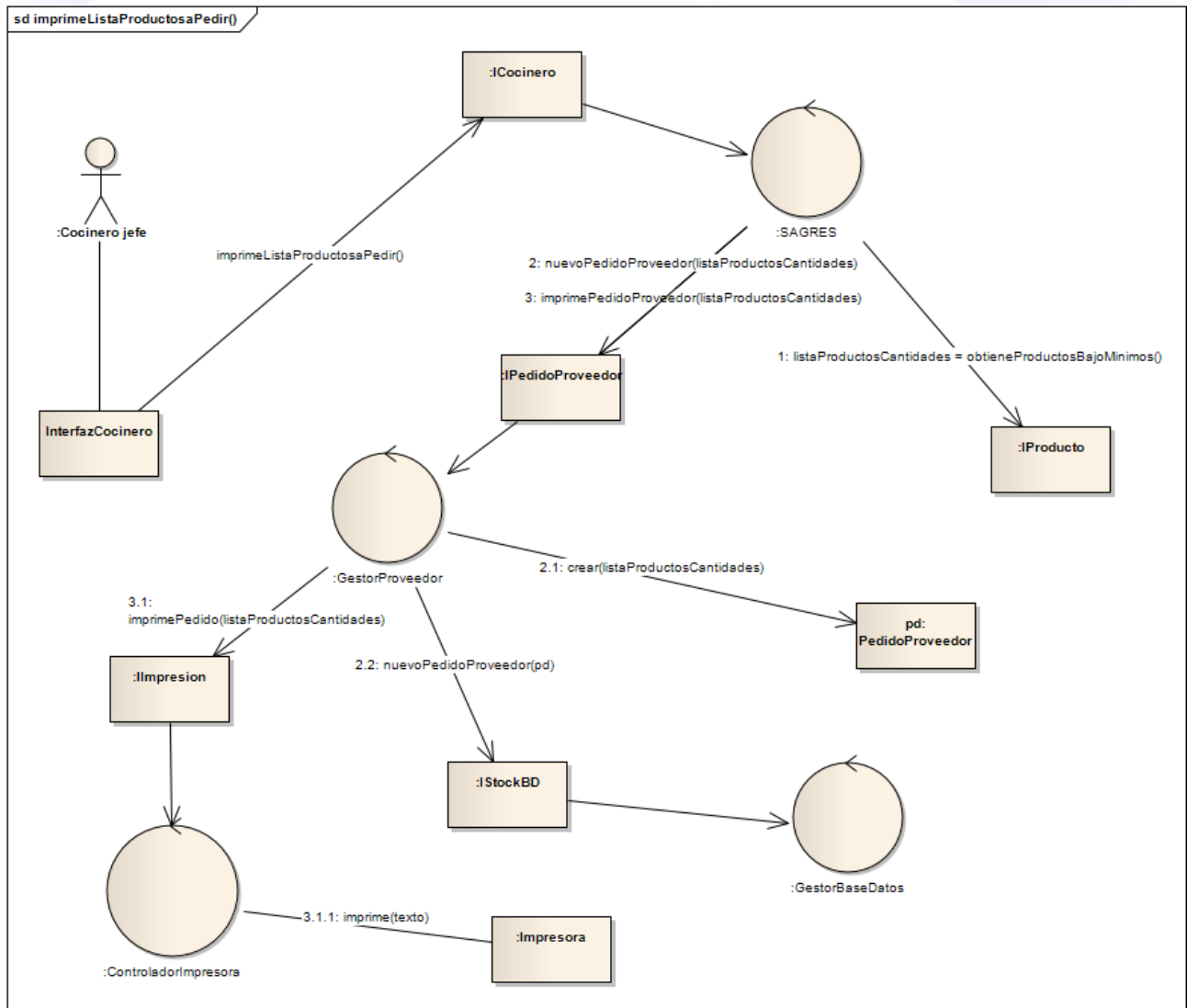
## modificaProducto



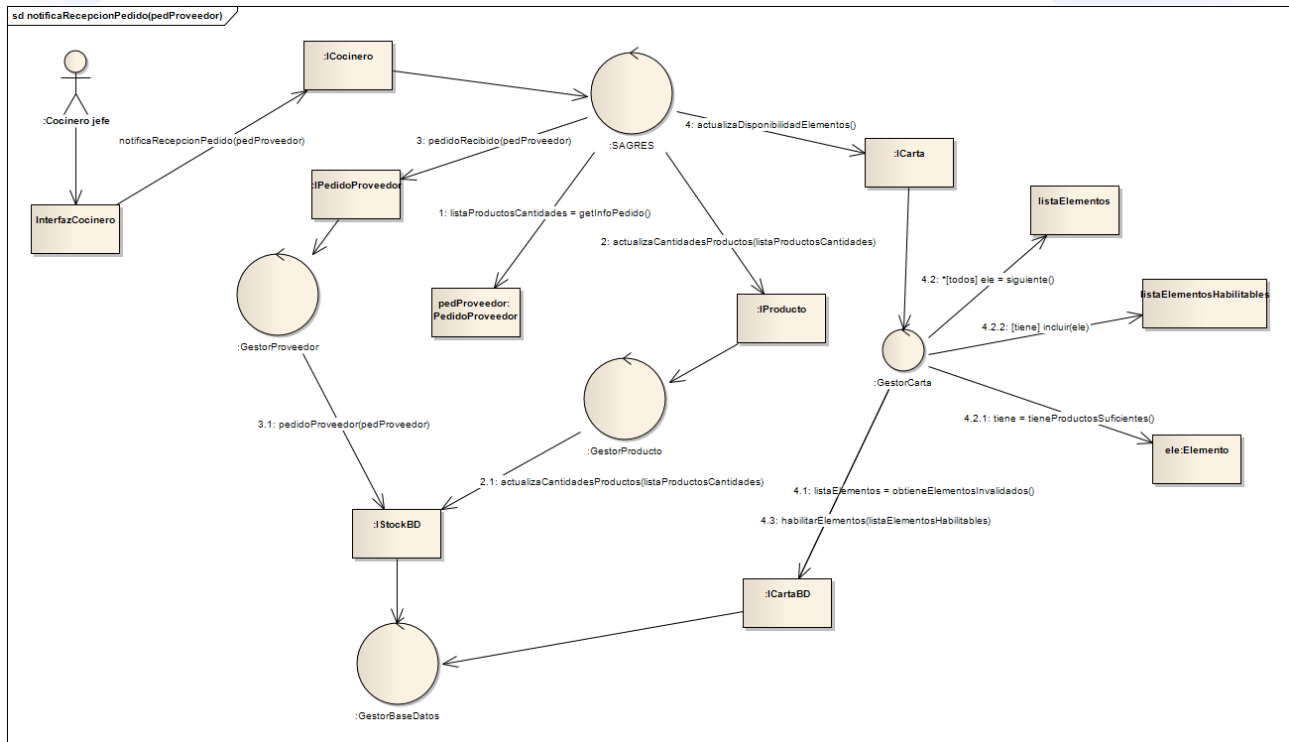
## modificaElemento



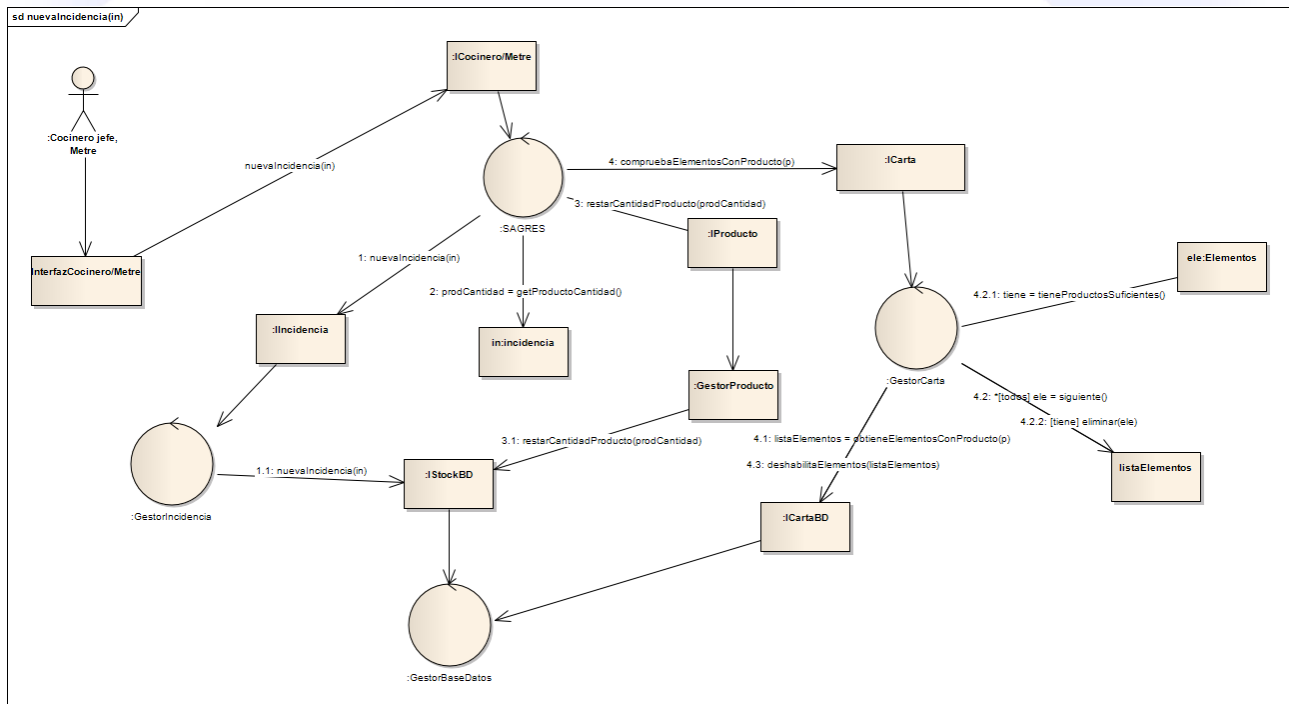
