



**Universidad de Cienfuegos**  
**Facultad de Ingeniería**  
**Carrera de Ingeniería Informática**

**Título:**

Sistema informático para la gestión de la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos

Trabajo de diploma para optar por el título de Ingeniería en Informática

**Autor:**

Yadira Falcón Rosell

**Tutor:**

MSc. Oscar Luis Muñoz González

**Cienfuegos, Cuba**

**Curso 2014 – 2015**

Declaro que soy la única autora de este trabajo y autorizo a la Facultad de Ingeniería de la Universidad de Cienfuegos y al Centro de Estudios de Ingeniería y Sistemas para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del \_\_\_\_\_.

---

Firma Autor

Yadira Falcón Rosell

---

Firma Tutor

MSc. Oscar Luis Muñoz  
González

*A mis padres*

El presente trabajo se desarrolló en la Residencia de Postgrado de la Universidad de Cienfuegos y tuvo como fin la implementación de un sistema informático para gestionar la información referente al servicio de alimentación en la entidad, que permita desarrollar las actividades que actualmente influyen en la prestación del servicio, de forma que se agilicen los procesos; así como obtener el Inventario a Precio de Venta (IPV) que muestra el control de productos normados, buscando hacer esta información persistente en el tiempo y fácil de manipular por los involucrados en el proceso, lográndose un manejo seguro de los datos.

Se empleó para el desarrollo de la aplicación Java como lenguaje de programación, Hibernate como herramienta ORM (Object Relational Mapping) para la persistencia de los datos y el framework integrador Seam; soportados sobre la Plataforma Java Enterprise Edition 5.0, además de PostgreSQL como Sistema Gestor de Bases de Datos.

Palabras claves: Sistema de gestión, Servicio de alimentación, Tecnología Java

This work was developed in Graduate Residence of the University of Cienfuegos and was aimed at implementing a system to manage information relating to food service in the state, allowing develop activities currently affect the provision service, so that the process be expedited; and obtain the Inventory Sales Price showing control regulated products, seeking to make this persistent information in time and easily manipulated by those involved in the process, achieving a secure data management.

It was used for the development of the Java programming language implementation, Hibernate as ORM (Object Relational Mapping) tool for data persistence and integrative framework Seam; supported on Java Platform Enterprise Edition 5.0, as well as PostgreSQL Database Manager System.

Key words: Management system, Food services, Java Technologies

Introducción.....	1
Capítulo I: Fundamentación teórica.....	5
1.1 Uso de las Tecnologías de la Información y las Comunicaciones (TIC) en los servicios gastronómicos. ....	5
1.2 Conceptos asociados al dominio del problema.....	7
1.3 Flujo del proceso y análisis crítico .....	9
1.4 Análisis de los sistemas informáticos existentes .....	10
1.4.1 Sistemas informáticos existentes a nivel internacional.....	11
1.4.2 Sistemas informáticos existentes nacionalmente .....	12
1.4.3 Análisis comparativo de las soluciones existentes con la propuesta	13
1.5 Metodologías de Desarrollo de Software .....	14
1.5.1 El Proceso Unificado de Desarrollo de Software (RUP).....	15
1.6 Arquitectura cliente-servidor .....	16
1.6.1 Arquitectura de N Capas .....	17
1.7 Patrón de arquitectura de software .....	18
1.7.1 Patrón de Diseño MVC.....	19
1.8 Sistema Gestor de Base de Datos.....	20
1.8.1 PostgreSQL.....	22
1.9 Lenguajes utilizados en el desarrollo del sistema informático .....	22

