Documento de MODELADO DE REQUISITOS

Sagres

Sistema de Administración y Gestión de RESTaurantes

Daniel Guerrero Martínez
Carlos Salas Morales
José David Dionisio Ruiz
Ángel Luis García Sánchez

V3.2

ÍNDICE DE CONTENIDO

Apartado de control de versiones....6
Introducción....7
Primera iteración....7
Segunda iteración....7
Tercera iteración....8
1. Modelo funcional....9
Identificación de actores....9
Cocinero Jefe....9
Metre....9
Cliente....9
Identificación de los requerimientos funcionales....10
Diagrama de Casos de Uso....12
Todo el sistema....12
Primera Iteración....13
Segunda iteración....14
Tercera Iteración....15
Descripción de los casos de uso....16
Gestión Carta....16
Añadir elemento a la carta....16
Eliminar elemento a la carta....18
Modificar elemento de la carta....19
Gestión Stock....20
Imprimir lista de productos a pedir....20
Notificar recepción de pedido del proveedor....21
Añadir ingrediente....22
Eliminar ingrediente....23
Modificar ingrediente....24
Notificar incidencia con Ingrediente....25
Añadir bebida....26
Eliminar bebida....27
Modificar bebida....28
Notificar incidencia con bebida....29
Cola de cocina....30

Cambia estado plato....30
Cola de bar....31
Cambia estado bebida....31
Pedidos de cliente....32
Nuevo pedido....32
Modifica pedido....33
Facturación....34
Cambia estado Factura....34
Pide Factura....35
Pedidos de Hotel....36
Nuevo pedido web....36
Modifica pedido web....37
Consulta Estadística....38
2. Subsistemas funcionales....39
Identificación de los subsistemas funcionales....39
Diagrama de paquetes....39
3. Requisitos no funcionales....40
Identificación y descripción de los requisitos no funcionales del sistema....40
Facilidad de uso....40
Fiabilidad....40
Rendimiento....40
Soporte....41
Implementación....41
Interfaz....41
Operaciones....41
Empaquetamiento....41
Legales....42
4. Operaciones del sistema....43
Diagramas de secuencia del sistema....43
Subsistema Gestión Carta....43
Añadir elemento a la carta....43
Eliminar elemento de la carta....44
Modificar elemento de la carta....45
Subsistema Gestión Stock....46
Imprimir lista de productos a pedir....46
Notificar recepción de pedido del proveedor....47

Añadir ingrediente....48
Eliminar ingrediente....49
Modificar ingrediente....50
Notificar incidencia con ingrediente....51
Añadir bebida....52
Eliminar bebida....53
Modificar bebida....54
Notificar incidencia con bebida....55
Subsistema Gestión Pedidos....56
Cambia estado plato....56
Cambia estado bebida....57
Nuevo Pedido....58
Modifica Pedido....59
Subsistema de gestión de facturas....60
Pide Factura....60
Cambia Estado Factura....61
Subsistema Gestión Pedido Web....62
Nuevo Pedido....62
Modifica Pedido....63
Subsitema Gestión Estadísticas....64
Consulta Estadística....64
Identificación de operaciones del sistema....65
Añadir elemento a la carta....65
Eliminar elemento de la carta....65
Modificar elemento de la carta....65
Imprimir lista de productos a pedir....66
Notificar recepción de pedido del proveedor....66
Añadir ingrediente....66
Eliminar ingrediente....66
Modificar ingrediente....67
Notificar incidencia con ingrediente....67
Añadir Bebida....67
Añadir Bebida....67
Modificar bebida....68
Notificar incidencia con bebida....68
Alta de pedidos....68
Modificar pedidos....68

Cola de cocina....69
Cola de bar....69
Facturas....70
Alta de pedidos de hotel....70
Modificar pedidos....70
Consulta Estadística....71
Apéndice 1.0....72
Apéndice 1.1....73
Apéndice 1.2....74
Apéndice 1.3....75
Apéndice 2.0....76
Apéndice 2.1....77
Apéndice 2.2....78
Apéndice 2.3....79
Apéndice 3.0....80
Apéndice 3.1....81
Apéndice 3.2....82

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.

Este documento es una recopilación de los documentos de modelado de requisitos de las pasadas iteraciones. Para indicar a que Iteración pertenece usaremos el primer dígito de la versión, ya que siempre fue 1.

Versión	Fecha	Descripción
V1.0	15/03/10	Documento inicial
V1.1	16/03/10	Revisión de la especificación de requisitos, diagramas de secuencia del sistema y operaciones del sistema.
V1.2	21/03/10	Revisión del diagrama de casos de uso, especificación de los casos de uso y diagramas de secuencia
V1.3	15/04/10	Revisión de las especificaciones de los casos de uso y de los diagramas de secuencia del sistema.
V2.0	15/04/10	Se ha generado el documento de Modelado de Requisitos.
V2.1	22/04/10	Se han corregido los errores mencionados en el fichero de revisión y cambios en las colas de bar y cocina.
V2.2	24/04/10	Se han realizado numerosos cambios en las especificaciones, diagramas y operaciones.
V2.3	06/05/10	Se han realizado cambios en los diagramas de secuencia, especificaciones y operaciones.
V3.0	18/05/10	Se ha generado el documento de Modelado de Requisitos.
V3.1	18/05/10	Se generá la primera revisión del documento de Modelados de Requisitos
V3.2	20/05/10	Revisión de la especificación del subsistema Gestión Estadísticas, y unificación de los documentos de Modelado de requisitos de las tres iteraciones.

Primera iteración.

En esta primera iteración se abordaran los subsistemas:

- Subsistema de gestión de carta.
- Subsistema de gestión de Stock, que se divide en:
  - Subsistema de gestión de productos en Stock.
  - Subsistema de gestión de bebidas.

El subsistema de gestión de carta, se encargara de administrar todos los platos y bebidas disponibles para el cliente, así como indicar los productos de que esta compuesto cada elemento de la carta.

El subsistema de gestión de Stock se encarga de los productos disponibles y necesarios para elaborar los platos de la carta, se divide en platos y bebidas administradas por subsistemas independientes.

Segunda iteración.

En esta segunda iteración, hemos decidido abordar con todos los subsistemas que hemos podido identificar salvo el subsistema de gestión de pedidos. Con ello, los subsistemas que trataremos en esta iteración son:

- Subsistema de pedidos de cliente
- Subsistema de cola de platos
- Subsistema de cola de bebidas
- Subsistema de facturación

El subsistema de pedidos de cliente se encarga de que los clientes puedan realizar, modificar y anular pedidos.

El subsistema de cola de platos se encargará de gestionar los platos de pedidos que se deben cocinar en cocina, englobando los estados posibles por los que pasa un plato.

El subsistema de cola de bebidas gestionará las bebidas que deberán ser servidas.

El subsistema de facturación recibe una lista de elementos, generará una factura y la apuntará a la cola de facturas del metre, con el estado "En cola", o similar, al igual que en la base de datos. La factura deberá ser emitida por el metre y entregada por un camarero. Una vez pagada, se elimina la factura de la lista de pendientes y se almacena en la base de datos como factura cerrada.

Todos estos subsistemas por funcionalidad, quedan recogidos en un único subsistema, ya que todos se encargan de tratar con pedidos, con lo que el subsistema que trataremos quedará definido como **Subsistema de gestión de pedidos**.

Tercera iteración.

Para la tercera iteración solo nos queda por analizar, diseñar e implementar:

- Subsistema de pedidos de cliente para la interfaz web
- Subsistema de estadísticas

El subsistema de pedidos de cliente desde la interfaz web se encarga de que los clientes puedan realizar, modificar y anular pedidos desde su habitación del hotel.

El subsistema de estadísticas se encarga de mostrar al Cocinero Jefe una serie de estadísticas útiles para el negocio. El motivo de que sólo se le muestre al Cocinero Jefe es que este es el encargado de gestionar la Carta.

Así mismo, también se podrán presentar a los clientes unas ciertas valoraciones sobre los elementos que van a pedir.

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 que utilice en la elaboración de los platos del restaurante. También deberá informar de cuando llega un pedido del proveedor.

Esta al tanto de cambiar el estado de los platos que se reciben para ser preparados. Según si los platos están a la espera de ser cocinados, cocinándose o preparados el cocinero notificará su estado.

Para poder ofrecer una cocina al gusto de todos y mejorar el rendimiento podrá consultar las estadísticas de consumo de los diferentes elementos de pedido, podrá ver cuales son los que más se piden, cuales no, etc.

Metre

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

Controla el estado de las bebidas que se reciben para ser servidas. También se ocupa de modificar el estado de las facturas, cuando un cliente quiera una factura, le llegará una notificación al metre y este cambiará su estado cuando la imprima para ser entregada y cuando sea pagada.

Cliente

El cliente pide pedidos, los cuales contienen tanto bebidas como platos. Puede elegir modificarlos mientras no se estén atendiendo, es decir, que puede quitar y añadir elementos de un pedido. La facturación se cargará directamente en la cuenta de la habitación.

El cliente realiza el pedido desde la habitación de su hotel en un terminal habilitado para ello.

Identificación de los requerimientos funcionales

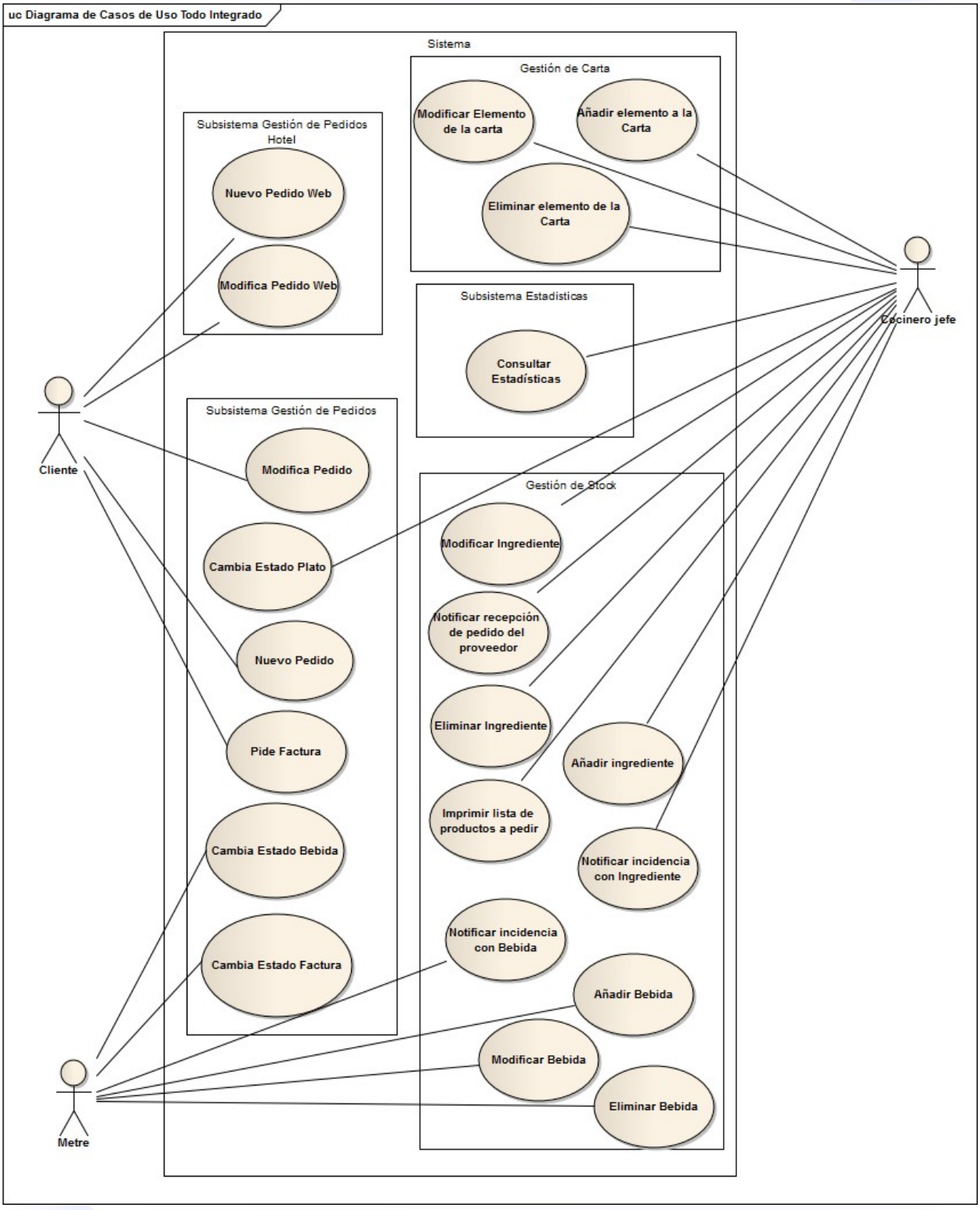
Requerimientos funcionales de toda la aplicación indicando en el primer dígito la iteración en la que se trato.

Identificador	Descripción
RF1.1	El sistema contempla a tres tipos de usuario: cliente, camarero jefe y metre.
RF1.2	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.
RF1.3	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.
RF1.4	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.
RF1.5	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.
RF1.6	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.
RF1.7	El cocinero jefe podrá modificar los datos de cualquier ingrediente.
RF1.8	El cocinero jefe también podrá eliminar ingredientes del sistema.
RF1.9	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.
RF1.10	El metre será el encargado de añadir, modificar y eliminar las bebidas del stock de productos.
RF1.11	Los datos necesarios para añadir una bebida serán los mismos que para un ingrediente. Sin embargo, en este caso también será necesario añadir la

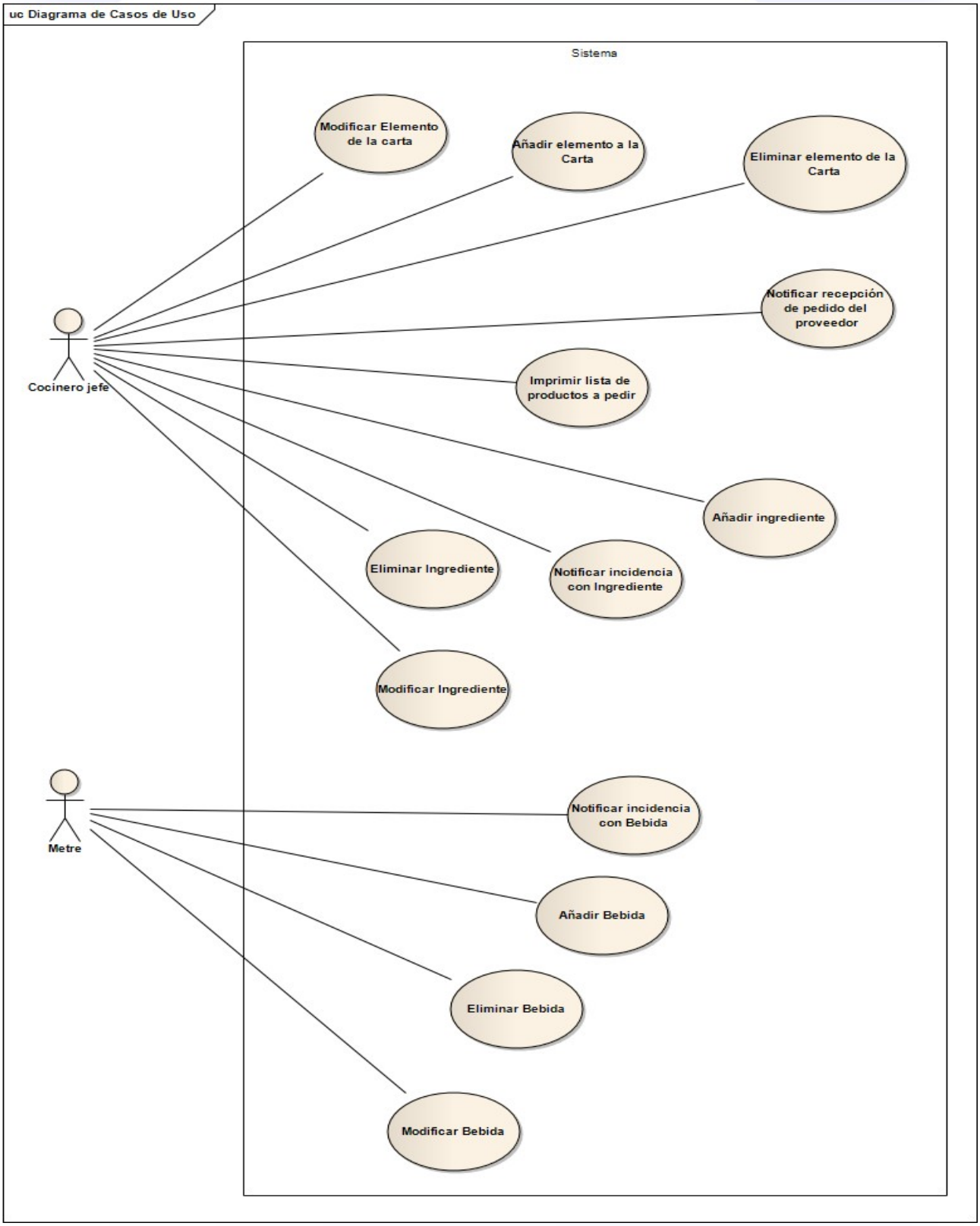
	cantidad de líquido que cabe en el envase en el que se encuentra dicha bebida.
RF1.12	El sistema deberá elaborar una lista de necesidades con las cantidades requeridas de cada uno de los productos que se deben comprar, a partir de los productos consumidos y del mínimo que debe haber de existencias. Además, el cocinero podrá imprimir esta lista cuando desee para que se pueda realizar el pedido.
RF1.13	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.
RF1.14	Si un producto llega a cantidad 0 en stock, el sistema invalidará los elementos de la carta que dependan de ese producto. Cuando dicho producto se reponga el sistema deberá rehabilitarlos.
RF2.2	El cocinero jefe ha de poder cambiar el estado de un pedido a medida que vaya cocinando los platos.
RF2.3	El metre ha de poder cambiar el estado de un pedido a medida que vaya teniendo listas las bebidas de este.
RF2.4	El sistema ha de notificar al metre la petición de una factura.
RF2.5	El metre ha de poder modificar el estado de una factura a medida que se complete la entrega y transacción.
RF2.6	Las facturas se imprimen primero luego se pagan.
RF2.7	Las facturas deben tener un formato especifico que contiene los elementos pedidos de una mesa mas la información del restaurante mas el total a pagar.
RF2.8	Un cliente debe poder elegir los elementos que desee para realizar un pedido.
RF2.9	Un cliente debe poder modificar y anular cualquiera de los elementos de su pedido, o todos ellos, siempre y cuando este no se haya comenzado a atender.
RF2.10	Un cliente debe poder pedir una factura en el momento que desee.
RF3.2	El cocinero jefe ha de poder consultar diferentes estadísticas del consumo de los elementos que se han pedido.
RF3.3	La facturación del pedido del cliente se cargará automáticamente en la cuenta de la habitación.
RF3.4	Un cliente debe poder elegir los elementos que desee para realizar un pedido.
RF3.5	Un cliente debe poder modificar y anular cualquiera de los elementos de su pedido, o todos ellos, siempre y cuando este no se haya comenzado a atender.
RF3.6	El cliente podrá visualizar a la hora de hacer un pedido la valoración del elemento que desea pedir.