# imprimeListaProductosaPedir



## notificaRecepcionPedidoProveedor



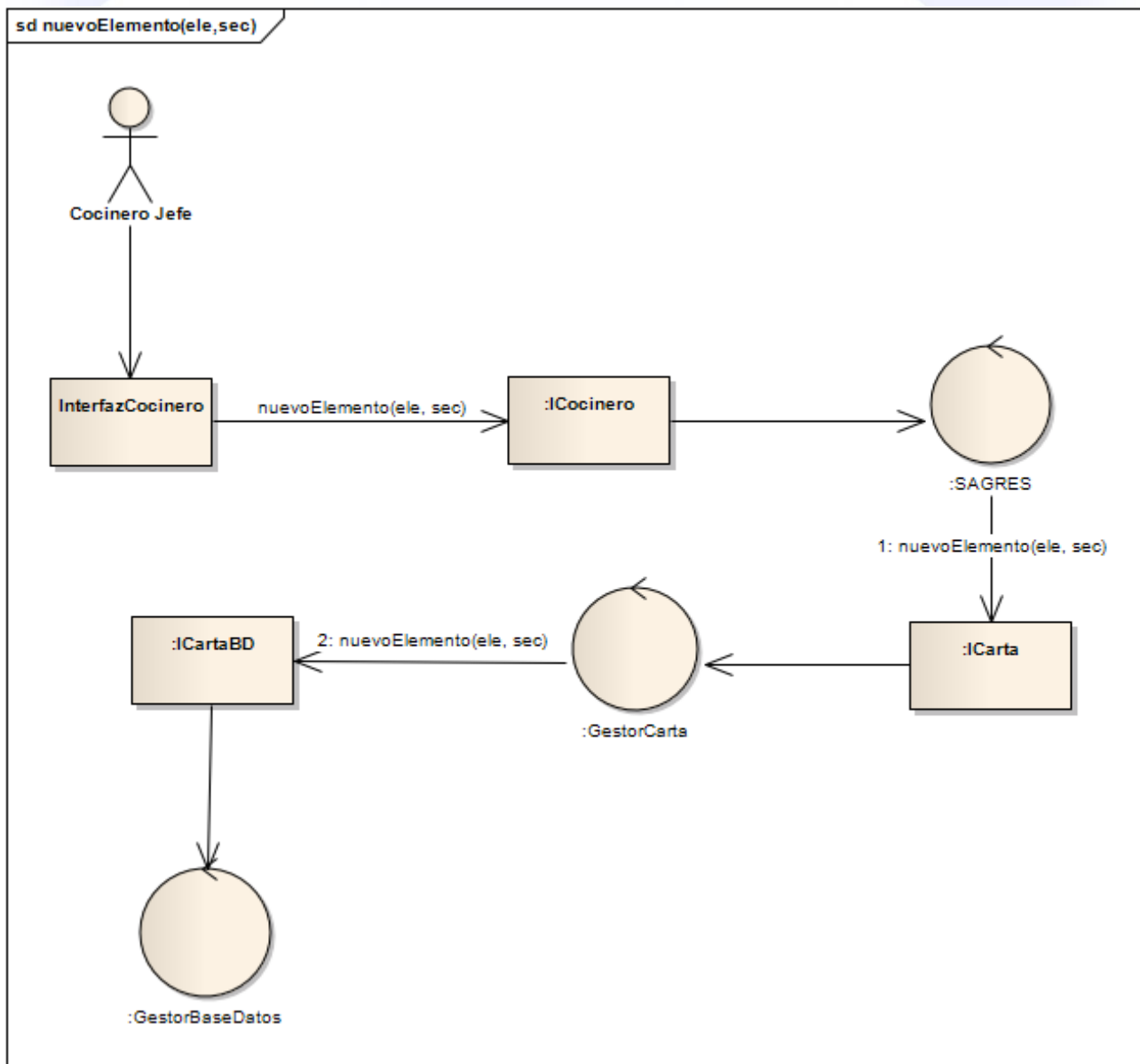
## nuevaIncidencia



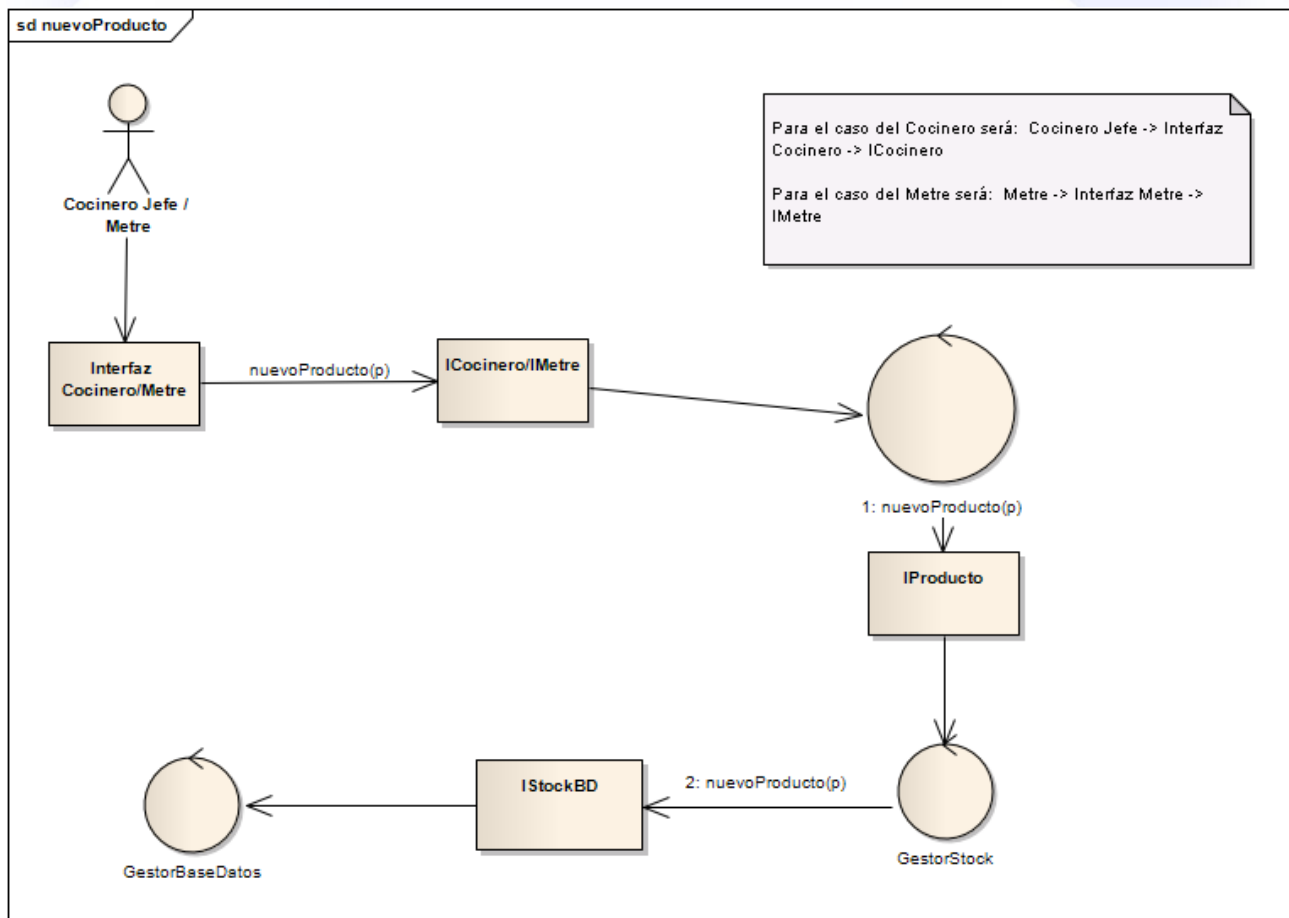