---

1.10 Servidor de aplicaciones.....	24
1.11 Framework, librerías y componentes .....	24
1.12 Herramientas utilizadas .....	28
Conclusiones del capítulo .....	30
Capítulo II: Descripción y construcción de la solución propuesta.....	31
Introducción al capítulo .....	31
2.1 Identificación de los procesos del negocio.....	31
2.2 Reglas del negocio a considerar.....	33
2.3 Modelo de casos de uso del negocio.....	34
2.3.1 Actores del negocio.....	34
2.3.2 Trabajadores del negocio.....	35
2.3.3 Diagrama de casos de uso del negocio. ....	36
2.3.4 Descripciones de los casos de uso del negocio.....	37
2.4 Modelo de objeto del negocio. ....	37
2.5 Descripción del modelo del sistema.....	38
2.5.1 Requerimientos funcionales. ....	39
2.5.2 Requerimientos no funcionales. ....	43
2.6 Modelo de caso de uso del sistema.....	45
2.6.1 Actores del sistema. ....	45
2.6.2 Casos de uso del sistema. ....	46

---

2.6.3 Diagrama de casos de uso del sistema.....	48
2.6.4 Descripción de los casos de uso del sistema .....	49
2.7 Construcción del sistema.....	50
2.7.1 Diagrama de clases del diseño. ....	50
2.7.2 Diseño de la base de datos.....	53
2.7.3 Modelo de despliegue. ....	54
2.8 Principios de diseño del sistema.....	55
2.8.1 Patrones de diseño .....	55
2.8.2 Estándares en la interfaz de la aplicación. ....	57
2.8.3 Tratamiento de excepciones. ....	57
2.8.4 Formato de los reportes. ....	58
2.8.5 Estándares de codificación. ....	58
Conclusiones del capítulo.....	59
Capítulo III: Estudio de factibilidad y diseño de los casos de prueba funcional	60
Introducción al capítulo.....	60
3.1 Estimación por puntos de casos de uso. ....	60
3.1.1 Planificación basada en casos de uso. ....	60
3.2 Análisis de los costos y beneficios.....	67
3.3 Diseño de pruebas funcionales.....	68
3.3.1 Caso de prueba para el Caso de Uso Autenticarse .....	68

---

3.3.2 Caso de prueba para el Caso de Uso Entrada de Productos.....	70
3.3.3 Caso de prueba para el Caso de Uso Gestionar menú.....	72
3.3.4 Caso de prueba para el Caso de Uso Salida de productos.....	74
3.3.5 Caso de prueba para el Caso de Uso Gestionar Producto .....	76
3.3.6 Caso de prueba para el Caso de Uso Asociar productos a un ingrediente .....	78
Conclusiones del capítulo .....	78
Conclusiones.....	80
Recomendaciones.....	81
Referencias bibliográficas .....	82
Bibliografía.. .....	87
Glosario de términos .....	94
Anexos.....	95
Anexo A: Descripción de los casos de uso del negocio.....	95
Anexo B: Diagramas de actividades del negocio.....	102
Anexo C: Descripción de los casos de uso del sistema.....	105
Anexo D: Diagramas de clases web .....	171
Anexo E: Diseño de la base de datos. ....	185

Tabla 1 Actores del negocio.....	34
Tabla 2 Trabajadores del negocio.....	35
Tabla 3 Casos de uso del negocio.....	37
Tabla 4 Actores del sistema.....	45
Tabla 5 Casos de uso del sistema.....	46
Tabla 6 Descripción de los casos de uso del sistema.....	49
Tabla 7 Diagrama de clases web.....	52
Tabla 8 Clasificación de los casos de uso.....	61
Tabla 9 Factor de complejidad técnica.....	64
Tabla 10 Factor de ambiente.....	65
Tabla 11 Esfuerzo de desarrollo del proyecto.....	67

Figura 1 Diagrama de casos de uso del negocio .....	36
Figura 2 Modelo de objeto del negocio .....	38
Figura 3 Diagrama de casos de uso del sistema .....	48
Figura 4 Diagrama de funcionamiento del framework Seam.....	51
Figura 5 Diagrama de despliegue .....	55
Figura 6 Caso de prueba Autenticarse.....	69
Figura 7 Caso de prueba Entrada de productos .....	70
Figura 8 Caso de prueba Gestionar Menú .....	72
Figura 9 Caso de prueba Gestionar Menú Siguiete .....	73
Figura 10 Caso de prueba Salida de productos .....	74
Figura 11 Caso de prueba Gestionar Producto Acción Insertar producto .....	76
Figura 12 Caso de prueba Gestionar Producto Acción Buscar producto .....	77
Figura 13 Caso de prueba Asociar productos a un ingrediente .....	78