Diagrama de Casos de Uso

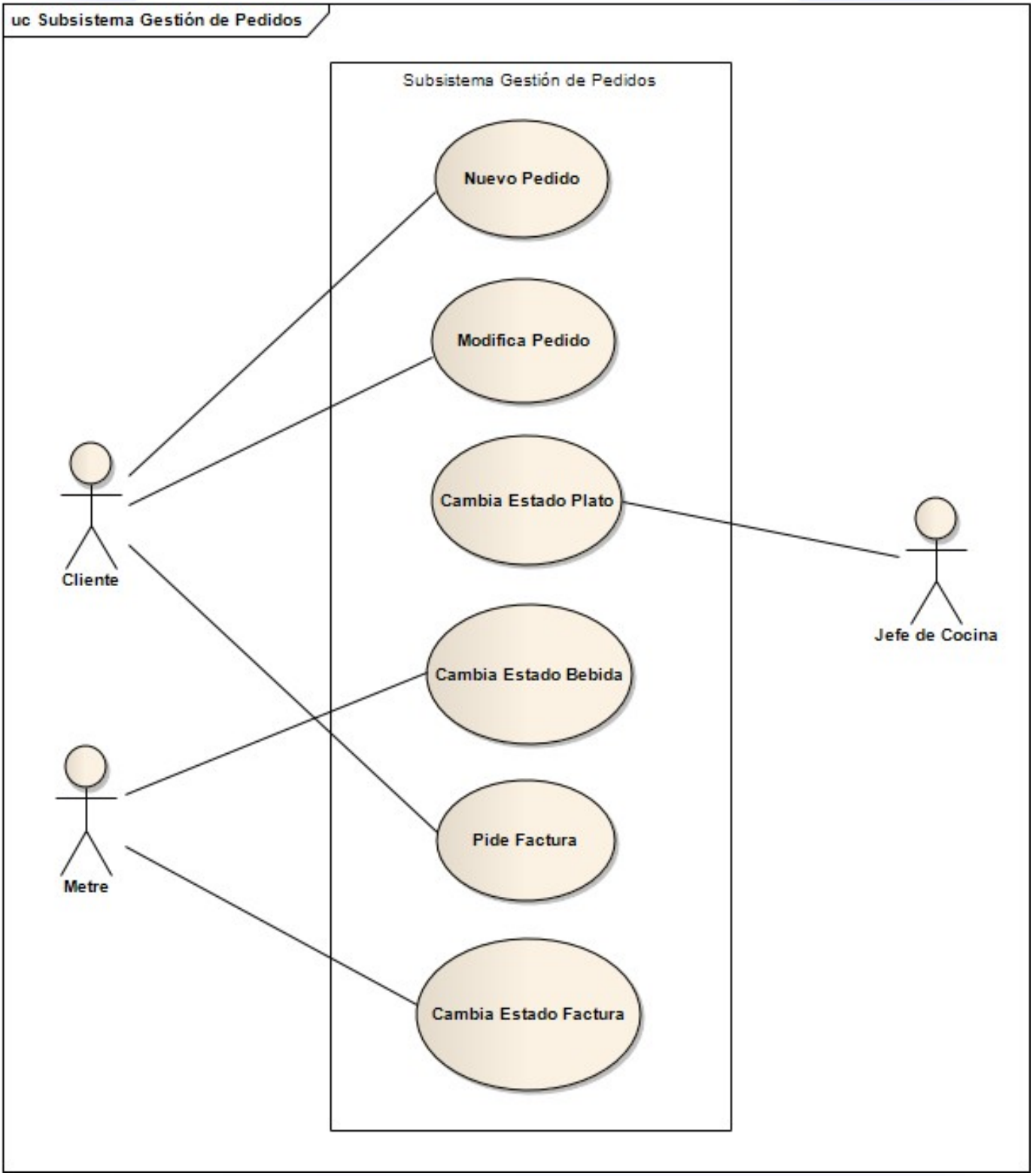
Todo el sistema



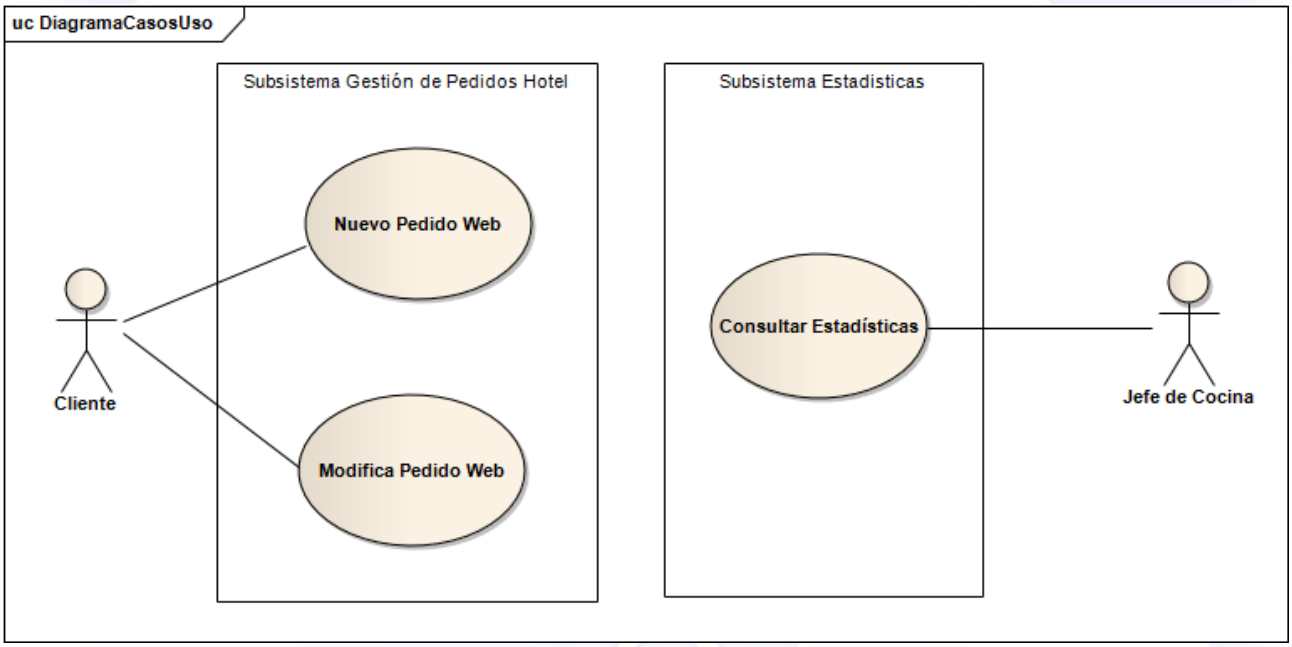
Primera Iteración



Segunda iteración



Tercera Iteración



Descripción de los casos de uso

Gestión Carta

Añadir elemento a la carta

Nombre del caso:	Añadir elemento a la carta
Resumen:	El cocinero jefe desea añadir un a la carta del restaurante. Introduce toda la información correspondiente al nuevo elemento y por último selecciona los ingredientes asociados a este con sus cantidades correspondientes.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha añadido un elemento nuevo a la carta
Curso normal:	1. El caso de uso comienza cuando el usuario desea añadir un elemento a la carta 2. El usuario especifica la sección en la que va a incluir el nuevo elemento. 3. El sistema muestra una lista productos dependiendo de la sección escogida. 4. El usuario rellena los datos correspondientes al elemento: 4.1.1. Introduce el nombre 4.1.2. Introduce la descripción 4.1.3. Introduce la foto 4.1.4. Introduce el precio 4.1.5. Introduce el número de divisiones del elemento 4.1.6. Selecciona la sección 4.1.6.1. Si la sección asignada no es la de bebidas 4.1.6.1.1 El usuario especifica el tiempo de elaboración del plato 4.2. Mientras el usuario quiera asociar productos al elemento: 4.2.1. El usuario selecciona un producto de la lista e introduce la cantidad deseada 4.3. El usuario confirma los datos 4.4. El sistema comprueba que los datos introducidos sean correctos 4.5. Los datos introducidos son correctos. El sistema genera un código asociado al elemento y registra todos los datos 4.6. El sistema confirma que se ha añadido un nuevo elemento a la carta satisfactoriamente
Cursos	4.2.1a. El usuario se ha equivocado al seleccionar un producto 4.2.1.1. El usuario elimina el producto deseado

alternativos:	4.2.1b. El usuario se ha equivocado al introducir la cantidad de un ingrediente 4.2.1.1. El usuario selecciona el producto de la lista e introduce la nueva cantidad 4.5. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3 *. El usuario cancela la operación
Observaciones:	
Requisitos no funcionales específicos:	- Las secciones asignables a un elemento de la carta serán: **Entrantes**, **Carnes**, **Pescados**, **Bebidas** o **Postres** - 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 **- La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb**

Eliminar elemento a la carta

Nombre del caso:	Eliminar elemento de la carta
Resumen:	El cocinero jefe desea eliminar un elemento de la carta del restaurante. Le basta tan solo con seleccionar el elemento que desea borrar y confirmarlo
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha eliminado un elemento de la carta
Curso normal:	1. El caso de uso comienza cuando el usuario desea eliminar un elemento de la carta 2. El sistema muestra una lista con los elementos de la carta registrados 3. El usuario selecciona el elemento de la carta que desea eliminar 4. El sistema muestra la información asociada al elemento y solicita la confirmación de la operación 5. El usuario confirma la operación 6. El sistema confirma que el elemento ha sido eliminado de la carta satisfactoriamente
Cursos alternativos:	5. El usuario no confirma la operación. Se vuelve al paso 2 *. El usuario cancela la operación
Observaciones:	
Requisitos no funcionales:	

Modificar elemento de la carta

Nombre del caso:	Modificar elemento de la carta
Resumen:	El cocinero jefe desea modificar un elemento de la carta del restaurante. Selecciona el elemento y modifica el o los datos deseados.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se han modificado uno o más datos de un elemento de la carta
Curso normal:	1. El caso de uso comienza cuando el usuario desea modificar un elemento de la carta 2. El sistema muestra una lista con los elementos de la carta registrados 3. El usuario selecciona un elemento de la lista 4. El sistema muestra la información asociada a ese elemento 5. Mientras el usuario desee modificar datos del elemento: 5.1 El usuario modifica el dato deseado 6. El usuario finaliza la modificación de los datos 7. El sistema solicita la confirmación de la modificación de los datos 8. El usuario confirma la operación 9. El sistema comprueba que los datos introducidos son correctos. 10. Los datos introducidos son correctos. El sistema registra los nuevos cambios y confirma que el elemento ha sido modificado con éxito
Cursos alternativos:	10. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 4 *. El usuario cancela la operación
Observaciones:	
Requisitos no funcionales:	

Imprimir lista de productos a pedir

Nombre del caso:	Imprimir lista de productos a pedir
Resumen:	El cocinero jefe desea imprimir una lista de los productos que están bajo mínimos en stock para poder hacer un pedido al proveedor
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha imprimido la lista de productos a pedir • Se ha guardado una copia de la lista en el sistema
Curso normal:	1. El caso de uso comienza cuando el usuario quiere imprimir la lista de productos y cantidades del pedido 2. El sistema obtiene la lista de productos que están bajo mínimos en stock y las cantidades necesarias de cada uno para alcanzar su máximo. 3. El sistema imprime dicha lista 4. El sistema guarda una copia de la lista 5. El sistema muestra la información asociada al pedido
Cursos alternativos:	
Observaciones:	
Requisitos no funcionales específicos:	La lista de productos estará dividida en dos partes: por un lado los ingredientes y por otro lado las bebidas

Notificar recepción de pedido del proveedor

Nombre del caso:	Notificar recepción de pedido del proveedor
Resumen:	El cocinero jefe informa de que ha llegado el pedido realizado al proveedor. Las cantidades de los productos pedidos se actualizan automáticamente
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha modificado la cantidad actual en stock de uno o más productos
Curso normal:	1. El caso de uso comienza cuando el cocinero jefe informa de que ha llegado el pedido del proveedor. 2. El sistema obtiene los datos del pedido y muestra los elementos de la carta inválidados que quedarán habilitados tras la actualización de sus cantidades en caso de que los haya. 3. El sistema actualiza las cantidades de los productos especificados en el pedido 4. El sistema habilita los elementos de la carta correspondientes. 5. El sistema devuelve la información de los productos actualizados y de los elementos habilitados.
Cursos alternativos:	
Observaciones:	El sistema utiliza la copia que posee del pedido para actualizar las cantidades de los productos correspondientes
Requisitos no funcionales específicos:	

Añadir ingrediente

Nombre del caso:	Añadir ingrediente
Resumen:	El cocinero jefe desea añadir un ingrediente al sistema. Introduce los datos correspondientes a dicho ingrediente, se comprueban si son correctos y se confirma la operación.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha registrado un ingrediente en el sistema
Curso normal:	1. El caso de uso comienza cuando el cocinero jefe desea añadir un ingrediente. 2. El usuario introduce los datos correspondientes al nuevo ingrediente: 2.1.1 Introduce el nombre 2.1.2 Introduce el límite mínimo en stock. 2.1.3 Introduce el máximo en stock. 2.1.4 Introduce la cantidad actual en stock 2.1.5 Introduce una foto. 3. El usuario confirma los datos. 4. El sistema comprueba si los datos introducidos son correctos 5. Los datos introducidos son correctos. El sistema genera un código de producto, registra todos los datos y confirma que ha sido añadido satisfactoriamente.
Cursos alternativos:	5. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 2 *. El usuario cancela la operación.
Observaciones:	
Requisitos no funcionales específicos:	- Tanto los límites como la cantidad actual se deberán dar en gramos. - La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb

Eliminar ingrediente

Nombre del caso:	Eliminar ingrediente
Resumen:	El cocinero jefe desea eliminar un ingrediente de nuestro sistema. Para ello lo selecciona de una lista de ingredientes y confirma su eliminación. Una vez hecho esto el sistema lo elimina.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha eliminado un ingrediente del sistema.
Curso normal:	1. El caso de uso comienza cuando el cocinero jefe desea eliminar un ingrediente. 2. El usuario selecciona el ingrediente que quiere eliminar. 3. El sistema obtiene información detallada del ingrediente que deseamos eliminar y la muestra al usuario. Además muestra una lista con los elementos de la carta que se verán afectados tras la eliminación del ingrediente. 4. El usuario confirma la eliminación. 5. El sistema elimina el ingrediente e invalida los elementos de la carta que cuentan con ese ingrediente. Se informa de ello al usuario.
Cursos alternativos:	4. El usuario no confirma la operación. Se vuelve al paso 2. *. El usuario cancela la operación.
Observaciones:	
Requisitos no funcionales específicos:	

Modificar ingrediente

Nombre del caso:	Modificar ingrediente
Resumen:	El cocinero jefe quiere modificar los datos de un determinado ingrediente. Una vez que se le muestran los datos, cambia los parámetros que quiera y una vez confirmados y comprobados estos se registran en el sistema.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha registrado uno o más cambios en los detalles de un ingrediente.
Curso normal:	1. El caso de uso comienza cuando el cocinero jefe desea modificar un ingrediente. 2. El usuario selecciona el ingrediente que quiere modificar. 3. El sistema muestra los datos asociados a ese ingrediente. 4. Mientras el usuario quiera modificar parámetros: 4.1. El usuario modifica el parámetro deseado. 5. El usuario confirma los cambios. 6. El sistema comprueba si los datos introducidos son correctos 7. Los datos introducidos son correctos. El sistema registra los cambios e informa que se han realizado satisfactoriamente.
Cursos alternativos:	5. El usuario no confirma los cambios. Se vuelve al paso 2 7. El sistema informa de que los datos introducidos son incorrectos. Se vuelve al paso 2. *. El usuario cancela la operación.
Observaciones:	
Requisitos no funcionales específicos:	