Sagres



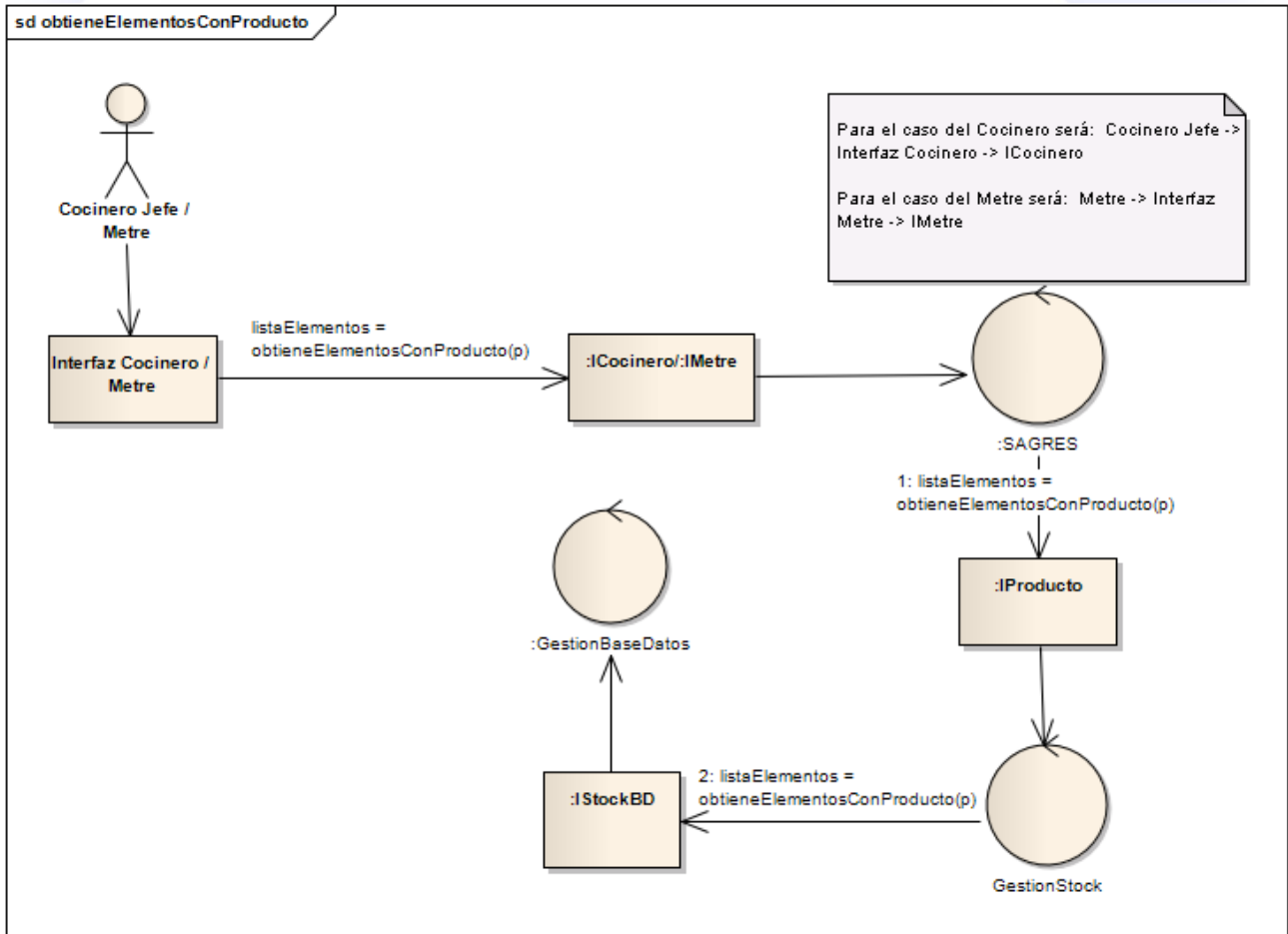
## nuevoElemento



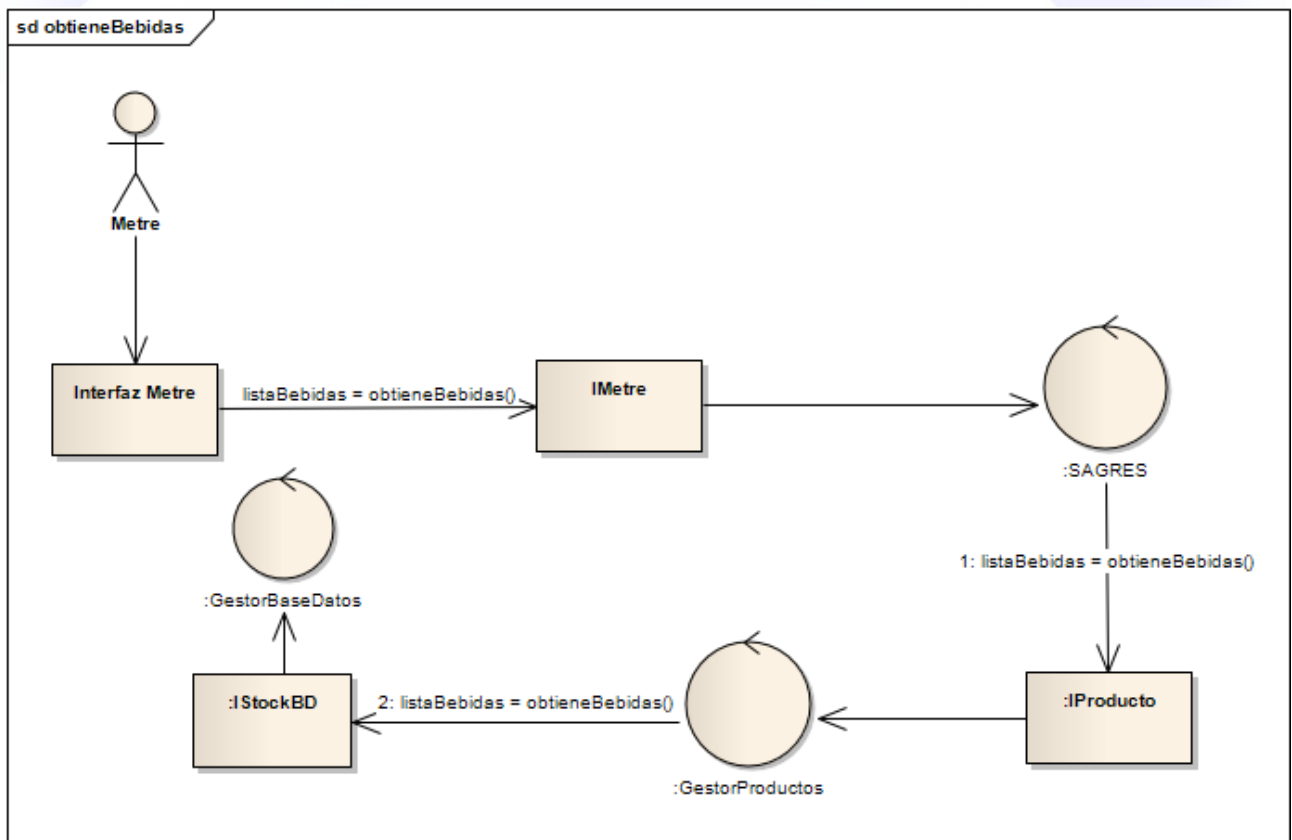
## nuevoProducto