## *Introducción*

Una universidad es un establecimiento o conjunto de unidades educativas de enseñanza superior e investigación, en la que se otorgan grados académicos y títulos profesionales. Hoy en día los términos maestro, doctor y profesor significan niveles diferentes de logro académico.[1]

Muchas son las universidades que brindan la oportunidad de obtener algunos de estos títulos profesionales. Cuba no se queda atrás, las universidades cubanas a la luz del desarrollo científico-técnico y las actuales circunstancias que vive el mundo, han tenido que recurrir necesariamente al intercambio, la interrelación y el vínculo entre profesionales, para encontrar un lenguaje común que ayude a fomentar y consolidar a la institución universitaria.

La Universidad de Cienfuegos (UCF) es considerada básicamente una universidad de postgrado, debido a que el número de programas en la educación postgraduada es superior al de pregrado. Llevar a cabo estos programas, exige de servicios de apoyo como son: alimentación, transporte, hospedaje, mantenimiento, entre otros. Para ello, la UCF cuenta con el Hotel Universitario “Residencia de Postgrado”, el cual incluye los servicios de alimentación y hospedaje; tanto para nacionales como extranjeros que se encuentran matriculados en programas de pregrado, postgrado y reuniones.

Debido a los diferentes tipos de huéspedes, el servicio de alimentación se complejiza, pues se deben planificar varias variantes de menú al día para cada uno de ellos; además de llevar el control de la entrada y salida de los productos en el almacén y determinar los costos de los menús. Una vez finalizado el día se debe elaborar el Inventario a Precio de Venta (IPV), que contiene información de cada producto como la cantidad con que inició el día, posibles entradas o salidas del mismo y la disponibilidad final. Además se elabora un Excel que indica las normas y cantidades a extraer para confeccionar cada menú según los diferentes tipos de clientes.

Este proceso en la actualidad se realiza en parte de forma manual y en parte controlado por medio de una hoja de cálculo Excel lo cual crea inconsistencias al comparar la cantidad real en el almacén con lo expuesto en los documentos.

Teniendo en cuenta estas afirmaciones, se define como **problema a resolver**:

La necesidad de gestionar la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos de forma rápida y confiable.

Identificándose como el **objeto de estudio** el proceso de servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos y como **campo de acción** la gestión de la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos.

La gestión, organización y control del flujo de información dentro de esta entidad resulta vital para la toma de decisiones y a la vez implica una alta responsabilidad teniendo en cuenta elementos como la fiabilidad, integridad, disponibilidad y seguridad. Es importante y necesario, que exista un buen flujo de información y comunicación, que facilite el trabajo; por lo que el uso de las tecnologías de la información y las comunicaciones (TIC), sería una posible solución a los problemas existentes.

El uso de las TIC representa una variación notable en la sociedad, y aún más su puesta en práctica en aquellas instituciones que ofertan servicios gastronómicos, pues sin lugar a dudas las mejoras en cuanto a tiempo y recursos son evidenciadas con la práctica.

Se define como **idea a defender** del presente trabajo: La elaboración de un sistema informático para gestionar y planificar la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos, permitirá agilizar este proceso, así como proveer un manejo de datos seguro.

Para dar solución al problema planteado anteriormente se define el siguiente objetivo general:

**El objetivo general es** elaborar un sistema informático para la gestión de la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos.