Notificar incidencia con Ingrediente

Nombre del caso:	Notificar incidencia con Ingrediente
Resumen:	El cocinero jefe desea notificar una incidencia ocurrida con un ingrediente. Para ello selecciona el ingrediente en cuestión e introduce la cantidad de este que se ha visto afectada. Por último especifica el tipo de incidencia.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha registrado una incidencia con un ingrediente
Curso normal:	1. El caso de uso comienza cuando el usuario desea notificar una incidencia ocurrida con un ingrediente 2. El usuario selecciona el ingrediente sobre el que desea notificar la incidencia 3. El usuario introduce la cantidad de producto afectado y especifica el motivo de la incidencia 4. El sistema comprueba la cantidad que queda del ingrediente en stock 4.1. Si la cantidad actual en stock del ingrediente afectado se encuentra por debajo de su mínimo 4.1.1 El sistema informa de que la cantidad de ese producto en stock se encuentra por debajo de su mínimo 5. El sistema comprueba que los datos introducidos son correctos. 6. Los datos introducidos son correctos. El sistema genera un código asociado a la incidencia y registra la incidencia 7. El sistema resta del stock la cantidad del ingrediente especificado por el usuario e invalida los elementos con el ingrediente asociado en caso de que sea necesario. 8. El sistema informa de que la incidencia se ha registrado correctamente
Cursos alternativos:	6. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 2 4.1. Si el ingrediente se ha agotado completamente 4.1.1. El sistema busca los elementos de la carta que usen ese ingrediente y los invalida. Se informa de ello al usuario *. El usuario cancela la operación
Observaciones:	
Requisitos no funcionales específicos:	

Añadir bebida

Nombre del caso:	Añadir bebida
Resumen:	El metre desea añadir una bebida al sistema. Introduce los datos correspondientes a la bebida, se comprueban si son correctos y se confirma la operación.
Dependencias:	
Actores:	Metre
Precondiciones:	
Postcondiciones:	• Se ha registrado una nueva bebida en el sistema
Curso normal:	1. El caso de uso comienza cuando el cocinero jefe desea añadir una bebida. 2. El usuario introduce los datos correspondientes a la bebida: 2.1. Introduce el nombre 2.2. Introduce el límite mínimo en stock. 2.3. Introduce el máximo en stock. 2.4. Introduce la cantidad actual en stock 2.5. Introduce una foto. 3. El usuario confirma los datos. 4. El sistema comprueba si los datos introducidos son correctos 5. Los datos introducidos son correctos. El sistema genera un código de producto, registra todos los datos y confirma que se ha añadido la bebida satisfactoriamente.
Cursos alternativos:	5. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 3.1 *. El usuario cancela la operación.
Observaciones:	
Requisitos no funcionales específicos:	- Los límites mínimo y máximo vendrán en número de unidades de envase - La foto deberá estar en formato .jpeg y no sobrepasar los 200 kb

Eliminar bebida

Nombre del caso:	Eliminar bebida
Resumen:	El metre desea eliminar una bebida de nuestro sistema. Para ello la selecciona de una lista de bebidas, y confirma su eliminación. Una vez hecho esto el sistema la elimina.
Dependencias:	
Actores:	Metre.
Precondiciones:	
Postcondiciones:	• Se ha eliminado una bebida del sistema.
Curso normal:	1. El caso de uso comienza cuando el metre desea eliminar una bebida. 2. El usuario selecciona la bebida que quiere eliminar. 3. El sistema obtiene información detallada de la bebida que deseamos eliminar y la muestra al usuario. Además muestra una lista con los elementos de la carta que se verán afectados tras la eliminación de la bebida. 4. El usuario confirma la eliminación. 5. El sistema invalida los elementos de la carta afectados. 6. El sistema elimina la bebida e informa de ello al usuario.
Cursos alternativos:	4. El usuario no confirma la operación. Se vuelve al paso 2. *. El usuario cancela la operación.
Observaciones:	
Requisitos no funcionales específicos:	

Modificar bebida

Nombre del caso:	Modificar bebida
Resumen:	El metre quiere modificar los datos de una determinada bebida. Selecciona la bebida que desea modificar, cambia los parámetros que quiera y una vez confirmados y comprobados estos se registran en el sistema.
Dependencias:	
Actores:	Metre
Precondiciones:	
Postcondiciones:	• Se ha registrado uno o más cambios en los detalles de una bebida.
Curso normal:	1. El caso de uso comienza cuando el metre desea modificar una bebida. 2. El usuario selecciona la bebida que quiere modificar. 3. El sistema muestra los datos asociados a dicha bebida. 4. Mientras el usuario quiera modificar parámetros: 4.1. El usuario modifica el parámetro deseado. 5. El usuario confirma los cambios. 6. El sistema comprueba si los datos introducidos son correctos 7. Los datos introducidos son correctos. El sistema registra los cambios e informa que se han realizado satisfactoriamente.
Cursos alternativos:	5. El usuario no confirma los cambios. Se vuelve al paso 2 6. El sistema informa de que los datos introducidos son incorrectos. Se vuelve al paso 2. *. El usuario cancela la operación.
Observaciones:	

Notificar incidencia con bebida

Nombre del caso:	Notificar incidencia con Bebida
Resumen:	El cocinero jefe desea notificar una incidencia ocurrida con una bebida. Para ello selecciona la bebida en cuestión e introduce la cantidad de esta que se ha visto afectada. Por último especifica el tipo de incidencia.
Dependencias:	
Actores:	Cocinero jefe
Precondiciones:	
Postcondiciones:	• Se ha registrado una incidencia con una bebida
Curso normal:	1. El caso de uso comienza cuando el usuario desea notificar una incidencia ocurrida con una bebida 2. El usuario selecciona la bebida sobre el que desea notificar la incidencia 3. El usuario introduce la cantidad de producto afectado y especifica el motivo de la incidencia 4. El sistema comprueba la cantidad que queda de la bebida en stock 4.1. Si la cantidad actual en stock de la bebida afectada se encuentra por debajo de su mínimo 4.1.1 El sistema informa de que la cantidad de ese producto en stock se encuentra por debajo de su mínimo 5. El sistema comprueba que los datos introducidos son correctos. 6. Los datos introducidos son correctos. El sistema genera un código asociado a la incidencia y registra la incidencia 7. El sistema resta del stock la cantidad de bebida especificada por el usuario e invalida los elementos con la bebida asociado en caso de que sea necesario. 8. El sistema informa de que la incidencia se ha registrado correctamente
Cursos alternativos:	6. El sistema informa de que los datos introducidos no son correctos. Se vuelve al paso 2 4.1. Si la bebida se ha agotado completamente 4.1.1. El sistema busca los elementos de la carta que usan esta bebida y los invalida. Se informa de ello al usuario *. El usuario cancela la operación
Observaciones:	
Requisitos no funcionales:	

Cambia estado plato

Nombre del caso:	Cambia Estado Plato
Resumen:	Este caso de uso permite al jefe de cocina modificar el estado del siguiente plato de la cola de platos no atendidos, o bien el estado de un plato cualquiera de la lista de platos en preparación.
Dependencias	
Actores:	Jefe de cocina.
Precondiciones:	• Existe algún plato, en la cola de platos no atendidos o en la lista de platos en preparación.
Postcondiciones:	• Se cambió el estado de un plato. • Se cambió el estado del pedido.
Curso normal:	1 El usuario selecciona un plato del siguiente pedido que le muestra el sistema en los pedidos en cola, o bien de cualquier plato en la lista de platos preparándose. 2 Si el plato se encuentra en estado "En cola", el sistema cambia su estado a "Preparándose", cambia el estado del pedido a "Bloqueado" y reduce la cantidad de productos del plato de stock. 3 Si el plato se encuentra en estado "Preparándose", el sistema comprueba si era el último plato por cambiar de estado: 3.1 Si el plato era el último que quedaba del pedido por cambiar de estado, el sistema pide confirmación al usuario para cerrar el pedido. 3.2 El usuario confirma la acción. 3.3 El sistema cambia el estado del plato a "Preparado".
Cursos alternativos:	3.1a No era el último plato que quedaba del pedido por cambiar de estado, se cambia el estado del plato a "Preparado". 3.2b El usuario cancela que ha finalizado el servicio del pedido. 3.3b El sistema no modifica el estado del plato, termina el caso.
Observaciones:	
Requisitos no funcionales específicos:	

Especificado por Sergio Rodríguez Lumley

Cambia estado bebida

Nombre del caso:	Cambia Estado Bebida
Resumen:	Este caso de uso permite al metre cambiar el estado de la siguiente bebida de la cola de bebidas no atendidas.
Dependencias	
Actores:	Metre.
Precondiciones:	• Existe alguna bebida en la cola de bebidas no atendidas.
Postcondiciones:	• Se cambió el estado de una bebida. • Se cambió el estado del pedido.
Curso normal:	1 El metre selecciona una bebida del siguiente pedido de la cola de bebidas no atendidas que le muestra el sistema. 2 El sistema cambia el estado del pedido a “Bloqueado” y comprueba si es la última bebida para cambiar de estado 2.1 Si la bebida era la última que quedaba del pedido por cambiar de estado, el sistema pide confirmación al usuario para cerrar el pedido de bebidas. 2.2 El usuario confirma que ha finalizado el servicio del pedido. 2.3 El sistema cambia el estado de la bebida a “Preparado”.
Cursos alternativos:	2.1a No era la última bebida que quedaba del pedido por cambiar de estado, se cambia el estado de la bebida a “Preparado”. 2.2b El usuario cancela que ha finalizado el servicio del pedido. 2.3b El sistema no modifica el estado de la bebida, termina el caso.
Observaciones:	
Requisitos no funcionales específicos:	

Especificado por Sergio Rodríguez Lumley

Pedidos de cliente

Nuevo pedido

Nombre del caso:	Nuevo Pedido
Resumen:	Este caso de uso permite al cliente realizar un nuevo pedido.
Dependencias	
Actores:	Cliente.
Precondiciones:	• Existe en la carta algún plato o bebida disponible.
Postcondiciones:	• Se creó un nuevo pedido asociado a la mesa.
Curso normal:	1 El cliente inicia el caso de uso. 2 El sistema le muestra al cliente la carta. 3 Mientras el cliente quiera modificar elementos del pedido: 3.1 Si añade un nuevo elemento: 3.1.1 El sistema añade el elemento del pedido. 3.1.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 3.2 Si elimina un elemento antes añadido: 3.2.1 El sistema elimina el elemento del pedido. 3.2.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 4 El cliente confirma el pedido. 5 El sistema registra el pedido y avisa al cliente que el pedido ha sido realizado con éxito.
Cursos alternativos:	4.a El cliente cancela el pedido. Se cancela el caso de uso.
Observaciones:	Cuando se confirma el pedido, los elementos de este deben encolarse en la cola de cocina y en la cola de bar.
Requisitos no funcionales específicos:	

Especificado por Adrián Víctor Pérez Lopera

Modifica pedido

Nombre del caso:	Modifica Pedido
Resumen:	Este caso de uso permite al cliente modificar un pedido en el cual ninguno de sus elementos están en preparación.
Dependencias	
Actores:	Cliente.
Precondiciones:	
Postcondiciones:	• Se modificó un pedido asociado a la mesa.
Curso normal:	1 El cliente inicia el caso de uso. 2 El sistema le muestra al cliente los pedidos que hay asociados a su mesa y que pueden modificarse. 3 El cliente selecciona el pedido. 4 El sistema muestra al cliente la carta y los elementos actuales que tiene su pedido. 5 Mientras el cliente quiera modificar elementos del pedido: 5.1 Si añade un nuevo elemento: 5.1.1 El sistema añade el elemento del pedido. 5.1.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 5.2 Si elimina un elemento antes añadido: 5.2.1 El sistema elimina el elemento del pedido. 5.2.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 6 El cliente confirma la modificación. 7 El sistema registra los cambios y avisa al cliente que el pedido ha sido modificado con éxito.
Cursos alternativos:	6.a El cliente cancela la modificación. Se cancela el caso de uso.
Observaciones:	Cuando se modifica un pedido, los elementos de este pasan a estar en último lugar tanto en la cola de cocina como en la de bar.
Requisitos no funcionales específicos:	

Especificado por Adrián Víctor Pérez Lopera

Cambia estado Factura

Nombre del caso:	Cambia Estado Factura
Resumen:	El Metre utiliza este caso de uso para cambiar estado de las facturas.
Dependencias:	
Actores:	Metre
Precondiciones:	
Postcondiciones:	
Curso normal:	1 El caso de uso comienza cuando el usuario quiere actualizar el estado de una o varias facturas. 2 El usuario selecciona: 2.1 Cambiar estado de factura(s) de una mesa: 2.1.1 El usuario selecciona la mesa. 2.1.1.1 El usuario selecciona la factura: 2.1.1.1.1 Si se quiere imprimir la factura seleccionada: 2.1.1.1.1.1 El sistema imprime la factura. 2.1.1.1.2 Si se quiere pagar la factura: 2.1.1.1.2.1 Si hay elementos en la mesa que no se han facturado el usuario imprime una nueva factura que contiene todos los elementos. 2.1.1.1.2.2 el usuario confirma el pago.
Cursos alternativos:	2. El usuario sale del caso del uso.
Observaciones:	Las facturas van a ser ordenadas en la cola de factura por mesas, un cliente puede hacer varios pedidos.
Requisitos no funcionales específicos:	

Especificado por Nabil Sabeg

Pide Factura

Nombre del caso:	Pide Factura
Resumen:	El usuario pide que se imprime su factura.
Dependencias:	
Actores:	Cliente
Precondiciones:	
Postcondiciones:	Cola de facturas pendiente de imprimir actualizada.
Curso normal:	1 El caso de uso comienza cuando el cliente quiere la factura, 2 El sistema actualiza la cola de facturas.
Cursos alternativos:	
Observaciones:	La información que debe tener la factura es los pedidos con sus elementos y el código de mesa correspondiente.
Requisitos no funcionales específicos:	

Especificado por Nabil Sabeg

Nuevo pedido web

Nombre del caso:	Nuevo Pedido Web
Resumen:	Este caso de uso permite al cliente realizar un nuevo pedido desde la habitación del hotel.
Dependencias	
Actores:	Cliente.
Precondiciones:	• Existe en la carta algún plato o bebida disponible.
Postcondiciones:	• Se creó un nuevo pedido asociado a la habitación.
Curso normal:	1 El cliente inicia el caso de uso. 2 El sistema le muestra al cliente la carta. 3 Mientras el cliente quiera modificar elementos del pedido: 3.1 Si añade un nuevo elemento: 3.1.1 El sistema añade el elemento del pedido. 3.1.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 3.2 Si elimina un elemento antes añadido: 3.2.1 El sistema elimina el elemento del pedido. 3.2.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 4 El cliente confirma el pedido. 5 El sistema registra el pedido y avisa al cliente que el pedido ha sido realizado con éxito.
Cursos alternativos:	4.a El cliente cancela el pedido. Se cancela el caso de uso. 5.a El pedido no contiene ningún elemento. Se cancela el pedido. Se cancela el caso de uso.
Observaciones:	Cuando se confirma el pedido, los elementos de este deben encolarse en la cola de cocina y en la cola de bar.
Requisitos no funcionales específicos:	

Especificado por Carlos Salas Morales

Modifica pedido web

Nombre del caso:	Modifica Pedido Web
Resumen:	Este caso de uso permite al cliente modificar un pedido web en el cual ninguno de sus elementos están en preparación.
Dependencias	
Actores:	Cliente.
Precondiciones:	• Existe un pedido asociado a la habitación. • Existe un elemento del pedido, el cual, aún no empezó a preparase.
Postcondiciones:	• Se modificó un pedido asociado a la habitación.
Curso normal:	1 El cliente inicia el caso de uso. 2 El sistema le muestra al cliente los pedidos que hay asociados a su habitación y que pueden modificarse. 3 El cliente selecciona el pedido. 4 El sistema muestra al cliente la carta y los elementos actuales que tiene su pedido. 5 Mientras el cliente quiera modificar elementos del pedido: 5.1 Si añade un nuevo elemento: 5.1.1 El sistema añade el elemento del pedido. 5.1.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 5.2 Si elimina un elemento antes añadido: 5.2.1 El sistema elimina el elemento del pedido. 5.2.2 El sistema muestra al cliente los elementos actuales que tiene su pedido. 6 El cliente confirma la modificación. 7 El sistema registra los cambios y avisa al cliente que el pedido ha sido modificado con éxito.
Cursos alternativos:	6.a El cliente cancela la modificación. Se cancela el caso de uso. 7.a El pedido no tiene ningún elemento. Se cancela el pedido. Se cancela el caso de uso.
Observaciones:	Cuando se modifica un pedido, los elementos de este pasan a estar en último lugar tanto en la cola de cocina como en la de bar.
Requisitos no funcionales específicos:	