## obtieneElementosConProducto

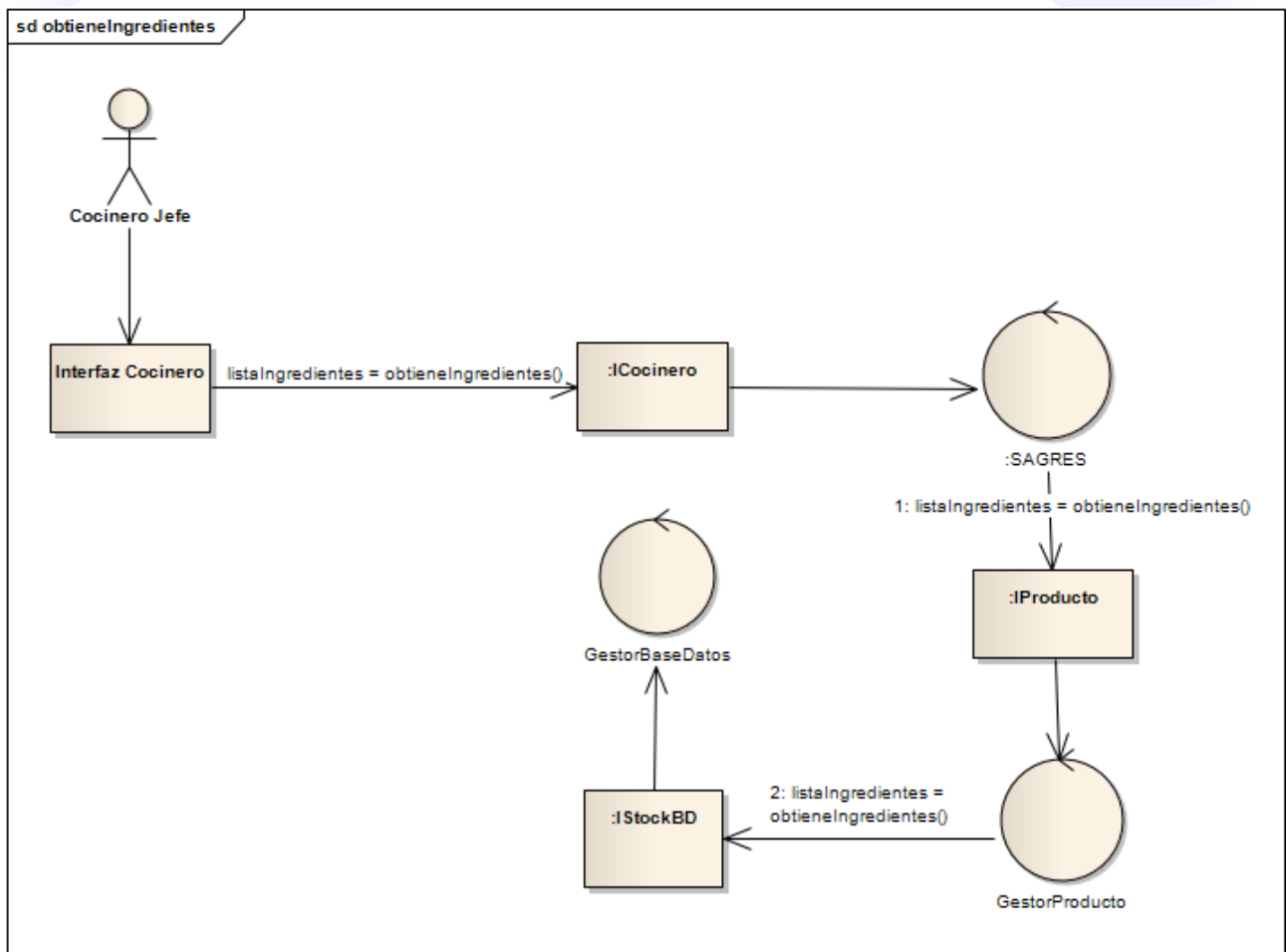


## obtieneBebidas

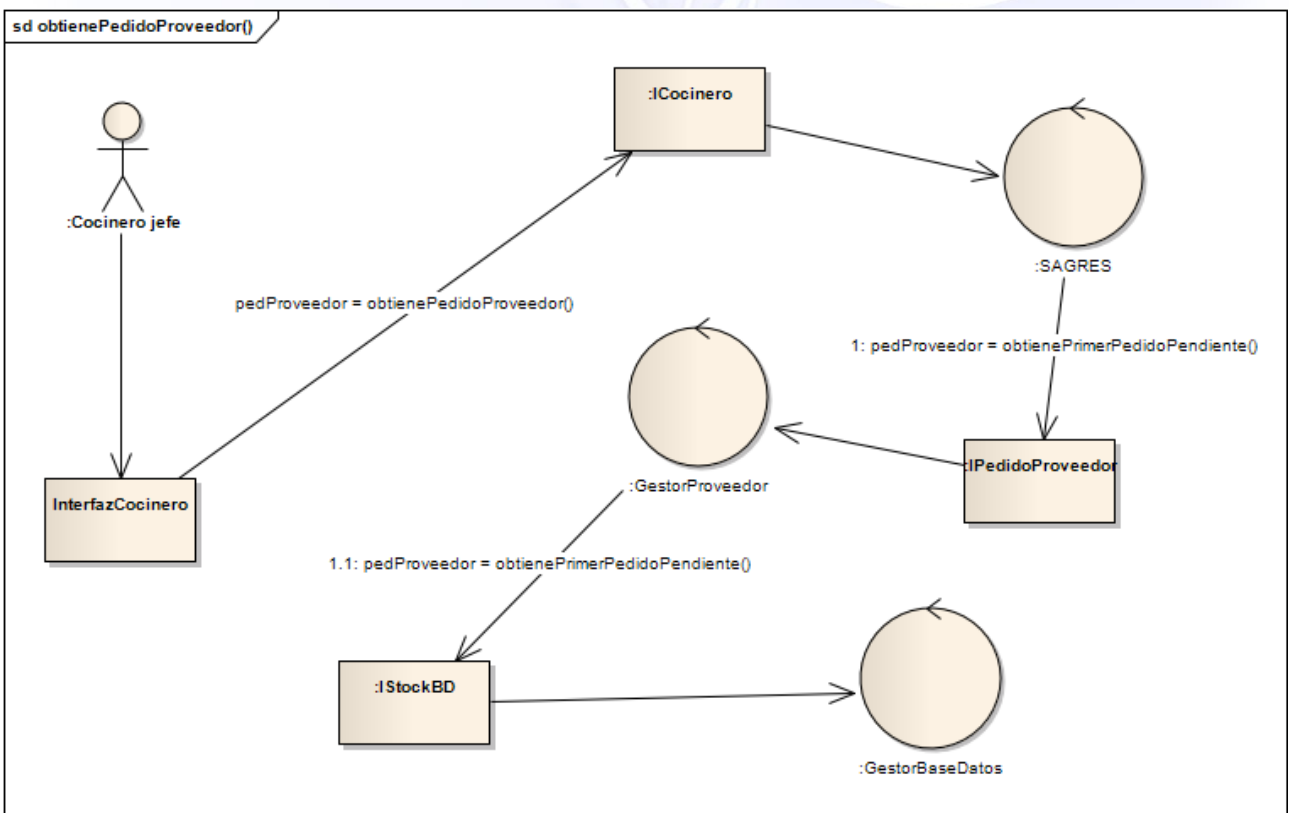
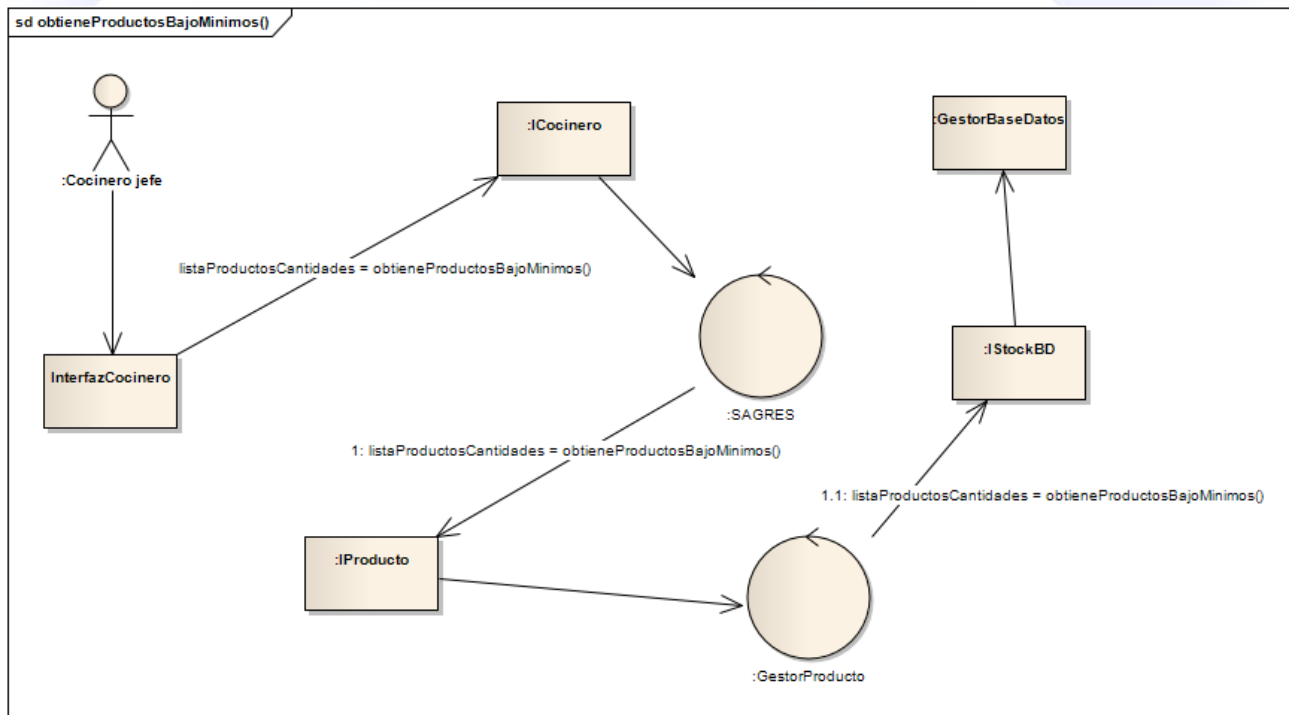