Para alcanzar este objetivo se precisa la realización de diferentes tareas dentro de la investigación, como son:

- Análisis de las particularidades del servicio de alimentación en la Residencia de Postgrado.
- Diseño de una arquitectura que responda a las características propias del trabajo en cuestión.
- Entrevista a los trabajadores de la Residencia de Postgrado de la Universidad de Cienfuegos que intervienen en la gestión y planificación del servicio de alimentación, con el fin de identificar las particularidades de este proceso.
- Estudio de las aplicaciones existentes relacionadas con el problema a resolver.
- Selección de las herramientas, metodologías, lenguajes y tecnologías para la implementación de la aplicación.
- Diseño de la interfaz gráfica de la aplicación.
- Implementación del diseño propuesto para la aplicación y los requerimientos funcionales.
- Determinación de la factibilidad del proyecto.
- Documentación de la información generada durante el análisis, diseño e implementación del sistema.
- Diseño de los casos de pruebas funcionales del sistema.

La puesta en acción de esta propuesta permitirá a la Residencia lograr el mejoramiento y control del servicio de alimentación, evitando la ocurrencia de errores en la confección de los reportes, además de facilitar el trabajo en la elaboración de estos. Agilizará considerablemente el tiempo empleado para el procesamiento de la información y se adaptará a las condiciones de trabajo de la entidad; lo cual logrará una mayor comodidad y satisfacción por parte de los usuarios.

El presente trabajo se estructura en tres capítulos, los cuales hacen referencia a:

Capítulo I: Fundamentación Teórica: En este capítulo se explica la fundamentación teórica del tema y los conceptos asociados al dominio del problema; el problema a resolver y el campo de acción donde se desarrolla, expone las tecnologías, lenguajes y metodologías utilizadas para su desarrollo teniendo en cuenta las tendencias actuales.

Capítulo II: Descripción y construcción de la solución propuesta: Este capítulo describe el proceso llevado a cabo para la construcción de la solución a la problemática planteada. Describe de forma general el funcionamiento de la aplicación. Define requerimientos funcionales y no funcionales y describe el modelo de casos de uso, el diseño y la implementación del sistema.

Capítulo III: Estudio de factibilidad y diseño de casos de prueba funcional: En este capítulo se realiza un estudio de la factibilidad de la aplicación que se propone, teniendo en cuenta el análisis de los costos, beneficios y planificación para el desarrollo del mismo. Además se diseñan los casos de prueba funcional para el sistema y se define la forma de verificar su correcto funcionamiento.

## *Capítulo I: Fundamentación teórica.*

### **Introducción del Capítulo.**

En el presente capítulo se brinda la base teórica y conceptual para el desarrollo del sistema de gestión del servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos. Como aspectos esenciales se abordan los principales conceptos relacionados con el dominio del problema y el campo de acción, los antecedentes existentes de sistemas similares tanto a nivel nacional como internacional.

Se realiza un estudio sobre las metodologías, tecnologías, lenguajes y herramientas que se pueden emplear para la solución del mismo, seleccionando las más adecuadas.

### **1.1 Uso de las Tecnologías de la Información y las Comunicaciones (TIC) en los servicios gastronómicos.**

Las tecnologías de la información y la comunicación (TIC) agrupan los elementos y las técnicas usadas en el tratamiento y la transmisión de las informaciones, principalmente de informática, internet y telecomunicaciones. Optimizan el manejo de la información y el desarrollo de la comunicación, además de permitir actuar sobre la información y generar mayor conocimiento e inteligencia. [2]

En la sociedad del conocimiento, la información ya no es estanca, se comparte, se transforma en cualquier lugar del mundo, se enriquece y se vuelve a compartir, con resultados impredecibles y sorprendentes, situación que para la gastronomía ofrece una oportunidad de ir más allá de lo imaginado. Descubrir y obtener conocimiento está dejando de ser una actividad individual o de un grupo pequeño y está pasando a ser un hecho de masas y una actitud. Es aquí, en la era del crowdsourcing, donde la gastronomía y las TIC podrían fusionarse e ir de la mano.[3]

El empleo de las TIC en la prestación de servicios gastronómicos no solo permite la comercialización de productos de este tipo vía online, sino que abarca una inmensa área, que incluye desde el control de la producción diaria de las cocinas y la combinación de las comidas de acuerdo a la estacionalidad de los géneros alimenticios y nutrientes recomendados a los comensales; hasta la generación de informes comparativos entre los menús planificados y realizados, evitando desperdicios y el aumento de costos.