Especificado por Carlos Salas Morales

Consulta Estadística

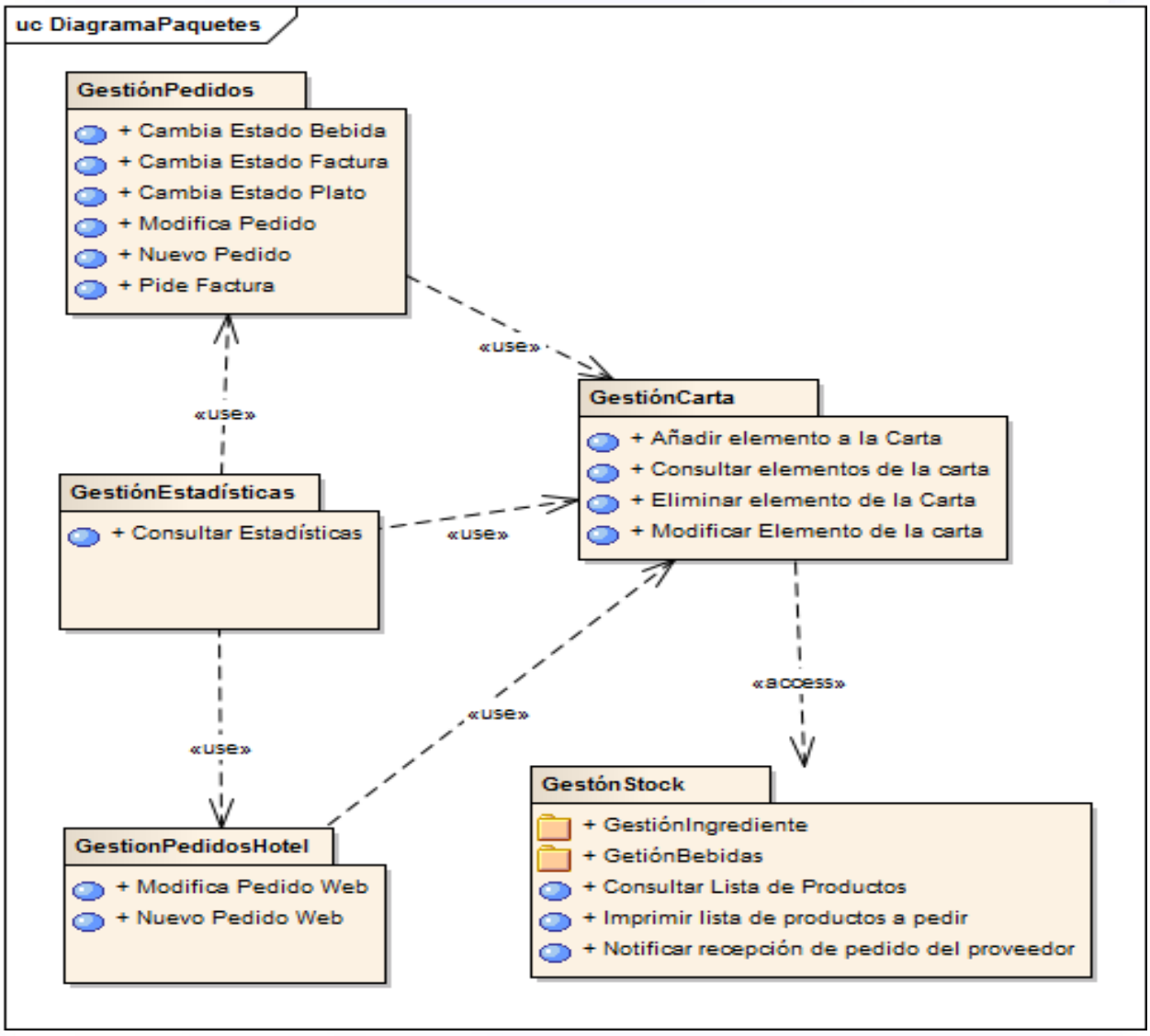
Nombre del caso:	Consulta Estadística
Resumen:	Muestra al usuario una serie de estadísticas predefinidas sobre la venta de elementos.
Dependencias:	
Actores:	Jefe de Cocina
Precondiciones:	
Postcondiciones:	
Curso normal:	1 El actor inicia el caso de uso. 2 El sistema da a elegir una serie de estadísticas predefinidas. 3 El actor elige la estadística que desea ver. 4 El sistema muestra un cuadro con la información solicitada.
Cursos alternativos:	3a. El actor cancela la acción. Se termina el caso de uso.
Observaciones:	
Requisitos no funcionales específicos:	- En la información mostrada, debería incluirse una gráfica.

Especificado por Ángel Luis García Sánchez

Identificación de los subsistemas funcionales

- Subsistema Gestión Pedidos.
- Subsistema Gestión Carta.
- Subsistema Gestión Stock.
- Subsistema Gestión Pedidos Hotel.
- Subsistema Gestión Estadísticas.

Diagrama de paquetes



Especificado por Ángel García Sánchez

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.
- El sistema debe sincronizar las peticiones simultaneas de los cliente.

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.
- El gestor de la base de datos debe ser compatible con estos equipos.

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

- Los administradores interaccionarán con el sistema cuando surja algún cambio imprevisto que el sistema no sea capaz de detectar.

Empaquetamiento

- La base de datos debe ser instalada y configurada para su uso por parte del sistema en el local.
- El sistema se ejecuta directamente usando la maquina virtual de java en caso de los terminales del restaurante , o bien usando pagina web en caso de terminales disponibles en la habitaciones del hotel.

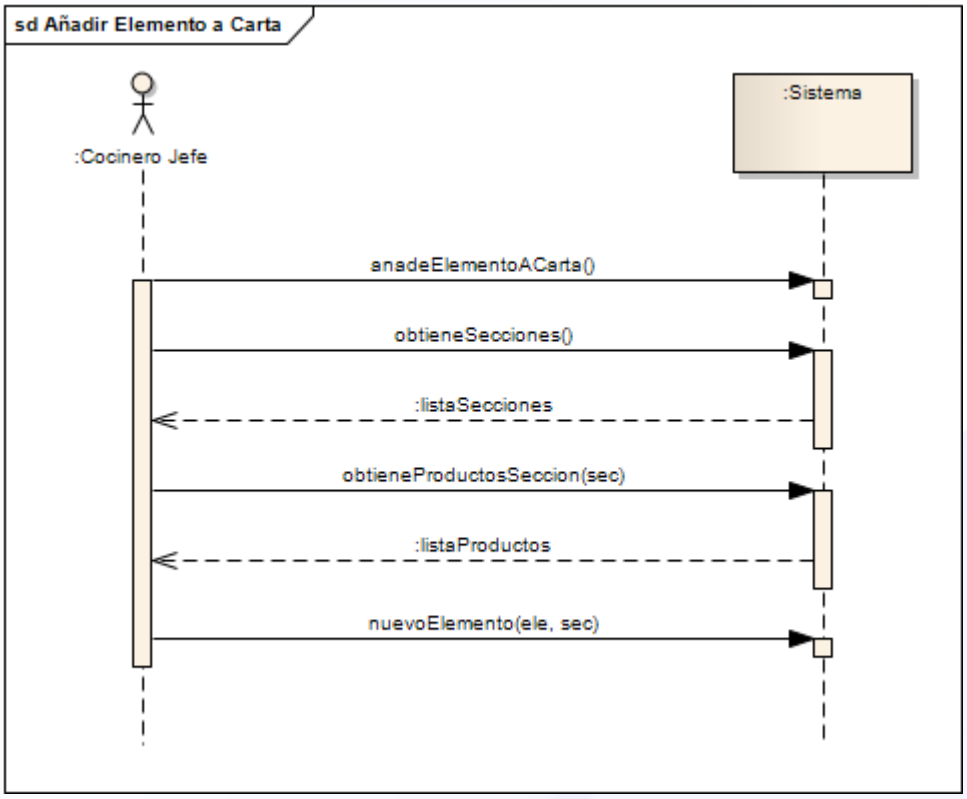
- El sistema debe cumplir las disposiciones recogidas en la Ley Orgánica de Datos Personales y en el Reglamento de medidas de seguridad.