Sagres

## obtieneIngredientes

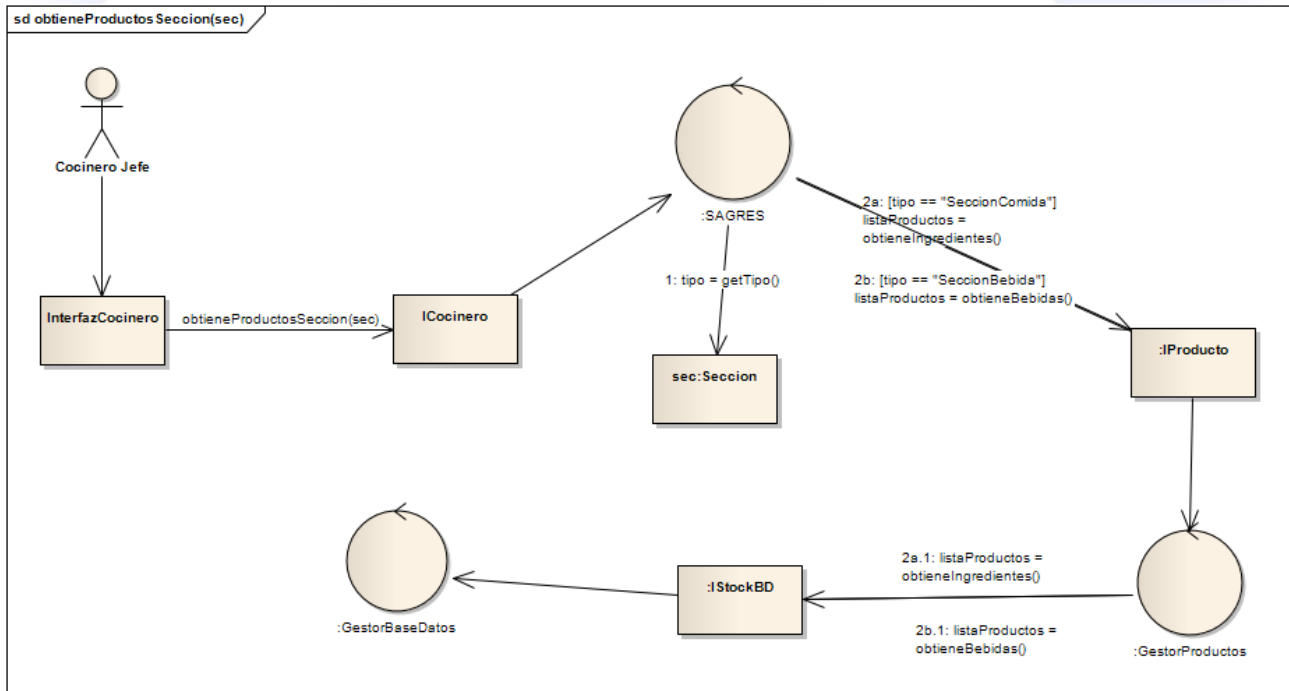


## obtieneProductosBajoMinimos



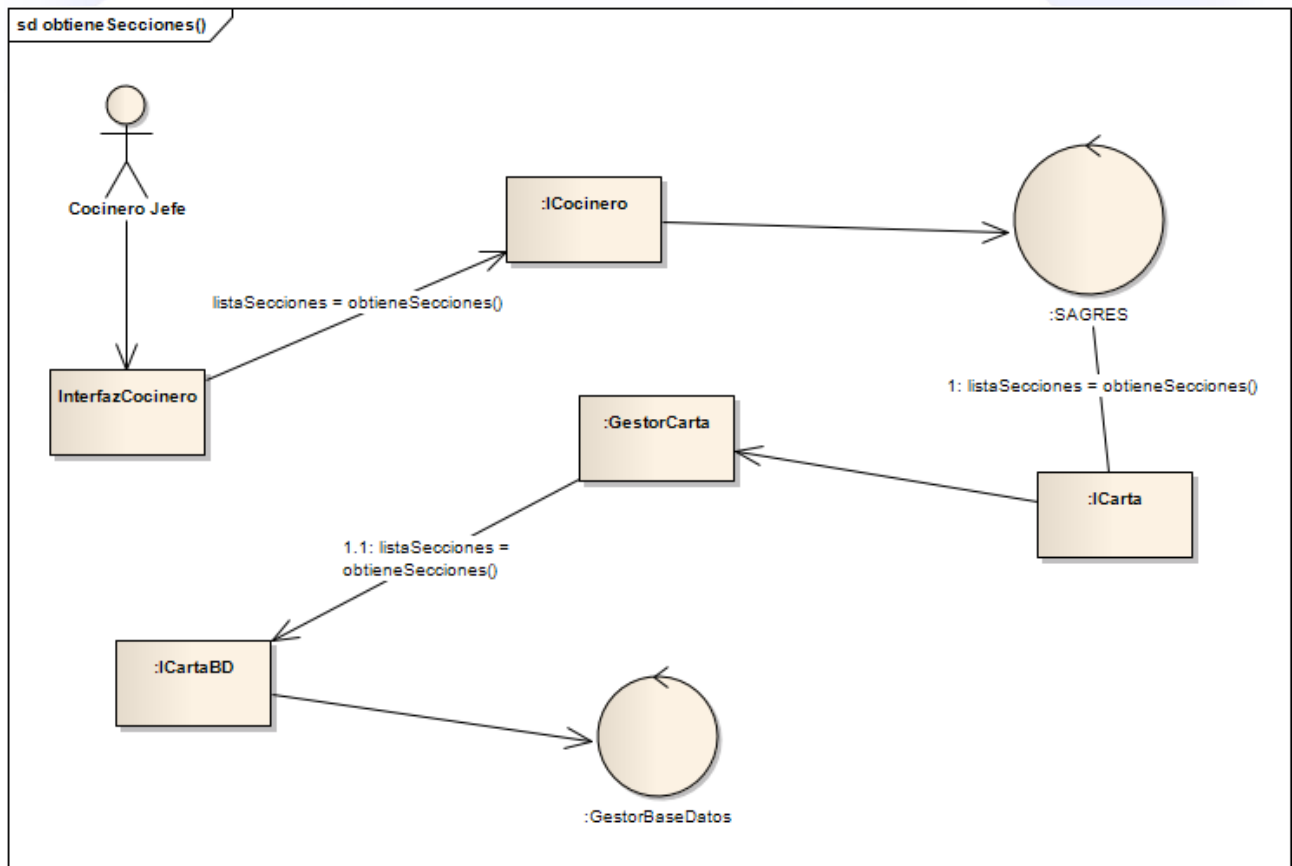


## obtieneProductosSeccion



Sagres

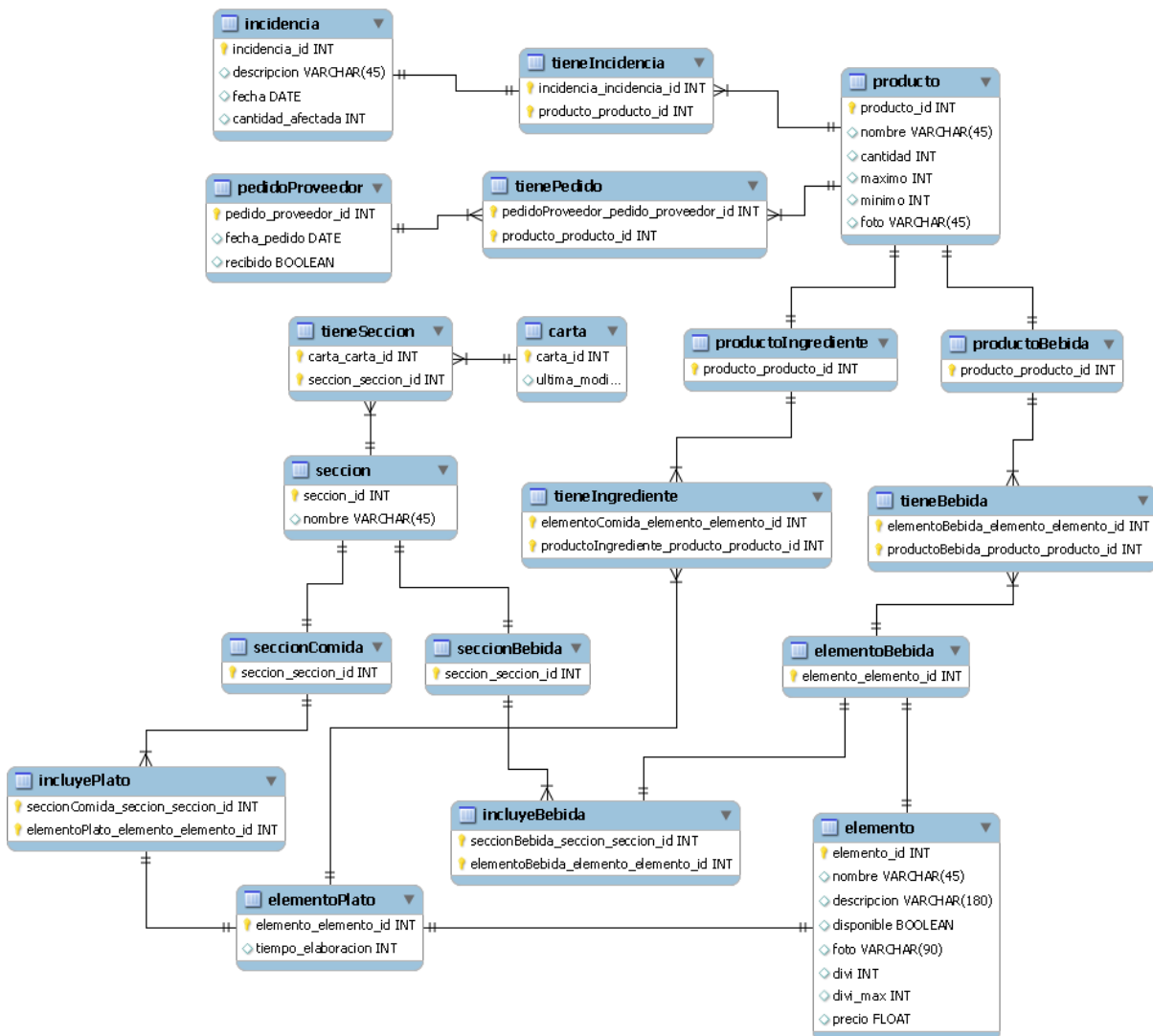
## obtieneSecciones



Sagres

## 6. DIAGRAMA RELACIONAL

En el diagrama relacional se muestran las tablas obtenidas directamente del diagrama de clases. También se ha obtenido el script SQL para implementar las tablas y lo podemos encontrar en su apartado correspondiente a esta iteración



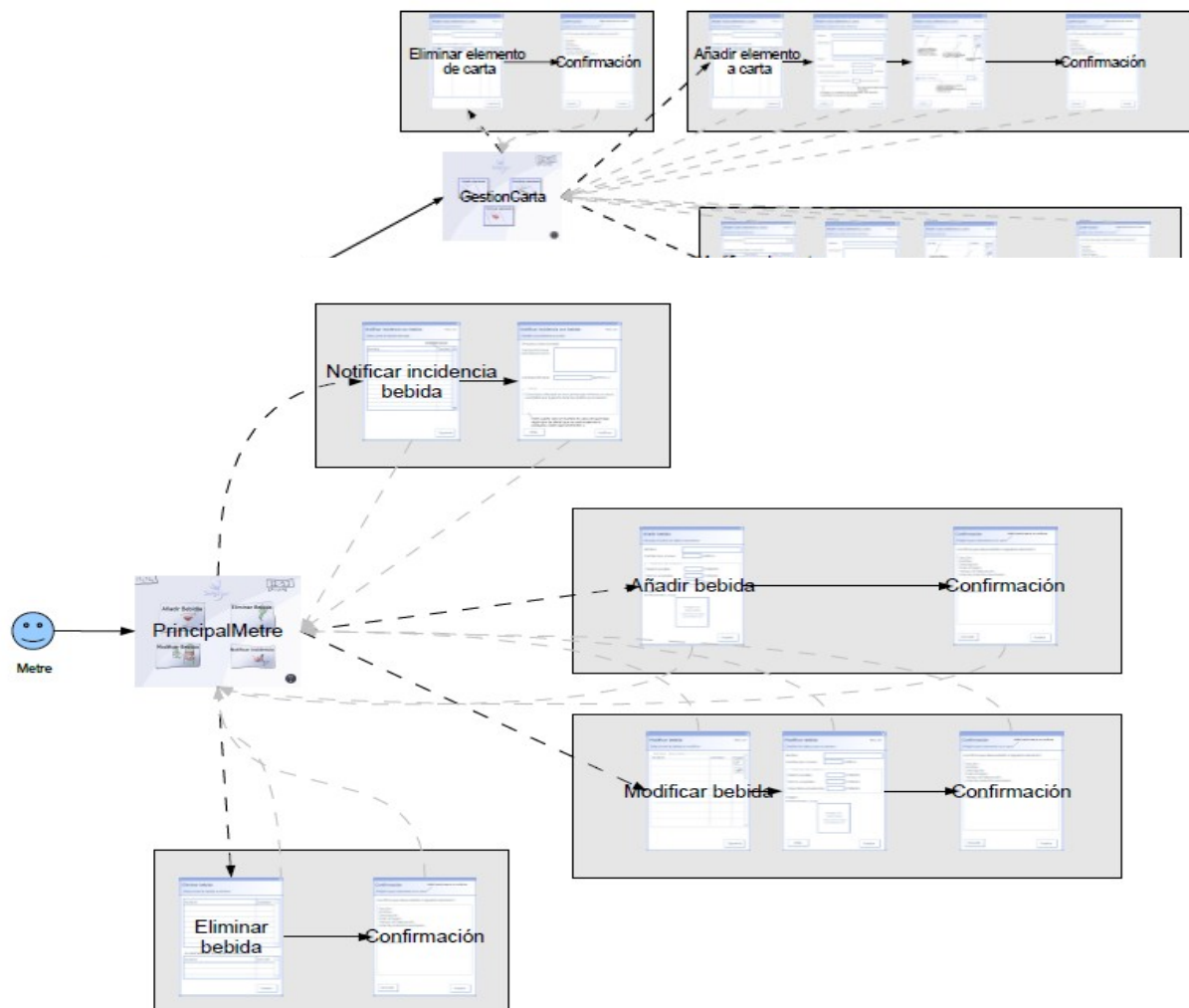
## 7. DISEÑO DE LA INTERFAZ DE USUARIO

### Consideraciones generales de nuestra interfaz

Nuestra interfaz estará compuesta por una serie de menús navegables que seguirán la siguiente jerarquía:

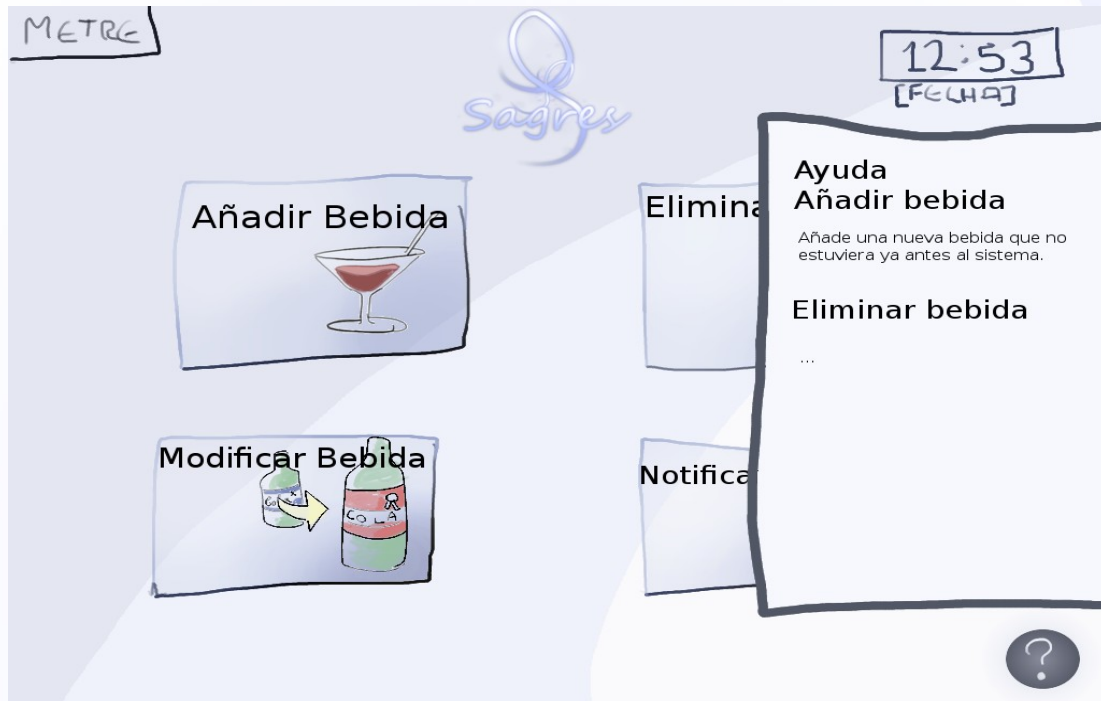
<p>Para el cocinero jefe:</p> <ul style="list-style-type: none"><li>+Menú principal</li><li>+Gestión de carta<ul style="list-style-type: none"><li>-Añadir elemento</li><li>-Modificar elemento</li><li>-Eliminar elemento</li></ul></li><li>+Gestion de ingredientes<ul style="list-style-type: none"><li>-Añadir ingrediente</li><li>-Modificar ingrediente</li><li>-Eliminar ingrediente</li><li>-Notificar incidente</li></ul></li></ul>	<p>Para el metre:</p> <ul style="list-style-type: none"><li>+Gestion de bebidas<ul style="list-style-type: none"><li>-Añadir bebida</li><li>-Modificar bebida</li><li>-Eliminar bebida</li><li>-Notificar incidente</li></ul></li></ul>
--	---

La disposición de los componentes del menú tendrá la siguiente estructura:



En cada menú:

- Mostraremos en todo momento la fecha y hora actuales en la esquina superior derecha de la pantalla.
- Dispondremos de un botón de ayuda que nos mostrará una breve descripción que nos ayudará a entender la funcionalidad de cada una de las opciones.



- Cada vez que realicemos una acción, el sistema nos pedirá confirmación a través de un diálogo con la siguiente estructura:

**Confirmación** Indica la acción que se va a confirmar

Añadir nuevo elemento a la carta

¿Confirma que desea añadir el siguiente elemento?

Sección :  
Nombre:  
Descripción:  
Ruta a imagen:  
Tiempo de elaboración:  
Lista de productos asociados:  
-  
-  
-  
Precio de venta:

Cancelar Aceptar

A continuación entraremos en detalle para cada uno de los menús que el sistema nos ofrece.

## Interfaz para el cocinero jefe

### Menú principal



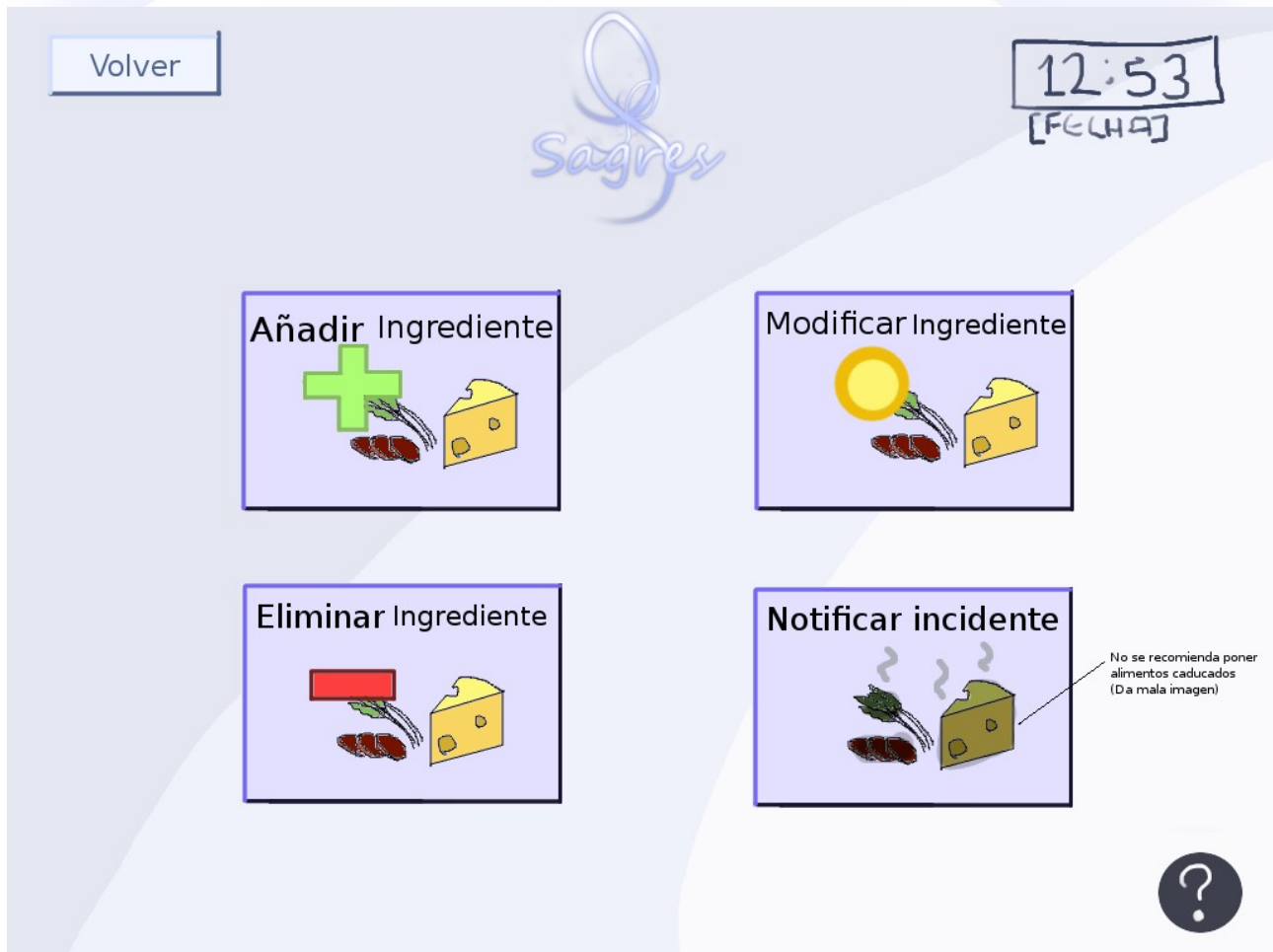
El cocinero jefe tendrá un menú principal que le ofrecerá 4 opciones diferentes:

- Gestión de carta: permitirá gestionar todo lo relativo a la carta.
- Gestión de ingredientes: permitirá gestionar todo lo relativo a los ingredientes registrados en el sistema.
- Imprimir productos a pedir: permitirá obtener una lista de los productos que se encuentran bajo mínimos en el stock.
- Notificar recepción de pedido: permitirá notificar la correcta recepción de una lista de productos por parte del proveedor.

Ofrece también un botón de ayuda que ayudará, mediante una descripción breve, a entender la funcionalidad de cada opción.

A continuación detallaremos cada una de estas opciones.

## Interfaz para la gestión de ingredientes



El cocinero jefe tendrá un menú para la gestión de ingredientes que le ofrecerá 4:

- Añadir ingrediente: registrar un nuevo ingrediente en el sistema.
- Modificar ingrediente: modificar un ingrediente previamente registrado en el sistema.
- Eliminar ingrediente: eliminar un ingrediente previamente registrado en el sistema.

Ofrece también un botón de ayuda que ayudará, mediante una descripción breve, a entender la funcionalidad de cada opción.

A continuación detallaremos cada una de estas opciones.

## Añadir nuevo ingrediente

✕

### Añadir ingrediente

*Introducir todos los datos necesarios*

Nombre

Cantidades

Máximo posible

gramos (gr)

Mínimo aceptable

gramos (gr)

Disponible actualmente

gramos (gr)

Imagen

(Tamaño 200x200 px, en .jpg)

Imagen no disponible

(Click sobre la imagen para insertar una)

Aceptar

Introducimos y confirmamos los datos del nuevo ingrediente en solo un paso.



## Modificar ingrediente


×

Modificar ingrediente

Paso 1/2

Seleccionar el ingrediente a modificar

Ingredientes disponibles

Nombre	Cantidad	Imagen
		
		

Aceptar

El sistema mostrará una lista de los ingredientes registrados en el sistema. El usuario elegirá el producto que desea modificar.

X

## Modificar ingrediente

Paso 2/2

*Cambiar los datos que se deseen*

Nombre

Cantidades

Máximo posible

gramos (gr)

Mínimo aceptable

gramos (gr)

Disponible actualmente

gramos (gr)

Imagen

(Tamaño 200x200 px, en .jpg)

Imagen no disponible

(Click sobre la imagen para insertar una)

Atrás

Aceptar

Se mostrarán los datos del ingrediente seleccionado y el usuario podrá modificarlos según sus intereses.