Esta situación resulta ventajosa para toda aquella institución que contemple la administración de cocinas industriales y restaurantes, así como instituciones hoteleras ya sea con fines turísticos o no.

Las universidades son uno de los tipos de institución que pueden brindar este tipo de servicios, con el objetivo de apoyar los procesos que en ellas se desarrollan. Un ejemplo de ello, es la Universidad de Cienfuegos, pues esta cuenta con un Hotel Universitario conocido como Residencia de Postgrado. Para esta entidad se hace imprescindible plantearse como uno de sus objetivos elevar el nivel de informatización para responder a las exigencias de los servicios que brinda, tomando como algunas de las medidas a ejecutar la implantación de nuevos sistemas que les permita gestionar información para el buen funcionamiento de la entidad, permitiéndole estar a la altura del mundo tecnológico moderno.

Uno de estos servicios es el de alimentación, lo que le permitirá entre otras ventajas poder utilizar nuevas técnicas, teniendo en cuenta las normas establecidas por la Dirección Provincial de Gastronomía y Servicios. La elaboración de un sistema informático que gestione toda esta información tendrá gran importancia para la entidad, debido a que un buen funcionamiento del mismo, se revertirá en una buena organización del trabajo en la actualización de las informaciones, así como el manejo de recursos y datos de forma rápida y confiable.

## **1.2 Conceptos asociados al dominio del problema**

La Universidad de Cienfuegos, localizada en la provincia de Cienfuegos, tiene como misión garantizar la formación integral y la superación continua de los profesionales que demanda la sociedad. Además, consolida, desarrolla y promueve la ciencia, la innovación y la cultura acorde con las exigencias del desarrollo sostenible del territorio y el país.[1]

Para cumplir con su misión en la universidad se desarrollan 3 tipos de procesos:[1]

Procesos directrices o de anticipación, dentro de los cuales se destaca la planeación estratégica, su implantación y el control de su gestión.

Procesos misionales:

- Proceso de proyección social, que incluye los programas de impacto social y los vínculos con la comunidad.
- Proceso docente educativo.
- Ciencia e innovación.

Procesos de apoyo:

- Gestión de recursos humanos
- Gestión de servicios universitarios
- Gestión de recursos financieros
- Gestión de conocimientos e información
- Relaciones internacionales

El proceso de Gestión de servicios universitarios es dirigido por la Dirección general de administración y servicio, que atiende las direcciones y departamentos siguientes:

- Dirección de mantenimiento e inversiones
- Departamento de mantenimiento constructivo
- Departamento de mantenimiento y equipos
- Dirección de aseguramiento material y transporte
- Departamento de aseguramiento técnico material
- Departamento de transporte
- Dirección de servicios internos
- Dirección de alojamiento y alimentación
- Subdirección de Residencia de Postgrado
- Departamento de alimentación ( Sede Carlos Rafael Rodríguez y Sede Conrado Benítez)

La dirección de alojamiento y alimentación incluye la subdirección Residencia de Postgrado. Esta última tiene como misión brindar servicios de hospedaje y alimentación, que garanticen el funcionamiento adecuado de todas las áreas de la universidad; con los aseguramientos necesarios, la calidad requerida y la presencia de los valores éticos y morales, en cumplimiento de las normas y regulaciones establecidas en la legalidad socialista. Además se encarga de:[1]

- Garantizar el funcionamiento de la infraestructura para el desarrollo de los servicios y actividades que se requieran en el centro.
- Organizar, dirigir y controlar las actividades de la Residencia de Postgrado.
- Garantizar con la calidad e higiene requerida, los servicios de alimentación y hospedaje, necesarios para el funcionamiento del centro.