Realizado por Daniel Guerrero Martínez

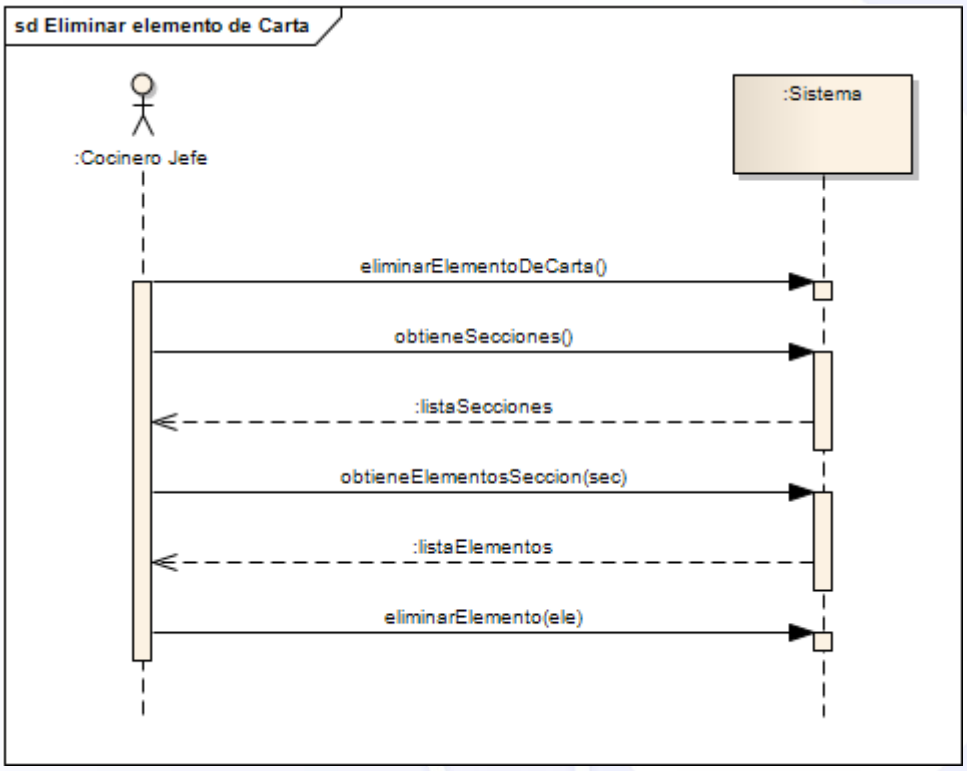
Diagramas de secuencia del sistema

Subsistema Gestión Carta

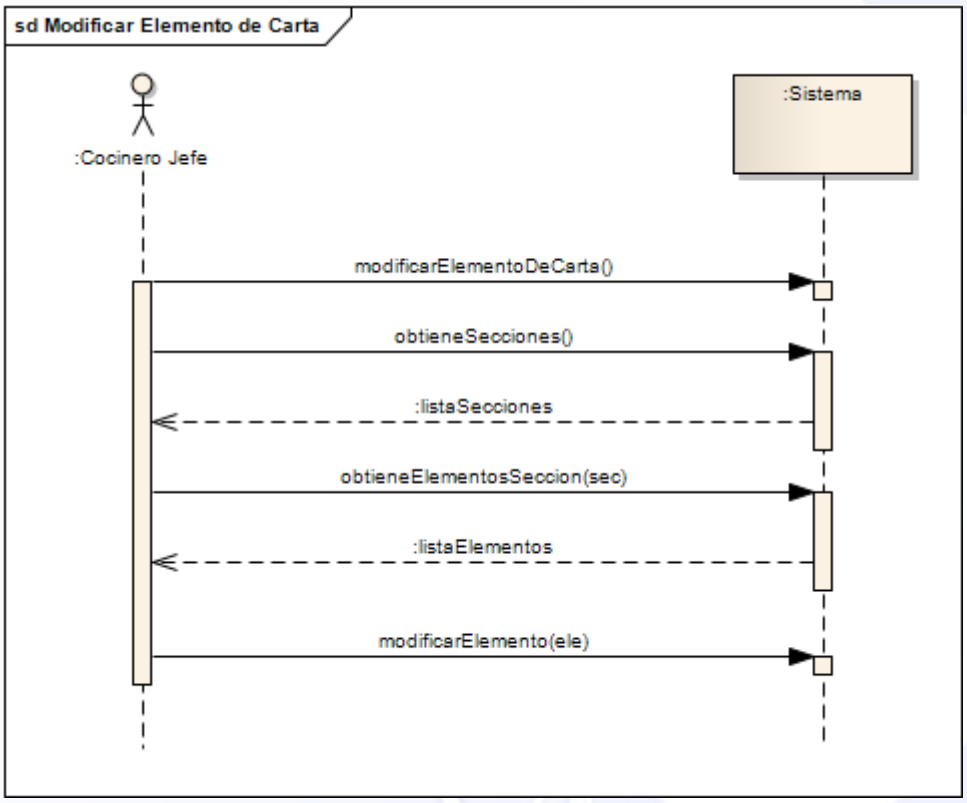
Añadir elemento a la carta



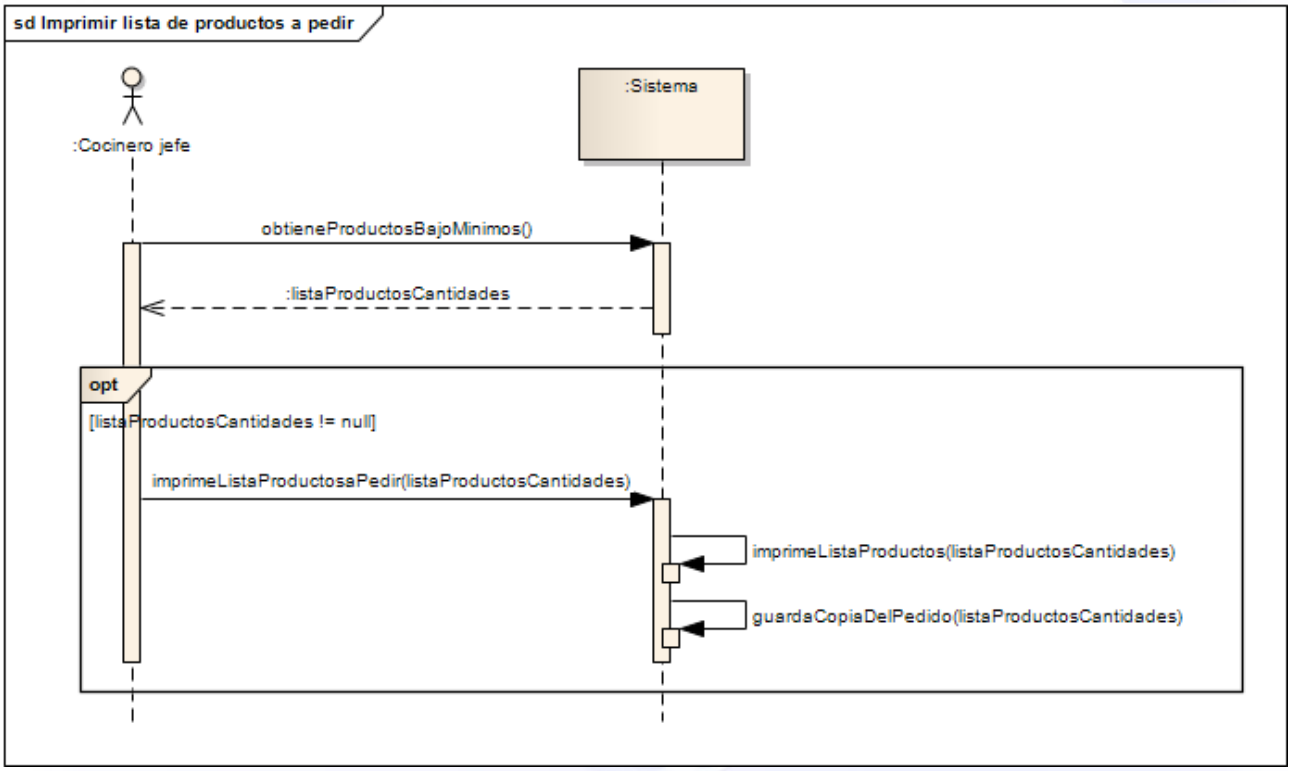
Eliminar elemento de la carta



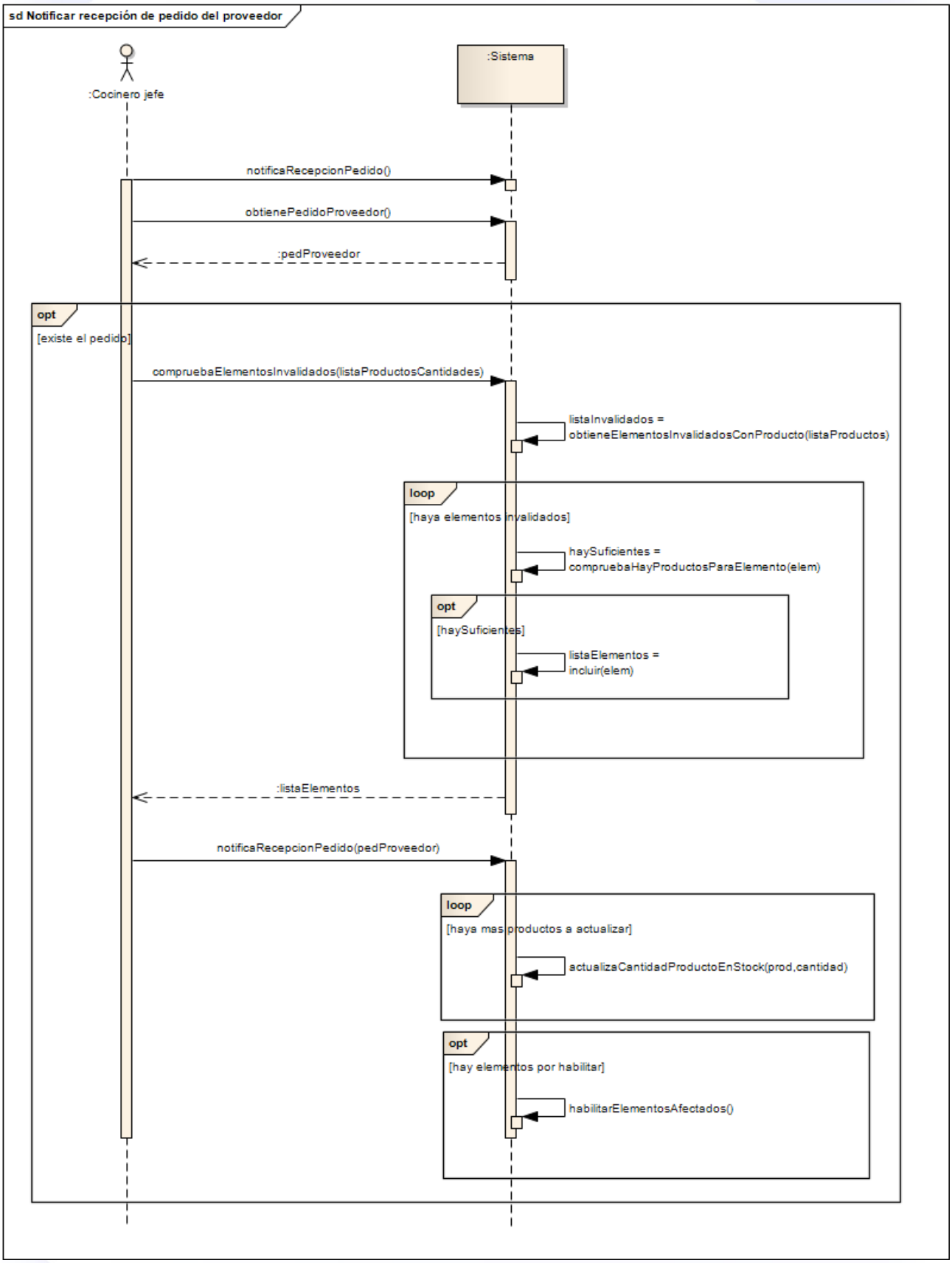
Modificar elemento de la carta



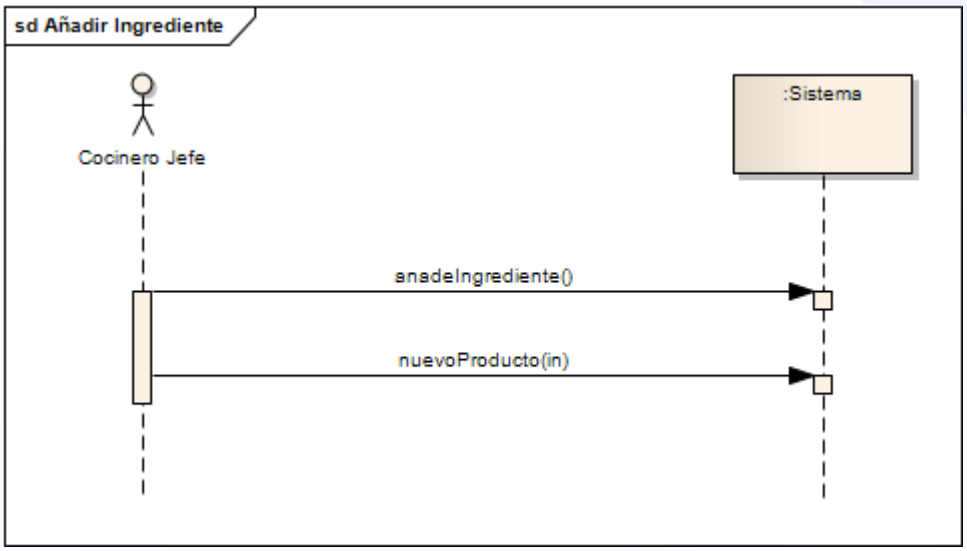
Imprimir lista de productos a pedir



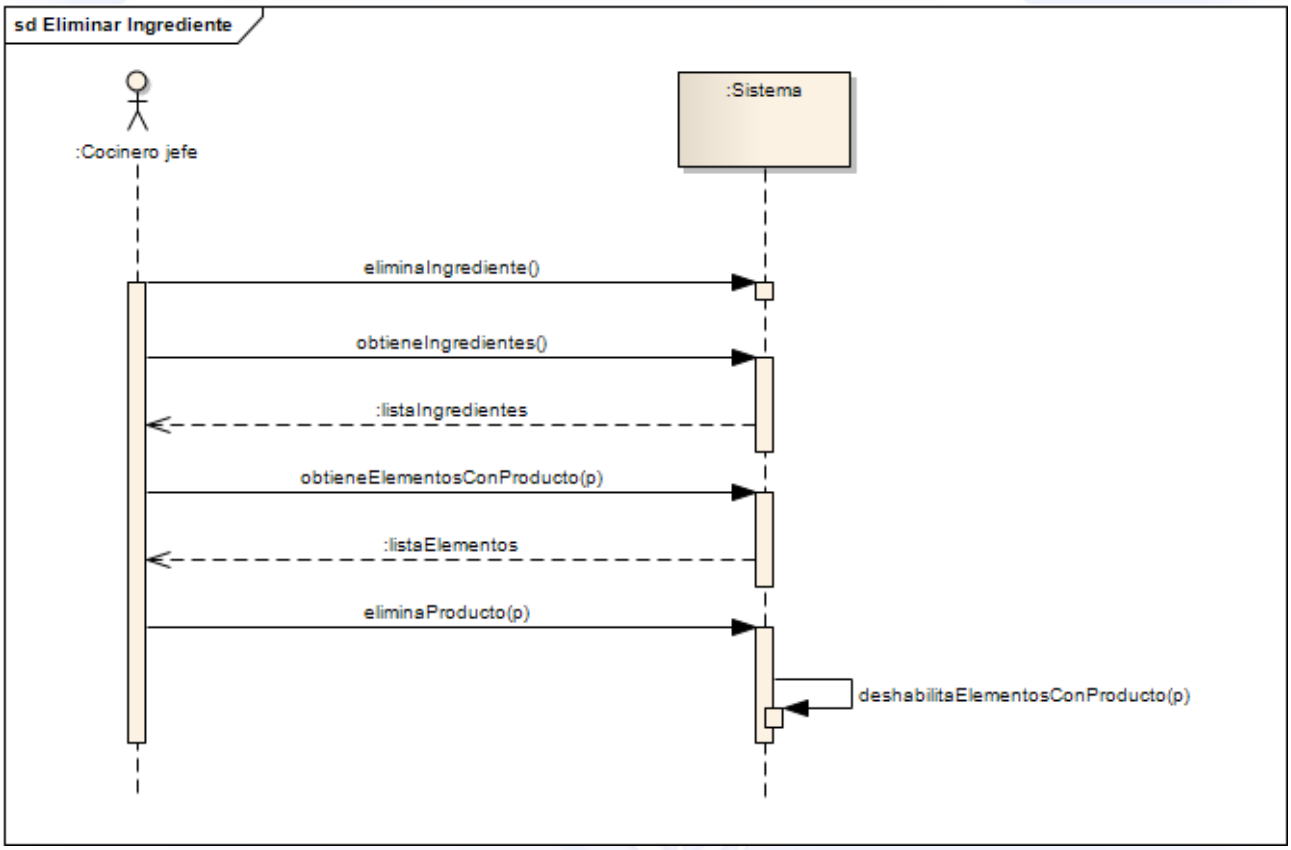
Notificar recepción de pedido del proveedor