## Eliminar ingrediente

✕

### Eliminar ingrediente

*Seleccionar el ingrediente a eliminar*

Nombre	Cantidad	

Se deshabilitarán los siguientes elementos

Nombre	Sección	

Aceptar

Al seleccionar el ingrediente a eliminar, el sistema nos mostrará una lista de los platos que pasarán a estar no disponibles por incluir dicho ingrediente.

## Notificar incidencia con ingrediente

Notificar incidencia con ingrediente

Paso 1/2

Seleccionar el ingrediente afectado

Cantidad actual

Nombre	Cantidad

Siguiente

El sistema nos mostrará la lista de ingredientes del sistema junto con su cantidad. Seleccionaremos el ingrediente afectado.

×

Notificar incidencia con ingrediente

Paso 2/2

Detallar el problema ocurrido

[Producto seleccionado]

Descipción breve  
(1000 caracteres máximo)

Cantidad afectada

[gr/litros/...]

Alerta

- El producto afectado se encuentra bajo mínimos en stock, se añadirá a la siguiente lista de pedidos a proveedor.

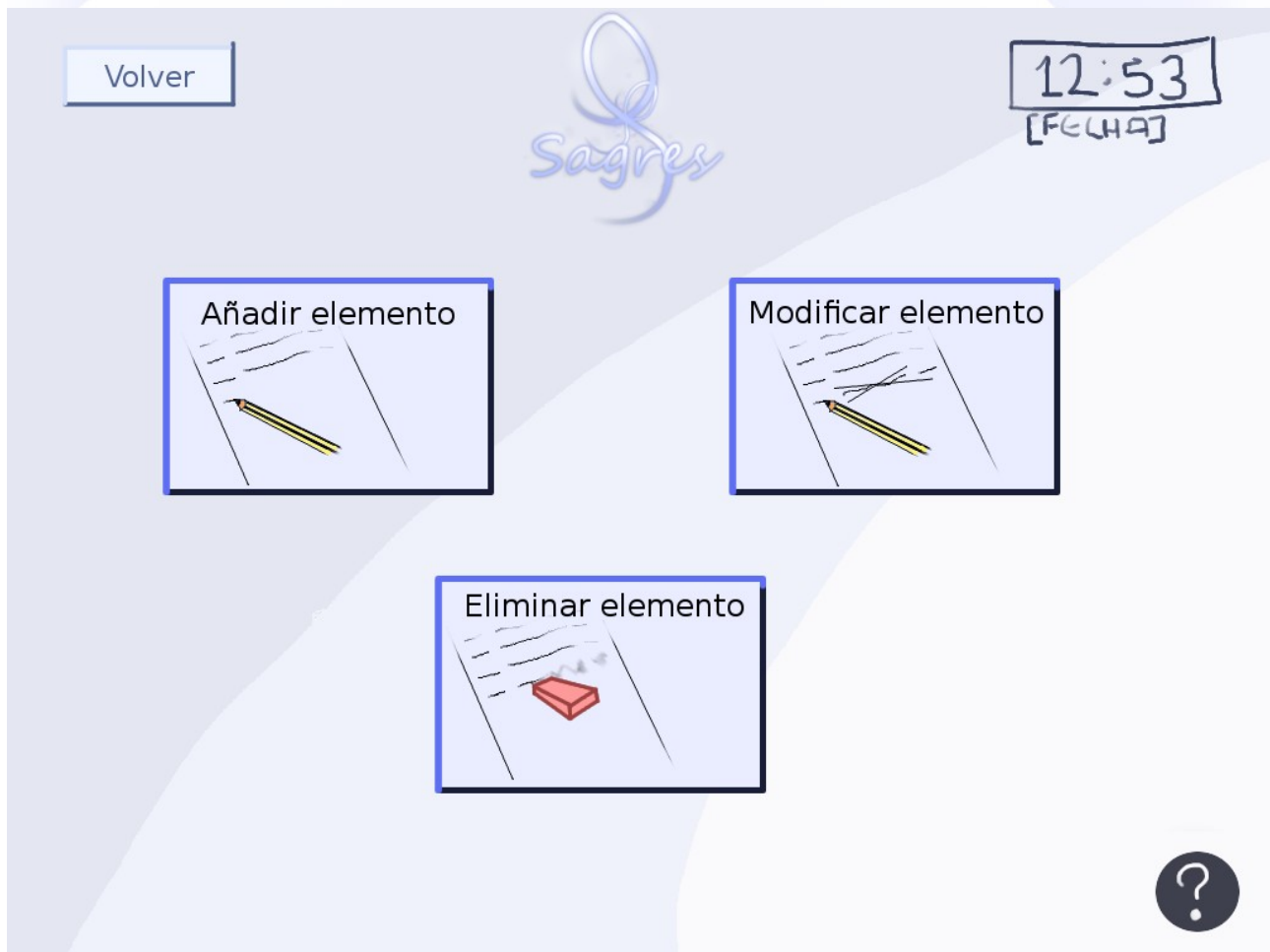
Este cuadro solo se muestra en caso de que haya algún tipo de alerta (que se esté acabando el producto, o bien que se termine...)

Atrás

Notificar

En un segundo y último paso describiremos brevemente la incidencia e indicaremos la cantidad afectada. El sistema nos mostrará una alerta si la cantidad no afectada del ingrediente se encuentra bajo mínimos en el stock.

## Interfaz para la gestión de la carta



El cocinero jefe tendrá un menú para la gestión de la carta que le ofrecerá 3 opciones:

- Añadir elemento: añade un nuevo elemento (plato o bebida) a la carta del restaurante.
- Modificar elemento: modifica un elemento previamente incluido en la carta.
- Eliminar elemento: elimina un elemento previamente incluido en la carta.

Ofrece también un botón de ayuda que ayudará, mediante una descripción breve, a entender la funcionalidad de cada opción.

A continuación detallaremos cada una de estas opciones.

## Añadir nuevo elemento a carta

Añadir nuevo elemento a carta

Paso 1/3

Sección a la que pertenece

Elija una sección:

Productos ya asociados a la sección

Nombre	Descripción	Precio	Duración

Siguiente

En este primer paso para incluir un nuevo elemento en la carta deberemos especificar la sección del nuevo elemento.

Añadir nuevo elemento a carta

Paso 2/3

Rellenar los datos del nuevo elemento

Nombre

Descripción

Imagen

Examinar

Precio de venta

€

Tiempo medio de elaboración

minutos

Atributos del plato

• El elemento se puede dividir en  porciones/raciones

0 ó vacío para indicar que es indivisible

El cuadro y su contenido solo se mostrarán si el elemento no pertenece a la sección de bebidas

Anterior

Siguiente

Una vez asignada la sección procederemos a completar los datos sobre el elemento.





## Modificar elemento de carta

Esta interfaz es similar a la de introducir un nuevo elemento. La diferencia está en que primero seleccionaremos un elemento ya existente y el sistema nos mostrará los datos que modificaremos según nuestros intereses.

×

Modificar un elemento de carta

Paso 1/3

Seleccionar elemento

Elija una sección:

Seleccione el producto de la lista

Nombre	Descripción	Precio	Duración

Siguiente

En un primer paso elegiremos la sección del elemento a modificar para que el sistema nos muestre los elementos asociados. Acto seguido seleccionaremos el elemento de la lista.

✕

Modificar un elemento de carta

Paso 2/3

Datos del elemento

Aparecerán todos los campos rellenos, justificados a la derecha

Nombre

Descripción

Imagen

Examinar

Precio de venta

€

Tiempo medio de elaboración

minutos

Atributos del plato

• El elemento se puede dividir en  porciones/raciones

0 ó vacío para indicar que es indivisible

El cuadro y su contenido solo se mostrarán si el elemento no pertenece a la sección de bebidas

Anterior

Siguiente

Los datos del elemento se mostrarán en pantalla como muestra la imagen. El usuario podrá modificar estos datos.



## Eliminar elemento de carta

✕

Eliminar un elemento de carta

Seleccionar elemento

Elija una sección:

Seleccione el producto de la lista

Nombre	Descripción	Precio	Duración

Siguiente

Elegimos la sección donde se encuentre el elemento que deseamos eliminar, lo seleccionamos y confirmamos.

## Imprimir lista de productos a pedir

### Notificación

*Listar productos a pedir*

Se van a pedir los siguientes productos

Nombre	Cantidad

CancelarPedir

El sistema mostrará una lista de los productos que estén por debajo de su mínimo en el stock.