- Desarrollar una estrategia de comercialización que permita el cumplimiento del presupuesto de ingresos previstos para el año.
- Garantizar el control interno contable y administrativo de los recursos humanos, materiales y financieros del área, así como las informaciones que del mismo se deriven.
- Instruir y capacitar a los trabajadores, con el fin de brindar un servicio en correspondencia con la calidad hotelera requerida.

El procedimiento a seguir para los huéspedes que participen en eventos, pregrados y postgrados, se alojen en la Residencia, es el establecido por el Ministerio de Educación Superior Cubano.

### **1.3 Flujo del proceso y análisis crítico**

Como se describió anteriormente una de las funciones fundamentales que tiene la Residencia es brindar el servicio de alimentación a sus huéspedes. Para ello se controla, por el almacenero de la Residencia, el flujo de los productos que entran y salen del almacén. Dichos productos son utilizados para la confección de platos, ofertados a los clientes a través de los menús.

El planificador del menú recibe de carpeta el parte de los comensales y la sugerencia de posible menú según la dirección de la Residencia. Con esta información planifica el menú teniendo en cuenta las normas establecidas, proceso que se debe ejecutar diariamente. En caso de no tener existencia suficiente de algún producto, se convenia con la dirección de la Residencia, la modificación del menú.

Luego de planificado se emite una hoja de cálculo con las normas para la confección de los platos que pasa al área de cocina. Se emite también un modelo llamado Inventario a Precio de Venta (IPV) que indica la cantidad de productos a extraer para la confección del menú y la existencia que debe quedar de estos en almacén.

El almacenero entrega los productos para la confección del menú a cocina, quienes se encargan de elaborar los platos siguiendo las normas indicadas en las hojas de cálculos y posteriormente el menú es ofertado al cliente, finalizando así el proceso.

La existencia de diferentes tipos de clientes conlleva a la planificación de un número significativo de menús al día, en los cuales pudiera variar la norma de gramaje por producto según el nivel del cliente. Para la confección de un menú se hace necesario además de la selección de sus platos, relacionar los ingredientes que requiere el plato así como los productos en cuestión que serán utilizados como ingredientes, especificando en cada caso la cantidad. Este proceso es realizado en parte de forma manual y en parte controlado por medio de una hoja de cálculo Excel.

Llevar el control de los productos existentes en el almacén, se convierte en una tarea engorrosa para el almacenero de la Residencia, pues la confección de cada menú, implica la salida del almacén de aquellos productos empleados en la elaboración del mismo; proceso en el que en reiteradas ocasiones se detectan inconsistencias al comparar la cantidad real en el almacén con lo expuesto en el documento.

Al finalizar el mes o en un período de tiempo determinado, el planificador debe elaborar un reporte con los costes que implican la elaboración de los menús en la fecha dada. Debido al gran volumen de información referente a la planificación de varias variantes de menús al día, esta operación se complejiza, lo cual trae consigo la demora de la entrega de dichos reportes, tal es el caso del IPV diario.

#### **1.4 Análisis de los sistemas informáticos existentes**

En el mundo existen varios sistemas informáticos para la gestión de la información referente al servicio de alimentación, evidenciando de esta forma el uso de las TIC en la prestación de servicios gastronómicos.

A continuación se abordarán los más relevantes respecto al tema.

### **1.4.1 Sistemas informáticos existentes a nivel internacional**

#### **SIPRO**

Next Computer Services ha desarrollado la plataforma SIPRO que ofrece soluciones de control de la producción para las empresas de catering/colectividades. [4]

SIPRO presenta una administración integrada en un entorno WEB de la producción de las cocinas, garantizando la trazabilidad completa en todas las fases y la reducción de costes mediante el control del consumo. Para ello se realiza la gestión de los almacenes, de pedidos a proveedores, de órdenes de cocina, de elaboración de recetas, de costes por menú, etc.