Añadir ingrediente

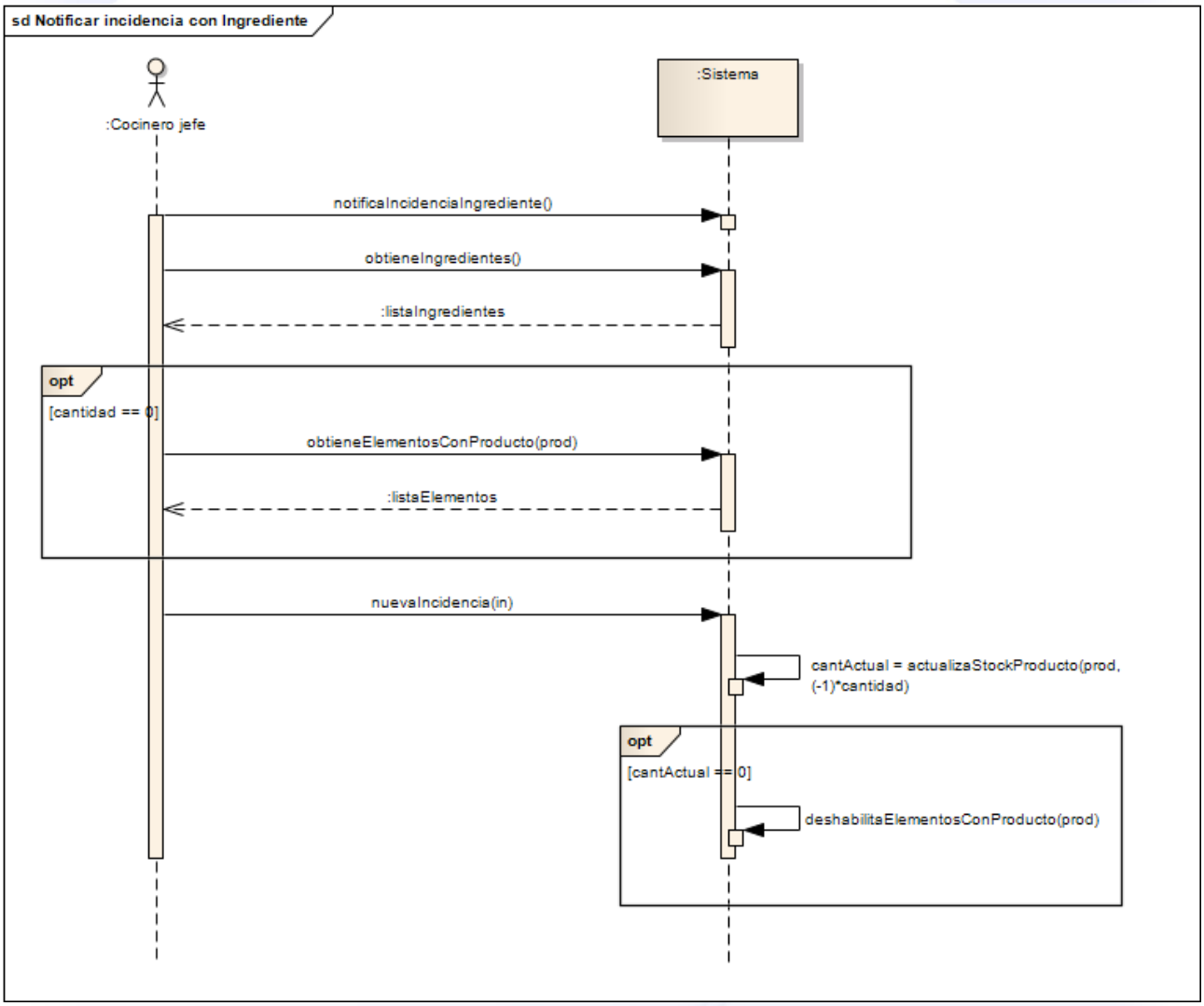


Eliminar ingrediente

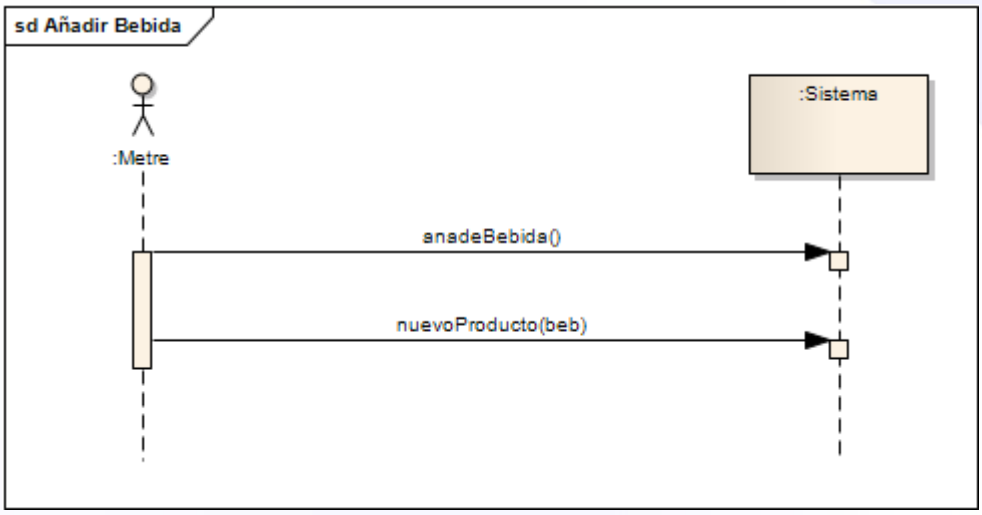


Modificar ingrediente

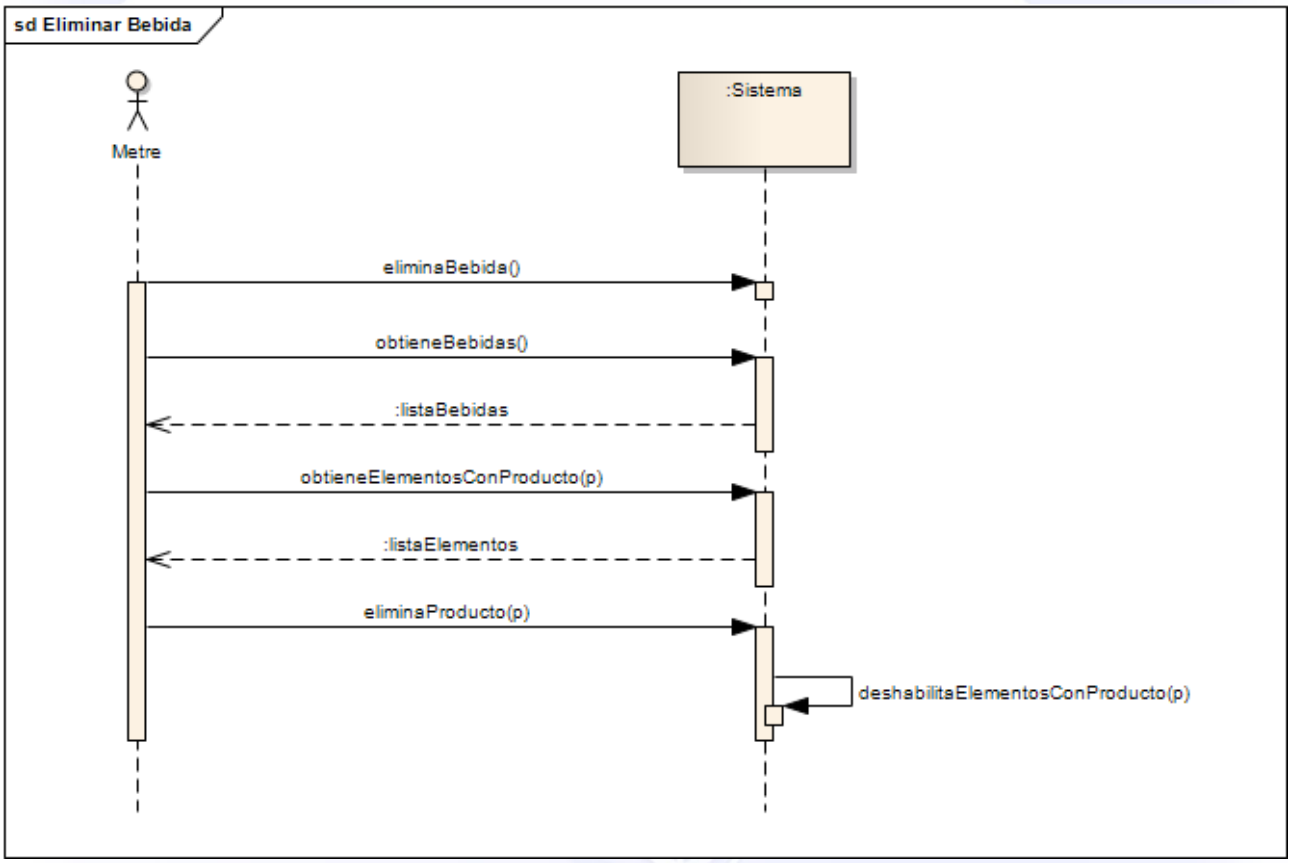
Notificar incidencia con ingrediente



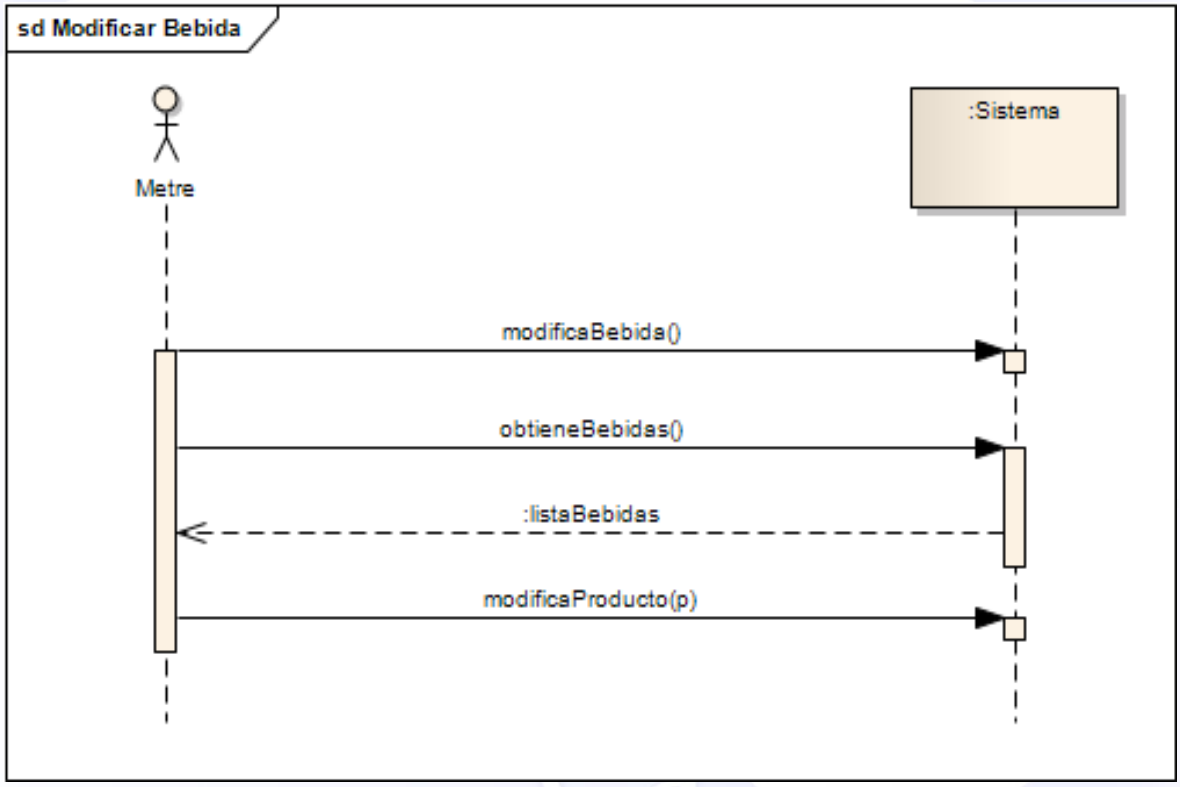
Añadir bebida



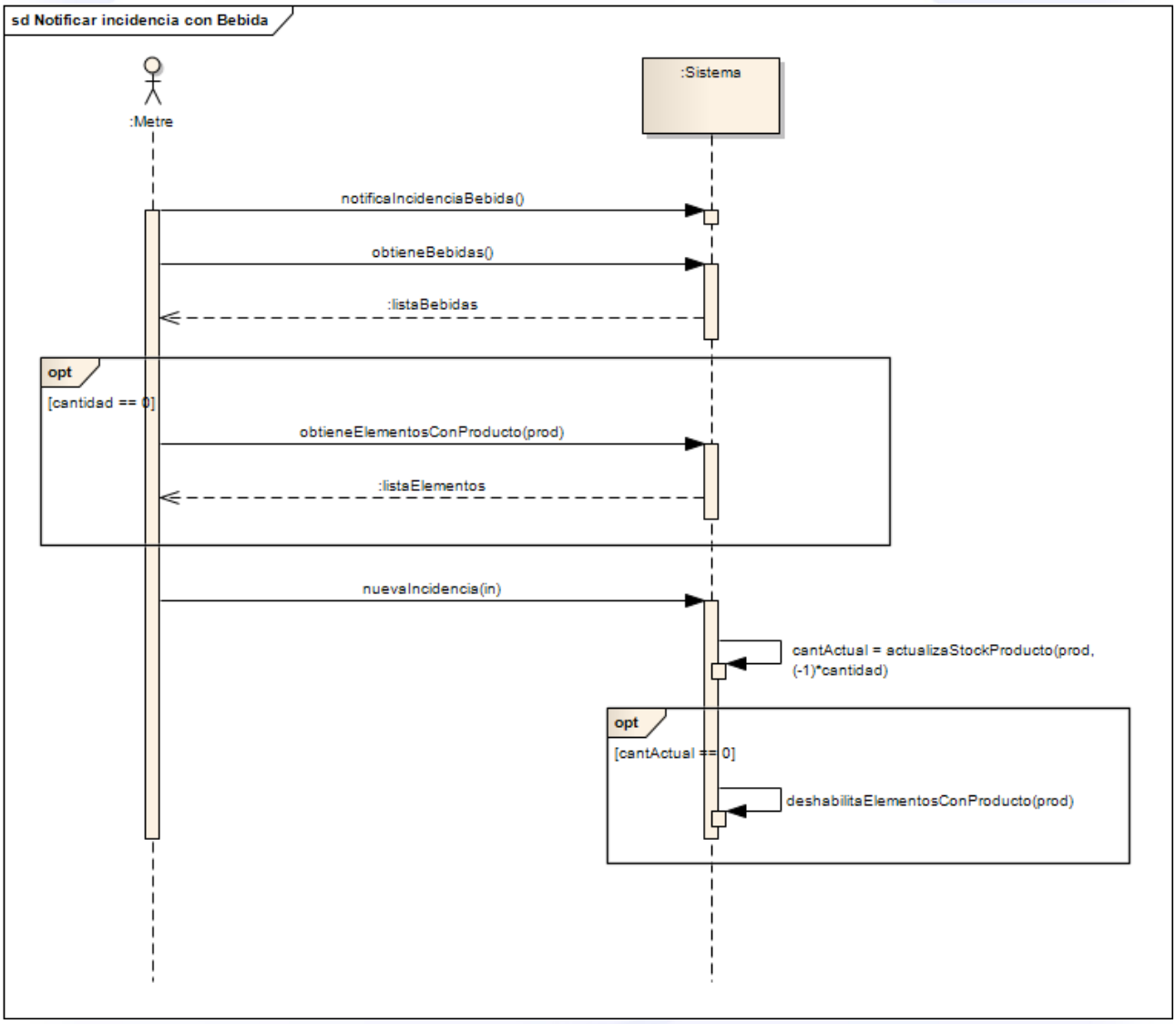
Eliminar bebida



Modificar bebida



Notificar incidencia con bebida



Cambia estado plato

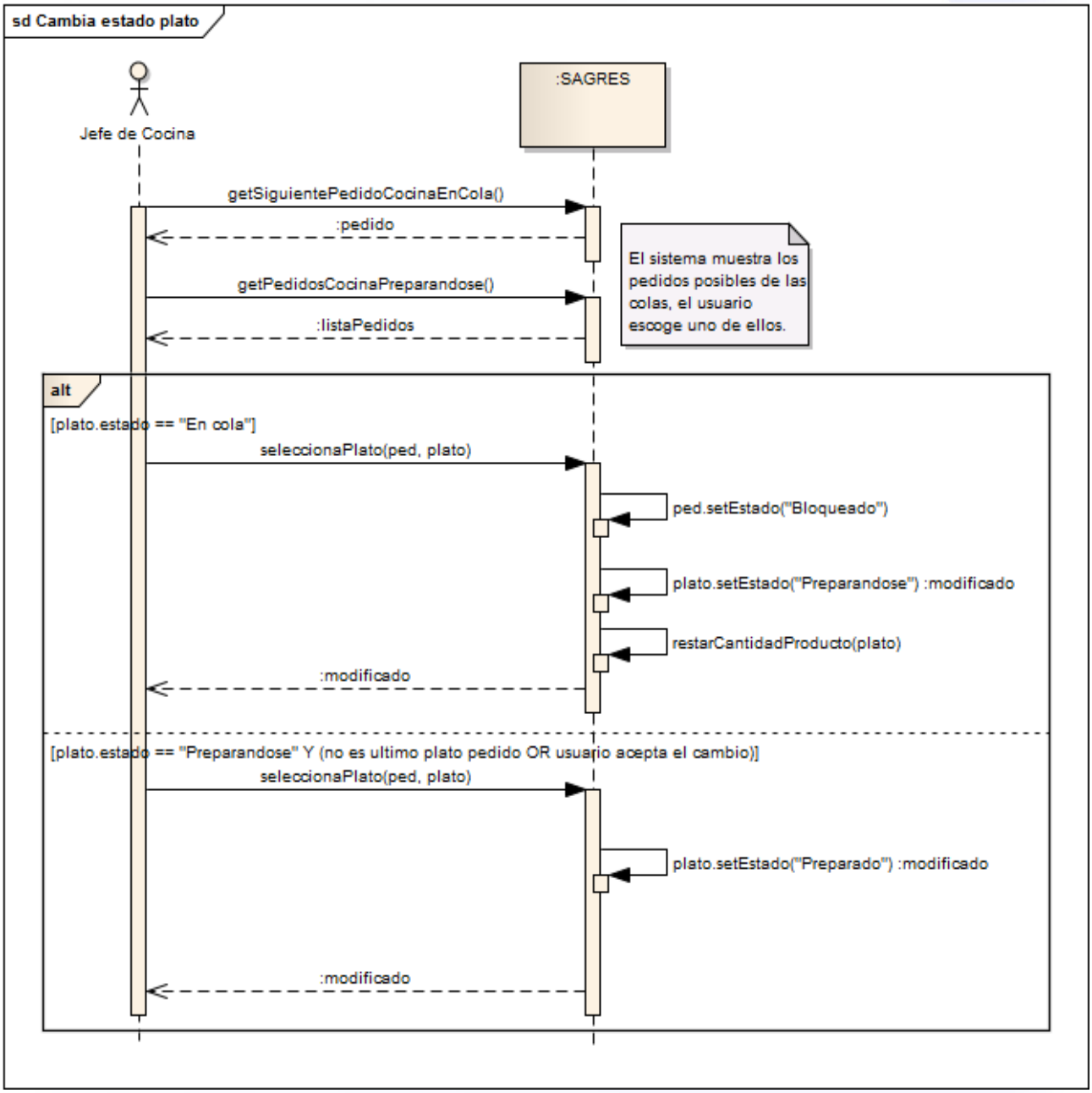


Diagrama realizado por Sergio Rodríguez Lumley

Cambia estado bebida

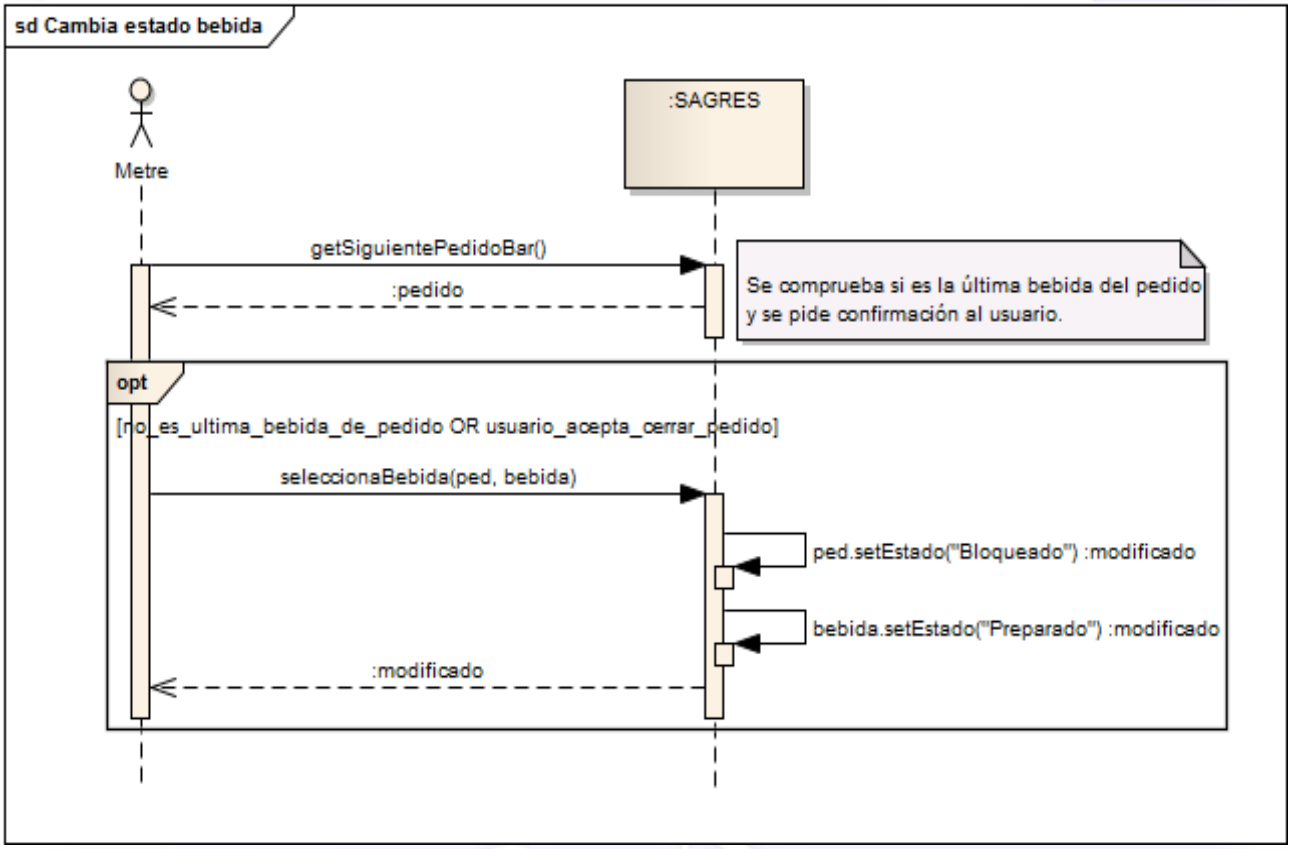


Diagrama realizado por Sergio Rodríguez Lumley

Nuevo Pedido

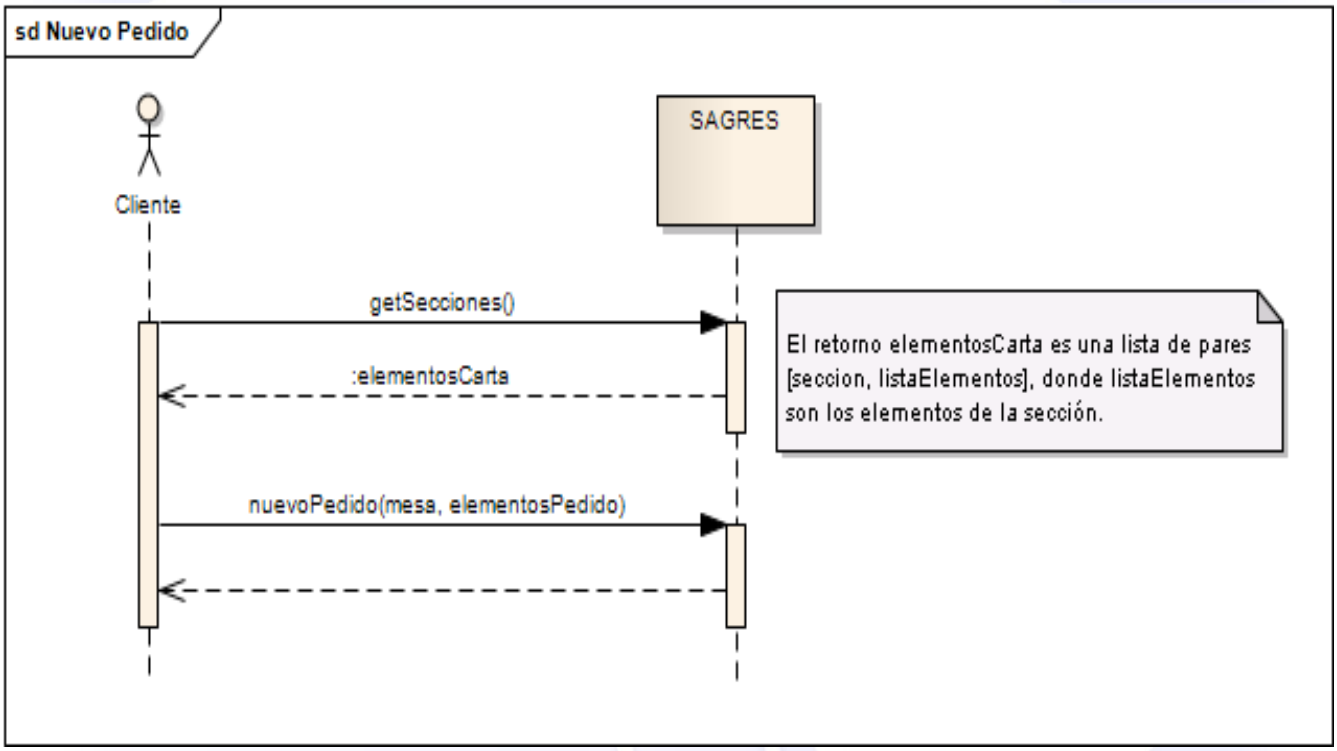


Diagrama realizado por Adrián Víctor Pérez Lopera

Modifica Pedido

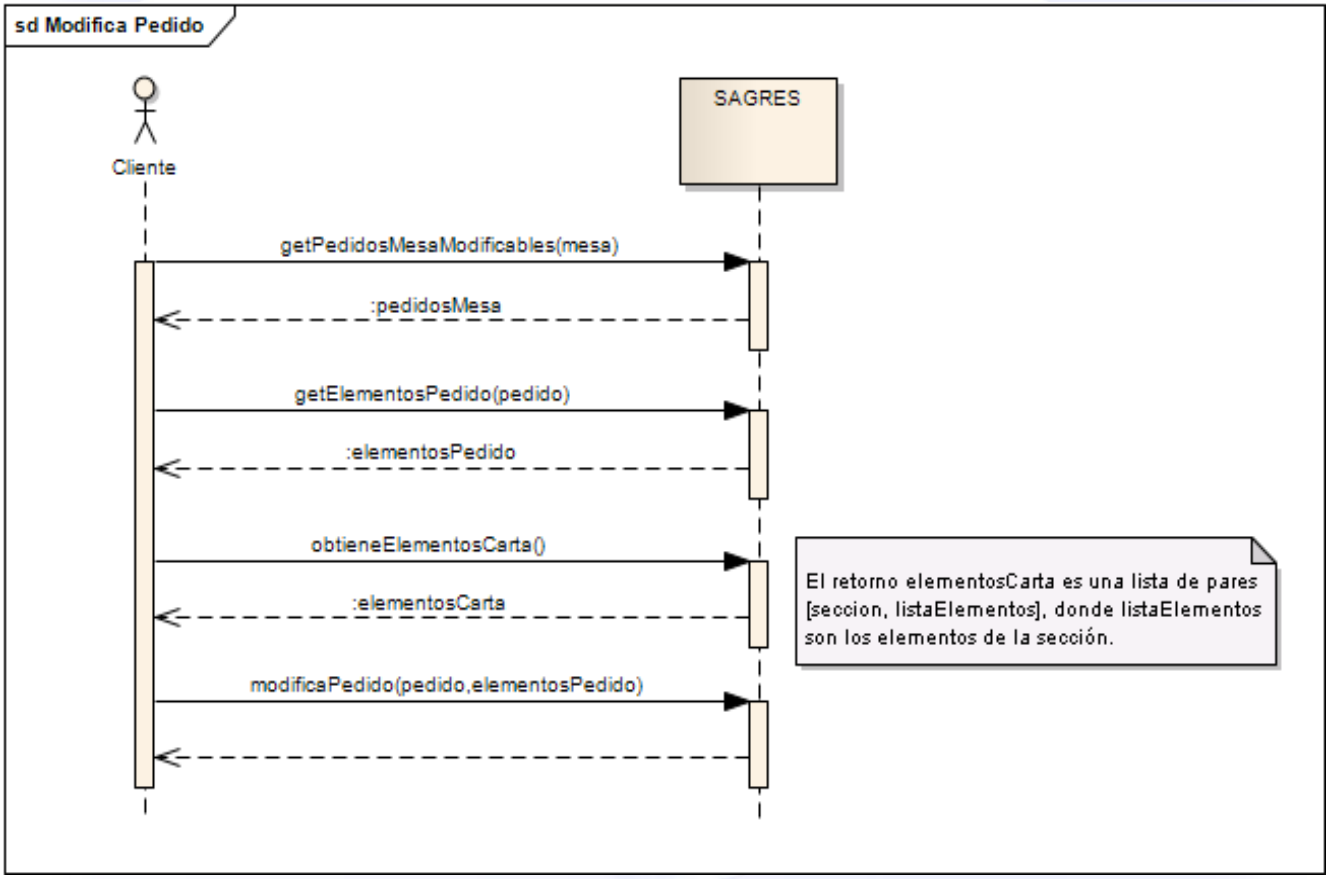


Diagrama realizado por Adrián Víctor Pérez Lopera

Subsistema de gestión de facturas

Pide Factura

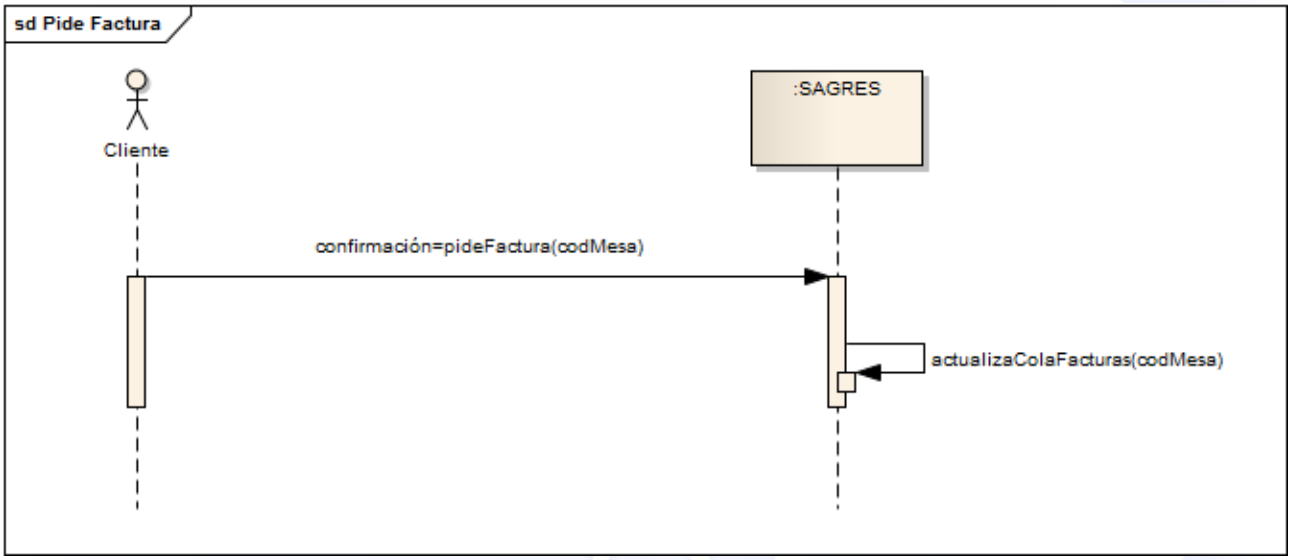
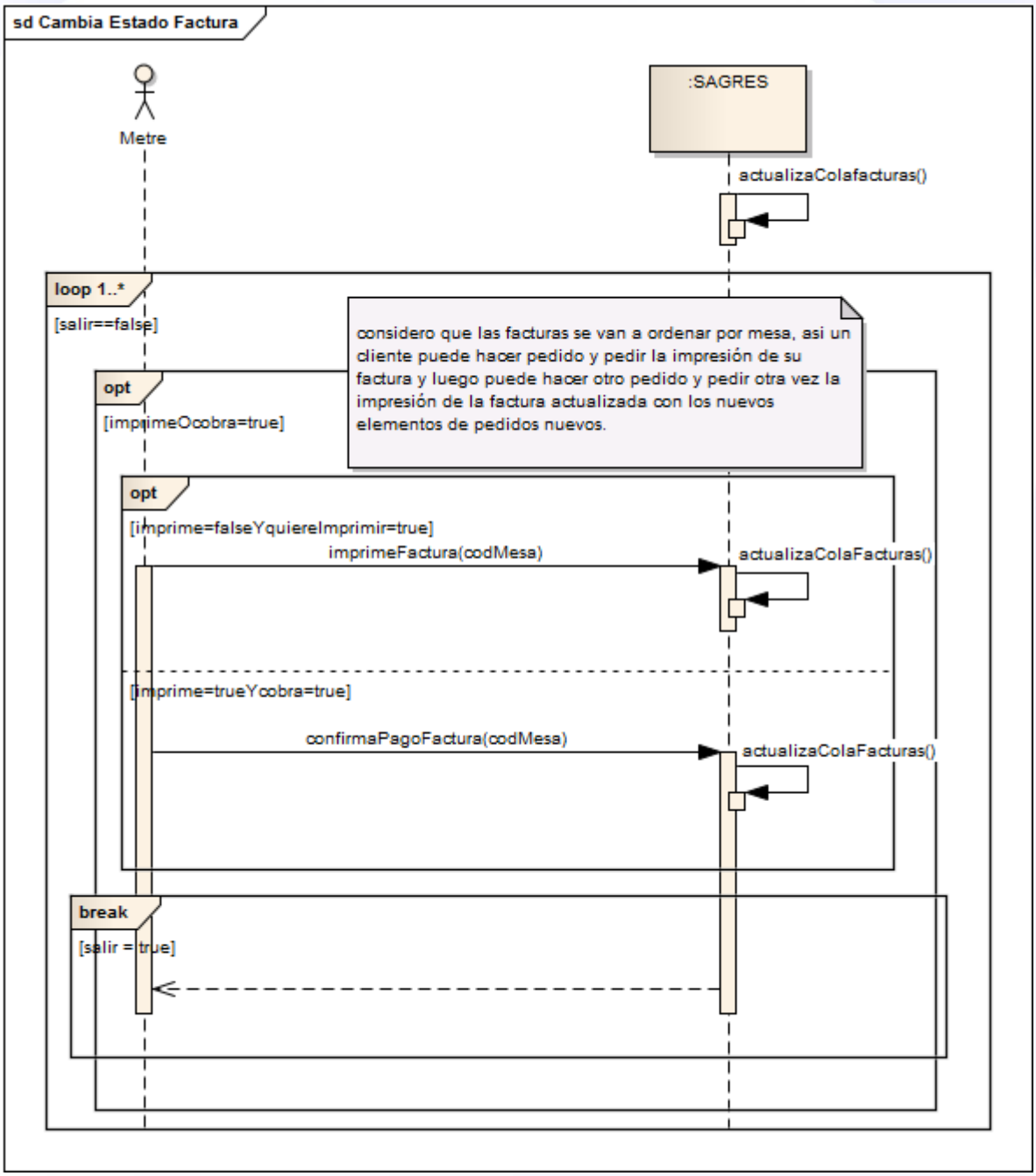


Diagrama realizado por Nabil Sabeg

Cambia Estado Factura



Nuevo Pedido

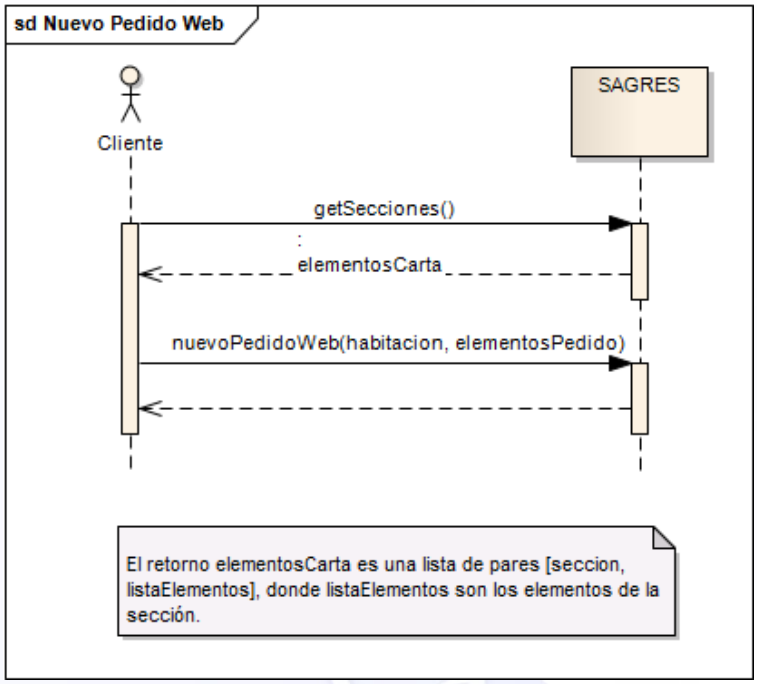


Diagrama realizado por Carlos Salas Morales

Modifica Pedido

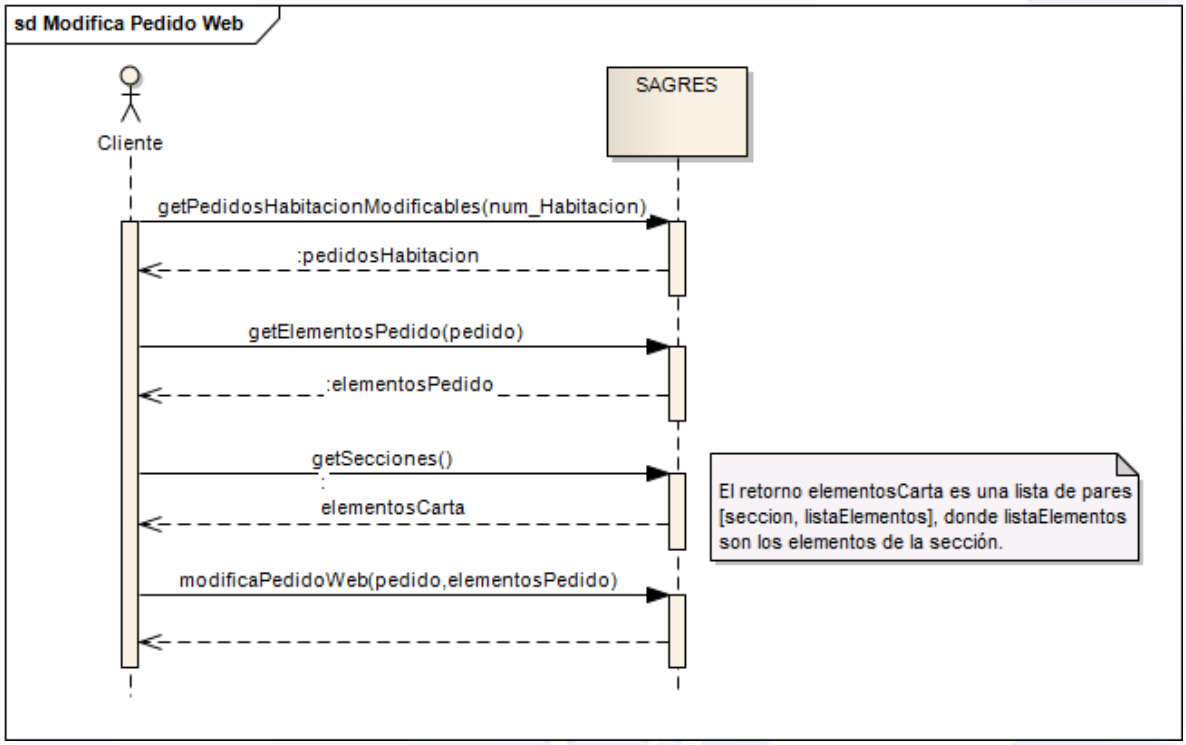


Diagrama realizado por José David Dionisio Ruiz

Consulta Estadística

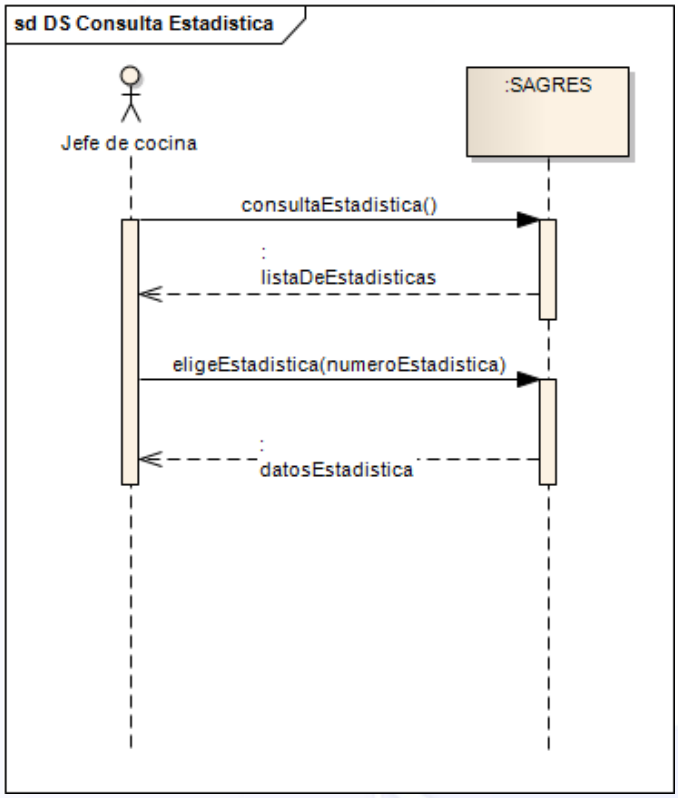


Diagrama realizado por Carlos Salas Morales

Identificación de operaciones del sistema

En este apartado vamos a enumerar las operaciones identificadas en el sistema ordenadas subsistema en el que se utilizan. Esto significa que puede haber operaciones que se repiten en distintos subsistemas.

Añadir elemento a la carta

Operación	Descripción
anadeElementoACarta()	operación que inicia el caso de uso para añadir un nuevo elemento a la carta.
obtieneSecciones()	Devuelve una lista con las secciones registradas en el sistema.
obtieneProductosSeccion(sec)	Devuelve una lista con las secciones registradas en el sistema.
nuevoElemento(ele,sec)	Registra un nuevo elemento en el sistema y lo asocia con su sección

Eliminar elemento de la carta

Operación	Descripción
eliminaElementoDeCarta()	operación que inicia el caso de uso para eliminar un elemento de la carta.
obtieneSecciones()	Devuelve una lista con las secciones registradas en el sistema.