## Notificar recepción de productos

### Notificación

*Llegada de productos pedidos*

Se confirma la llegada de los siguientes productos

Nombre	Cantidad

Se habilitarán los siguientes elementos

Nombre	Sección

Cancelar

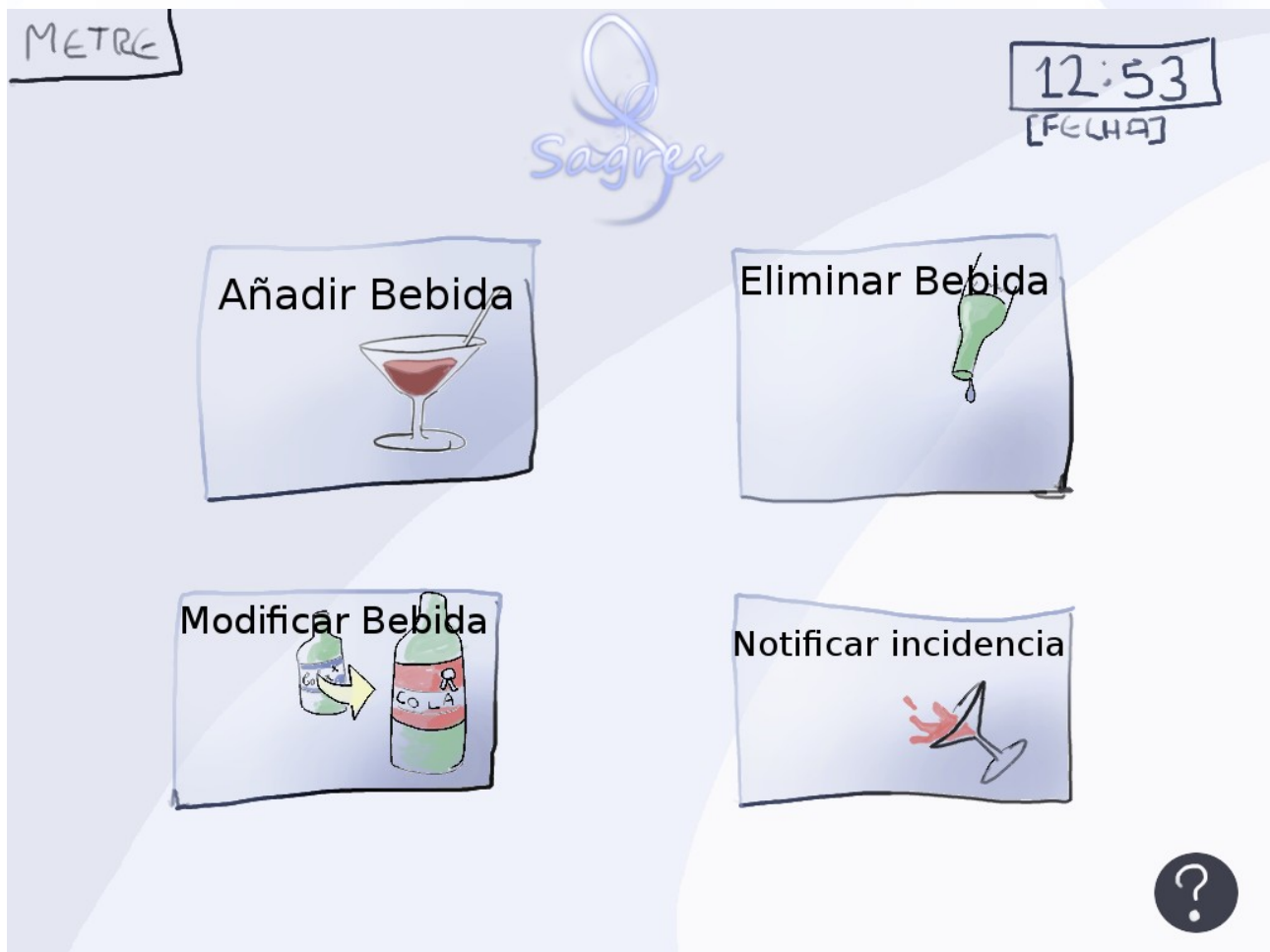
Aceptar

Es el proceso inverso a la eliminación de un producto: el sistema ofrecerá la lista de productos del sistema y el usuario seleccionará los productos recibidos junto con su cantidad.

Por cada elemento recibido, el sistema actualizará y mostrará una lista de elementos que se encontraban deshabilitados y que quedarán de nuevo disponibles tras la recepción de productos.

## Interfaz para el metre

### Menú principal



El metre tendrá un menú principal que le ofrecerá 4 opciones:

- Añadir bebida: registrar un nueva bebida como producto en el sistema.
- Eliminar bebida: eliminar una bebida previamente registrada en el sistema.
- Modificar bebida: modificar una bebida previamente registrada en el sistema.

Ofrece también un botón de ayuda que ayudará, mediante una descripción breve, a entender la funcionalidad de cada opción.

A continuación detallaremos cada una de estás opciones.



## Añadir nueva bebida

✕

### Añadir bebida

*Introducir todos los datos necesarios*

Nombre

Cantidad por envase

mililitros

Número de envases

Máximo posible

Unidades

Mínimo aceptable

Unidades

Disponible actualmente

Unidades

Imagen

(Tamaño 200x200 px, en .jpg)

Imagen no disponible

(Click sobre la imagen para insertar una)

Aceptar

Introducimos y confirmamos los datos de la nueva bebida en solo un paso.



## Modificar bebida

Modificar bebida

Paso 1/2

*Seleccionar la bebida a modificar*

Bebidas disponibles

Nombre	Cantidad	Imagen
		
		

Siguiente

El sistema mostrará una lista de las bebidas registrados en el sistema. El usuario elegirá la bebida que desea modificar.

✕

Modificar bebida

Paso 2/2

Cambiar los datos que se deseen

Nombre

Cantidad por envase

mililitros

Número de envases

Máximo posible

Unidades

Mínimo aceptable

Unidades

Disponible actualmente

Unidades

Imagen

(Tamaño 200x200 px, en .jpg)

Imagen no disponible

(Click sobre la imagen para insertar una)

Atrás

Aceptar

Se mostrarán los datos de la bebida seleccionada y el usuario podrá modificarlos según sus intereses.

## Eliminar bebida

✕

### Eliminar bebida

*Seleccionar la bebida a eliminar*

Nombre	Cantidad	△
		▽

Se deshabilitarán los siguientes elementos

Nombre	Sección	△
		▽

Aceptar

Al seleccionar la bebida a eliminar, el sistema nos mostrará una lista de las bebidas que pasarán a estar no disponibles por incluir el producto.

### Notificar incidencia con bebida

[illegible]

El sistema nos mostrará la lista de bebidas del sistema junto con su cantidad. Seleccionaremos la bebida afectada.

×

Notificar incidencia con bebida

Paso 2/2

Detallar el problema ocurrido

[Producto seleccionado]

Descripción breve  
(1000 caracteres máximo)

Cantidad afectada

[gr/litros/...]

Alerta

- El producto afectado se encuentra bajo mínimos en stock, se añadirá a la siguiente lista de pedidos a proveedor.

Este cuadro solo se muestra en caso de que haya algún tipo de alerta (que se esté acabando el producto, o bien que se termine...)

Atrás

Notificar

En un segundo y último paso describiremos brevemente la incidencia e indicaremos la cantidad afectada. El sistema nos mostrará una alerta si la cantidad no afectada de la bebida se encuentra bajo mínimos en el stock.

## APÉNDICE 1.0

<i>Fecha</i>	29-03-2010
<i>Descripción del problema</i>	
<i>Impacto del problema</i>	
<i>Soluciones adoptadas</i>	<ul style="list-style-type: none"><li>• Se ha generado el documento de análisis inicial.</li></ul>
<i>Anexos a la versión</i>	

Sagres

## APÉNDICE 1.1

<i>Fecha</i>	31-03-2010
<i>Descripción del problema</i>	<p>Se han encontrado algunas operaciones que habíamos previsto en un principio y luego quizás no sean necesarias.</p> <p>Algunos nombres de operaciones no cumplen el estilo que el equipo de planificación había estipulado.</p> <p>Se han encontrado errores en algunos diagramas de colaboración de diseño.</p>
<i>Impacto del problema</i>	<p>Varias operaciones como por ejemplo, generarCodigoProducto... no son necesarias, puesto que el código puede ser generado por la base de datos y es más conveniente que sea así.</p> <p>Varios diagramas de colaboración tienen errores en el uso de interfaces y utilizan operaciones que no deben.</p>
<i>Soluciones adoptadas</i>	<ul style="list-style-type: none"><li>• Se ha cambiado nombres de operaciones para que su uso esté de acuerdo al estilo previsto en el documento de planificación.</li><li>• Se han eliminado algunas operaciones que no son necesarias.</li><li>• Se ha rehecho parte de los diagramas de colaboración de Diseño.</li><li>• Se ha informado al equipo de implementación de estos hechos para que tomen las medidas necesarias.</li></ul>
<i>Anexos a la versión</i>	



## APÉNDICE 1.2

<i>Fecha</i>	05-04-2010
<i>Descripción del problema</i>	El equipo de implementación ha descubierto algunos fallos en los diagramas de colaboración que debían ser solventados para clarificar el funcionamiento de la aplicación.
<i>Impacto del problema</i>	<p>Los distintos gestores que componen nuestro sistema controlaban métodos de control de datos introducidos por el usuario. Este hecho era erróneo puesto que esta gestión deberían llevarla las interfaces, encargándose los gestores de trabajar siempre con datos correctos.</p> <p>En los anteriores diagramas de colaboración no se contemplaban las relaciones entre objetos, por ejemplo Incidencia-Producto. Esto podría haber conllevado graves problemas puesto que, si seguimos el ejemplo anterior, al extraer los datos de una incidencia de la base de datos no sabríamos con que producto esta relacionada.</p> <p>También se ha encontrado un fallo con la inserción de objetos en la base de datos. El proceso que se seguía era el de insertar los datos proporcionados, generar un código, y obtener de la base de datos los del objeto completos con código y todo e introducirlo en la lista de objetos del gestor correspondiente. Ese proceso se ha cambiado porque resultaba poco eficiente y presentaba conflicto a la hora de introducir las relaciones entre objetos.</p>
<i>Soluciones adoptadas</i>	<ul style="list-style-type: none"><li>• Eliminadas las funciones de comprobación de los gestores y por lo tanto de los diagramas de colaboración</li><li>• Modificado el proceso de inserción de objetos en la base de datos de los diagramas de colaboración</li></ul>
<i>Anexos a la versión</i>	

## APÉNDICE 1.3

<i>Fecha</i>	18-04-2010
<i>Descripción del problema</i>	La revisión de los documentos de modelado de requisitos y de análisis ha supuesto un cambio radical en el diseño del sistema.
<i>Impacto del problema</i>	<p>El diseño planteado en el anterior documento de diseño presentaba una serie de fisuras graves corregidas por el profesor:</p> <ul style="list-style-type: none"><li>• La caché de objetos contemplada en cada uno de los gestores suponía algo imposible de manejar puesto que al encontrarse la aplicación en varios terminales resultaba imposible mantenerla actualizada. Para ello deberíamos haber utilizado objetos distribuidos y era algo demasiado engorrosa para una aplicación de tan pequeñas magnitudes</li><li>• El controlador principal no presentaba ninguna funcionalidad.</li><li>• Faltaban métodos que hacían que algunas vistas de la interfaz no se mostraran al completo.</li><li>• La gestión de la base de datos no se llevaba a cabo correctamente</li></ul>
<i>Soluciones adoptadas</i>	<ul style="list-style-type: none"><li>• Se ha eliminado la caché de los gestores.</li><li>• Se ha asignado al controlador principal la funcionalidad de capa de abstracción entre la vista y el modelo. Los paquetes ya no se comunicaban entre ellos sino que se delega esta función al controlador principal.</li><li>• Se ha añadido una capa en la gestión de la base de datos para el paso de operaciones con objetos a sentencias SQL.</li><li>• Se han añadido los métodos necesarios para el funcionamiento completo de las vistas.</li></ul>
<i>Anexos a la versión</i>	