SIPRO incluye numerosos módulos para el control de la producción diaria de las cocinas, como por ejemplo: Compras, Almacenes, Configurador de menús y Cálculo de costes de menús. Estos módulos permiten desde la gestión de los ingredientes y pedidos de mercancías hasta el control de ajustes de inventario y la valoración de existencias.

SIPRO es un software multiplataforma y de carácter privativo. [4]

#### **TecFood**

Comprendiendo las necesidades tecnológicas del mercado de foodService y aprendiendo sus matices, TEKNISA lleva produciendo soluciones específicas para cada área. Para la administración de cocinas industriales y restaurantes, ha desarrollado el software TecFood. Con el sistema, además de promover la reducción de costos se puede obtener otros beneficios que también son de gran importancia como: más facilidad en la visualización de las operaciones, más oportunidades para la gestión de la calidad de proveedores, del proceso y del producto final, optimización del tiempo de los gerentes, nutricionistas, compradores, almacenadores y transparencia de la administración. Y, para acompañar el crecimiento de su empresa, se desarrolló el sistema TecFood en plataforma web, que permite el control de todas

las sucursales desde un sólo lugar. Dentro de sus principales funcionalidades se encuentran: [5]

- Gestionar la producción de las comidas, a través de fichas técnicas y servicios.
- Elaborar el menú y al mismo tiempo realizar el análisis de todos los costos para la producción del mismo.
- Combinar las comidas de acuerdo a la estacionalidad de los géneros alimenticios y nutrientes recomendados a los comensales.
- Proporcionar una gestión centralizada en la matriz.
- Facilitar la normalización de todo el proceso con el acceso web de las sucursales.
- Mostrar el menú mensual durante su elaboración.
- Generar informes comparativos entre el menú planificado y realizado, evitando desperdicios y el aumento de costos.

#### **1.4.2 Sistemas informáticos existentes nacionalmente**

##### **PEEM**

La aplicación informática PEEM Programa de Elaboración y Evaluación de Menús está destinada al diseño, planificación, gestión y control de los menús alimentarios servidos en las instituciones hospitalarias de salud. La aplicación descrita, responde a una utilización laboral, es de tipo cliente-servidor, orientada a un entorno multiusuario, y se puede instalar sobre una intranet. La aplicación se desarrolló con tecnología PHP-MySQL orientada principalmente al cliente Internet Explorer versión 6.0 o superior, de Windows (Microsoft, Estados Unidos). Ofrece menús interactivos para la captación del dato necesario; permite la emisión de los reportes requeridos por la gestión administrativa; y ofrece resúmenes estadísticos diarios y acumulados a los fines de evaluación del proceso. La aplicación hace posible la automatización del trabajo técnico-administrativo del Departamento hospitalario de Dietética, y con ello, la

reducción de la ocurrencia de errores durante el proceso de diseño y planificación de menús alimentarios. [6]

### **1.4.3 Análisis comparativo de las soluciones existentes con la propuesta**

Luego del estudio detallado sobre softwares existentes, relacionados con la gestión de la información referente al servicio de alimentación, tanto a nivel internacional como nacional, se obtuvo como resultado que ninguna de las posibles soluciones encontradas incluye todas las funcionalidades necesarias, además de ser muy costosas en el mercado. En el caso de PEEM, se centra más en el trabajo con la planificación de menús alimentarios servidos en las instituciones hospitalarias de salud, permitiendo la automatización del trabajo técnico-administrativo en un Departamento hospitalario de Dietética en la ciudad de La Habana, o sea dichos menús cumplen con restricciones alimenticias, característica que no exigen los menús planificados en la Residencia. Además el proceso de control de los productos en el almacén no está contemplado en el sistema y por tanto ninguno de los reportes que genera este flujo de datos.