obtieneElementosSeccion(sec)	Devuelve una lista de elementos asociados a esa sección
eliminaElemento(ele)	Elimina el elemento del sistema

Modificar elemento de la carta

Operación	Descripción
modificaElementoDeCarta()	operación que inicia el caso de uso para modificar un elemento de la carta. Devuelve la lista de elementos registrados en el sistema.
obtieneSecciones()	Devuelve una lista con las secciones registradas en el sistema.
obtieneElementosSeccion(sec)	Devuelve una lista de elementos asociados a esa sección
modificaElemento(ele)	Registra en el sistema los cambios realizados en el elemento.

Imprimir lista de productos a pedir

Operación	Descripción
obtieneProductosBajoMinimos()	Devuelve una lista de productos cuya cantidad en stock esta por debajo de su mínimo y las cantidades necesarias para que alcancen su máximo.
imprimeListaProductosaPedir()	Ordena a la impresora que imprima el pedido y guarda una copia en el sistema.

Notificar recepción de pedido del proveedor

Operación	Descripción
notificaRecepcionPedido()	operación que inicia el caso de uso para notificar al sistema que se ha recibido el pedido del proveedor.
obtienePedidoProveedor()	Devuelve un objeto con la informacion del último pedido pendiente de recibir.
compruebaElementosInvalidados(listaProductosCantidades)	Devuelve una lista con los elementos de carta que se habilitarían tras actualizar las cantidades de los productos recibidos.
notificaRecepcionPedido(pedProveedor)	Actualiza las cantidades de los productos en el sistema y habilita los elementos de carta correspondientes.

Añadir ingrediente

Operación	Descripción
anadeIngrediente()	Operación que inicia el caso de uso para añadir un nuevo ingrediente.
nuevoProducto(in)	Registra un nuevo ingrediente en el sistema.

Eliminar ingrediente

Operación	Descripción
eliminaIngrediente()	Operación que inicia el caso de uso para eliminar un ingrediente.
obtieneIngredientes()	Devuelve una lista con los ingredientes registrados en el sistema
obtieneElementosConProducto(p)	Devuelve una lista con los elementos de carta que tiene el producto “p” asociado.
eliminaProducto(p)	Elimina el producto p del sistema e invalida los elementos de

	carta que tienen asociado dicho producto.

Modificar ingrediente

Operación	Descripción
modificaIngrediente()	Operación que inicia el caso de uso para modificar un ingrediente.
obtieneIngredientes()	Devuelve una lista con los ingredientes registrados en el sistema
modificaProducto(p)	Registra en el sistema los cambios efectuados en el producto

Notificar incidencia con ingrediente

Operación	Descripción
notificaIncidenciaIngrediente()	operación que inicial el caso de uso para notificar una incidencia con un ingrediente.
obtieneElementosConProducto(prod)	Devuelve una lista con los elementos que tienen ese producto asociado.
nuevaIncidencia(in)	Registra una nueva incidencia en el sistema y actualiza la cantidad del producto involucrado. Si la cantidad es igual a 0 deshabilita los elementos de la carta que tengan asociado ese producto.

Añadir Bebida

Operación	Descripción
anadeBebida()	Operación que inicia el caso de uso para añadir una nueva bebida.
nuevoProducto(beb)	Registra una nueva bebida en el sistema

Añadir Bebida

Operación	Descripción
eliminaBebida()	Operación que inicia el caso de uso para eliminar una bebida.
obtieneBebidas()	Devuelve una lista con las bebidas registrados en el sistema
obtieneElementosConProducto(p)	Devuelve una lista con los elementos de carta que tiene el producto “p” asociado.
eliminaProducto(p)	Elimina el producto p del sistema e invalida los elementos de

	carta que tienen asociado dicho producto.

Modificar bebida

Operación	Descripción
modificaBebida()	Operación que inicia el caso de uso para modificar una bebida.
obtieneBebidas()	Devuelve una lista con las bebidas registrados en el sistema
modificaProducto(p)	Registra en el sistema los cambios efectuados en el producto

Notificar incidencia con bebida

Operación	Descripción
notificaIncidenciaBebida()	operación que inicial el caso de uso para notificar una incidencia con una bebida.
obtieneElementosConProducto(prod)	Devuelve una lista con los elementos que tienen ese producto asociado.
nuevaIncidencia(in)	Registra una nueva incidencia en el sistema y actualiza la cantidad del producto involucrado. Si la cantidad es igual a 0 deshabilita los elementos de la carta que tengan asociado ese producto.

Alta de pedidos

Operación	Descripción
elementosCarta = getSecciones()	Retorna las secciones de la carta con sus elementos.
nuevoPedido (mesa, elementosPedido)	Incluye un nuevo pedido con estado “modificable” en el sistema.

Operaciones definidas por Adrián Víctor Pérez Lopera

Modificar pedidos

Operación	Descripción
pedidosMesa = getPedidosMesaModificables(mesa)	Retorna los pedidos modificables de una mesa.
ElementosPedido = getElementosPedido(pedido)	Obtiene los elementos asociados a un pedido con la cantidad de cada uno.

elementosCarta = getSecciones ()	Retorna las secciones de la carta con sus elementos. Es la misma operación que la definida en “Nuevo Pedido”.
modificaPedido (pedido, elementosPedido)	Cambia los elementos de un pedido por los contenidos en elementosPedido. Al modificar un pedido, este pasa a estar al final de la comanda.

Operaciones definidas por Adrián Víctor Pérez Lopera

Cola de cocina

Operación	Descripción
Pedido = getSiguientePedidoCocinaEnCola() ;	Devuelve el pedido con la fecha más lejana (es decir, el más antiguo) que tenga algún ElementoColaCocina en estado “En cola”.
Pedido = getPedidosCocinaPreparandose();	Devuelve los pedidos con algún plato en estado “Preparandose”.
Bool = seleccionaPlato(Pedido pedido, ElementoColaCocina plato);	Selecciona un “plato” del Pedido “pedido” y cambia el estado del plato. Si estaba “En cola” lo cambia a “Preparandose” y el del pedido a “Bloqueado”. Si estaba “Preparandose” se cambiará a “Preparado” devolviendo “True”, o “False” si no ha podido cambiarlo.

Operaciones definidas por Sergio Rodríguez Lumley

Cola de bar

Operación	Descripción
Pedido = getSiguientePedidoBar()	Devuelve el pedido con la fecha más lejana (es decir, el más antiguo) que tenga algún ElementoColaBar en estado “En cola”.
Bool = seleccionaBebida(Pedido pedido, ElementoColaBar bebida);	Selecciona una “bebida” del Pedido “pedido”, cambia el estado y también el de la bebida. Si estaba “En cola” se cambiará a “Preparado” devolviendo “True”, o “False” si no ha cambiado el estado.

Operaciones definidas por Sergio Rodríguez Lumley

Facturas

Operación	Descripción
Confirmacion=pideFactura(Cod Mesa)	Manda al sistema la petición de imprimir una factura en la mesa con el código “codMesa”
actualizaColaFacturas()	Actualiza la cola de facturas.
actualizaColaFacturas(codMesa, listaPedidos)	Actualiza la cola de facturas.
imprimeFactura(codMesa)	Imprime la factura de la mesa con el código (codMesa).
confirmaPagoFactura(codMesa)	Confirma el pago de una factura y verifica si hay elementos en la mesa con código (codMesa) no han sido facturados, en caso de que ocurra el caso genera la factura de todos los elementos.

Operaciones definidas por sabeg nabil

Alta de pedidos de hotel

Operación	Descripción
elementosCarta = getSecciones ()	Retorna las secciones de la carta con sus elementos.
nuevoPedidoWeb (habitación, elementosPedido)	Incluye un nuevo pedido con estado “modificable” en el sistema.

Operaciones definidas por Carlos Salas Morales

Modificar pedidos

Operación	Descripción
pedidosHabitacion = getPedidoHabitacionModificables(habitacion)	Retorna los pedidos modificables de una habitación.
ElementosPedido = getElementosPedido(pedido)	Obtiene los elementos asociados a un pedido con la cantidad de cada uno.
elementosCarta = getSecciones ()	Retorna las secciones de la carta con sus elementos. Es la misma operación que la definida en “Nuevo Pedido”.
modificaPedidoWeb (pedido, elementosPedido)	Cambia los elementos de un pedido por los contenidos en elementosPedido. Al modificar un pedido, este pasa a estar al final de la comanda.

Operaciones definidas por José David Dionisio Ruiz

Consulta Estadística

Operación	Descripción
consultaEstadistica()	Devuelve la lista de estadísticas disponibles.
eligeEstadistica(estadistica)	Elige un tipo de estadística entre las disponibles y muestra los datos relacionados. Existen estos tipos: • Plato mas pedido • Plato menos pedido • Elementos mas afectados por la falta de un ingrediente. • Balance de ganancias

Operaciones definidas por Carlos Salas Morales

Apéndice 1.0

Fecha	15/03/10
Descripción del problema	-
Impacto del problema	-
Soluciones adoptadas	• Se generado el documento de modelado de requisitos inicial
Anexos a la versión	

Apéndice 1.1

Fecha	16/03/10
Descripción del problema	Se han encontrado unos pequeños problemas en las especificaciones de algunos casos de uso que nos han obligado a modificar tanto estos como sus respectivos diagramas de secuencia. Además, el hecho de que los diagramas de secuencia se hayan repartido ha provocado que aparezcan distintas denominaciones para operaciones del sistema que realmente tienen la misma funcionalidad. También nos hemos percatado de que no habíamos contemplado el requisito funcional RF14, el cual especifica que si un producto se agota se deben deshabilitar los elementos de la carta que dependan de él.
Impacto del problema	Los problemas mencionados en la descripción del problema son concretamente que en algunos casos de uso realizábamos demasiadas llamadas a operaciones de consulta. Esto podría resultar perjudicial para la eficiencia de nuestra aplicación. El hecho de que haya operaciones con distinto nombre pero igual funcionalidad iba a suponer un coste tremendo para nosotros en primer lugar porque hubiéramos hecho los contratos de operaciones repetidas, y en segundo lugar para el equipo de implementación puesto que habrían invertido un tiempo valioso en repetir operaciones del sistema absurdamente. No tener en cuenta el requisito funcional RF14 hubiera provocado que el cliente hubiese podido seleccionar un plato para el cual no hubiera habido ingredientes para prepararlo.
Soluciones adoptadas	• Se han corregido los resúmenes de los casos de uso para que fuesen más claros. • Se han eliminado consultas innecesarias en los cursos normales de algunos casos de uso. • Se han unificado los nombres de las operaciones del sistema con funcionalidad equivalente. • Se han modificado los diagramas de secuencia para que se contemplen las soluciones descritas en los puntos anteriores. • Se ha incluido el requisito funcional RF14 tanto es la especificaciones de los casos de uso involucrados como en los diagramas de secuencia de esos casos de uso.
Anexos a la versión	

Apéndice 1.2

Fecha	21/03/10
Descripción del problema	Se han encontrado unos pequeños problemas en los diagramas de secuencia del sistema de algunos casos de uso. Estos problemas tenían que ver con algunas operaciones de consultas y con los nombres de algunas operaciones del sistema que no explicaban bien su cometido. No queda del todo claro la función de los casos de uso "Consultar..."
Impacto del problema	Algunos diagramas de secuencia tenían la operación salir() cuando en realidad no hacía falta. Esto podría suponer un error de interpretación para el equipo de implementación. Algunos nombres de las operaciones no se correspondían con su cometido, a entendimientos ambiguos por parte de los integrantes de nuestro equipo. El carácter cíclico que hemos querido darles a los casos de uso nos ha supuesto muchos problemas en el diseño de los diagramas de secuencia.
Soluciones adoptadas	• Se han eliminado tanto de los diagramas de secuencia como de las especificaciones de los casos de uso el último paso correspondiente siempre a "el usuario finaliza la operación". • Se han modificado algunos nombres de operaciones y la correspondiente lista final de éstas. • Se ha optado por borrar los casos de uso que hacían referencia a todo tipo de "consultas", puesto que consideramos que es mejor tratarlas como operaciones privadas del sistema. • También se ha corregido el formato de los cursos alternativos de algunas especificaciones de casos de uso ya que no quedaban del todo claros. • Se ha eliminado de los casos de uso la posibilidad de realizarlos varias veces. Ahora cada caso de uso solo se ejecutará una vez, quedando su especificación mucho más clara.
Anexos a la versión	

Apéndice 1.3

Fecha	15/04/10
Descripción del problema	Tras la entrega de toda documentación de la 1ª iteración y la defensa de ésta, el equipo de planificación de la 2ª iteración ha propuesto una revisón completa de la 1ª iteración para solventar los fallos encontrados por el profesor.
Impacto del problema	Cuando se diseñaron los diagramas de secuencia del sistema, se llegó al acuerdo de que las operaciones del sistema recibirían los parámetros necesarios para la creación de los objetos. Esto presenta un problema a la hora de añadir nuevo parámetros a los objetos, lo cual obligaría a cambiar las cabeceras de las funciones. Otro problema detectado ha sido que el usuario no podía tener conocimiento previo de qué consecuencias tendría la realización de una operación concreta antes de realizarla.
Soluciones adoptadas	• Se han modificado las cabeceras de las funciones para trabajar directamente con objetos. • Se han añadido operaciones de comprobación para que el usuario pueda ver el estado futuro del sistema tras la realización de una operación.
Anexos a la versión	

Fecha	15/04/10
Descripción del problema	Se ha generado el documento de Modelado de Requisitos.
Impacto del problema	
Soluciones adoptadas	•
Anexos a la versión	

Apéndice 2.1

Fecha	22/04/10
Descripción del problema	1. Se especificó en la introducción que las facturas se guardan en la base de datos únicamente cuando se cierra la factura. 2. La especificación de los casos de uso "Cambia estado plato" y "Cambia estado bebida" estaban poco detallados. En consecuencia, los diagramas de secuencia y sus operaciones correspondientes contenían errores. 3. Se han especificado una serie de puntos en el documento de revisión del "Documento de Modelado de Requisitos v1.0".
Impacto del problema	1. Puede provocar inconsistencia en la persistencia de datos. 2. Falta de entendimiento del sistema de colas de bar y de cocina, lo que puede dar lugar a un mal análisis y diseño del mismo. 3. Posibles errores de interpretación.
Soluciones adoptadas	1. Se ha especificado que las facturas se guardan en todo momento. 2. Se han vuelto a especificar los casos de uso "Cambia estado plato" y "Cambia estado bebida", se han vuelto a generar sus diagramas de secuencia y sus operaciones correspondientes. 3. Se han corregido varios de estos puntos.
Anexos a la versión	

Apéndice 2.2

Fecha	24/04/10
Descripción del problema	1. Los diagramas de secuencia del sistema “Nuevo pedido” y “Modifica pedido” contenían operaciones al sistema que no producían cambios en este. 2. En las especificaciones de los casos de uso “Nuevo pedido” y “Modifica pedido” se realizaba el retorno de los elementos de la carta cada vez que se insertaba o eliminaba un elemento del pedido en curso. 3. En los diagramas de secuencia del “Cola de bar” y “Cola de cocina” se mostraron partes que se esperaba encontrar en la vista.
Impacto del problema	1. Resultaba confusa la interpretación de estos diagramas en la fase de análisis. En concreto, su definición acarreaba un problema a la hora de construir los contratos de sus operaciones de sistema. 2. Estas acciones eran innecesarias en este nivel de abstracción. 3. Se daba la sensación de que toda la funcionalidad mostrada debía de aparecer como operaciones del sistema, en el controlador, lo que habría dado más trabajo sin razón alguna.
Soluciones adoptadas	1. Se han redefinido los diagramas por completo, de manera se llevan a cabo las funcionalidades de una manera más intuitiva de cara a las fases de análisis y diseño posteriores. En consecuencia, se han redefinido también las operaciones del sistema asociadas a ambos casos de uso. 2. Se han eliminado estas acciones del curso normal en las especificaciones de ambos casos de uso. 3. Han sido modificados para mostrar únicamente la parte del controlador, reduciendo así también el número de operaciones.
Anexos a la versión	

Apéndice 2.3

Fecha	06/05/10
Descripción del problema	1. Según el requisito funcional RF11 se podía anular facturas. 2. El en diagrama de secuencia "pideFacturas" se llevaba como parámetro listaPedidos. 3. Error en los aspectos legales del sistema. 4. La especificación del caso de uso Cambia estado plato indicaba que solo se mostraba un pedido en estado "Preparandose". 5. La operación imprimeFacturas() se ha quitado porque no presentaba mucha utilidad en el sistema. 6. El segundo parámetro de La operación pideFactura(codMesa,listaElementos) no era correcto. 7. Algunas de las operaciones de sistema de los diagramas de secuencia del sistema "Nuevo Pedido" y "Modifica Pedido" resultaban redundantes.
Impacto del problema	1. El sistema resulta más complejo a la hora de decidir que es lo que se anula en una factura. 2. No se pueden mandar listas de pedidos. 3. Se usan herramientas que no son gratuitas en el desarrollo. 4. El metre solo podía ver el pedido menos reciente en la cola de preparándose. 5. Se complica mucho el sistema al ahora de realizar la operación. 6. El segundo parámetro listaElementos no tenia sentido ponerlo ya que los pedidos de una factura se podían saber desde el sistema. 7. En etapas posteriores del modelado surgirían problemas a la hora de desarrollar en más profundidad dichas operaciones.
Soluciones adoptadas	• Se ha quitado RF11 • se han cambiado los diagramas de secuencia relacionados con el requisito y la descripción de algunas operaciones. • Se han cambiado los aspectos legales del sistema. • Se ha modificado la especificación, el diagrama y las operaciones para que muestre todos los pedidos con algún plato preparándose, y se ha especificado que se reduce la cantidad en stock. • Se ha eliminado la operación y se ha actualizado el diagrama de secuencia "cambia estado factura", la descripción del mismo y se ha quitado la operación de la lista de operaciones del sistema. • Se ha quitado el parámetro de la operación. • Se han modificado los diagramas de secuencia del sistema "Nuevo Pedido" y "Modifica Pedido".
Anexos a la versión	

Apéndice 3.0

Fecha	18/05/10
Descripción del problema	Se ha generado el documento de Modelado de Requisitos.
Impacto del problema	
Soluciones adoptadas	•
Anexos a la versión	

Apéndice 3.1

Fecha	18/05/10
Descripción del problema	Se ha generado la versión V1.1 del documento de Modelado de Requisitos. Problemas resueltos: • Descripción confusa de los subsistemas. • Descripción de actor cliente. • Requerimiento funcional de valoración de elementos de pedido. • Nombres en casos de uso. • Descripción de casos de uso incompleta. • Diagrama de secuencia con operaciones incoherentes. (obtieneElementosCarta() en modificaPedidoWeb) • Otros errores.
Impacto del problema	
Soluciones adoptadas	• Actualización de las descripciones de los subsistemas. • Añadida información adicional sobre el actor Cliente. • Se añade el RF6. • Se especifica un nuevo nombre para los casos de uso. • Se añade información adicional sobre los casos de uso. • Se corrigen las operaciones erróneas y se añade información adicional. • Solucionados problemas sintácticos.
Anexos a la versión	

Apéndice 3.2

Fecha	18/05/10
Descripción del problema	• Falta la definición de la funcionalidad de las estadisticas ofrecidas por el subsistema gestor de estadisticas. • Disgregación de los documentos.
Impacto del problema	
Soluciones adoptadas	• Definición de las funcionalidades de las estadística. • Unificación de los 3 documentos de Modelado de Requisitos. ◦ Añadido diagrama de casos de uso global ◦ Añadido diagrama de paquetes global ◦ Incluidas todos los diagramas de secuencia. ◦ Incluidos todos las operaciones del sistema.
Anexos a la versión	