De forma general estos sistemas planifican los menús que serán mostrados a los clientes, pero ese no es el objetivo que persigue el sistema propuesto, sino elaborar un menú genérico, establecido una vez que ya los clientes han elegido las ofertas.

El negocio exige la existencia de varios tipos de clientes, lo cual no se tiene en cuenta en ninguno de los sistemas encontrados. Además, las hojas de cálculo emitidas una vez confeccionados los menús del día, deben tener en cuenta la variedad de clientes, es por ello que se elabora una hoja de cálculo por tipo de cliente y menú.

Una vez analizados los resultados obtenidos, se tomó la decisión de realizar un sistema informático que gestione la información referente al servicio de alimentación en la Residencia de Postgrado de la Universidad de Cienfuegos.

La solución propuesta tendrá como principal ventaja sobre cualquier otra existente, que no será un sistema modificado y adaptado a las características de la entidad y

gestionará tanto la entrada y salida de los productos del almacén, como la planificación del menú, teniendo en cuenta todos los detalles necesarios, así como las leyes establecidas por la Dirección Provincial de Comercio, Gastronomía y Servicios que el negocio exige.

### **1.5 Metodologías de Desarrollo de Software**

Un proceso de software detallado y completo suele denominarse “Metodología”. Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, evolutivo, incremental, espiral, entre otros). Adicionalmente una metodología debería definir con precisión los artefactos, roles y actividades involucrados, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto, guías para uso de herramientas de apoyo, etc. Habitualmente se utiliza el término “método” para referirse a técnicas, notaciones y guías asociadas, que son aplicables a una (o algunas) actividades del proceso de desarrollo, por ejemplo, suele hablarse de métodos de análisis y/o diseño. [7]

La comparación y/o clasificación de metodologías no es una tarea sencilla debido a la diversidad de propuestas y diferencias en el grado de detalle, información disponible y alcance de cada una de ellas. A grandes rasgos, si se toma como criterio las notaciones utilizadas para especificar artefactos producidos en actividades de análisis y diseño, las metodologías se clasifican en dos grupos: Metodologías Estructuradas y Metodologías Orientadas a Objetos. Por otra parte, considerando su filosofía de desarrollo, aquellas metodologías con mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales (o también denominadas Metodologías Pesadas, o Peso Pesado). Otras metodologías, denominadas Metodologías Ágiles, están más orientadas a la generación de código con ciclos muy cortos de desarrollo, se dirigen a equipos de desarrollo pequeños, hacen especial hincapié en aspectos humanos asociados al trabajo en equipo e involucran activamente al cliente en el proceso.[8]

En el momento de seleccionar una metodología para aplicar en la construcción de un sistema es necesario tener en cuenta las características del proyecto y del equipo. Una de las características principales a tener en cuenta es la complejidad del sistema a desarrollar, es decir, es necesario valorar la dificultad del proceso a informatizar, la cantidad de requisitos que deben ser implementados en el sistema y la información que se maneja en el proceso.

Teniendo en cuenta que la única documentación que existirá del software será la elaborada durante la investigación, es preciso que esta sea lo más detallada posible a medida que se desarrolla el proceso de construcción del software, para su posterior mantenimiento; por lo que se propone utilizar la metodología de desarrollo Rational Unified Process (RUP).

### **1.5.1 El Proceso Unificado de Desarrollo de Software (RUP)**

RUP es un proceso formal: Provee un acercamiento disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga los requerimientos de los usuarios finales (respetando cronograma y presupuesto). Fue desarrollado por Rational Software, y está integrado con toda la suite Rational de herramientas. Puede ser adaptado y extendido para satisfacer las necesidades de la organización que lo adopte. Utiliza UML como lenguaje de notación. [7]

#### **Características [7]**

- Dirigido por Casos de Uso:

Los casos de uso son los artefactos primarios para establecer el comportamiento deseado del sistema

- Centrado en la Arquitectura:

La arquitectura es utilizada para conceptualizar, construir, administrar y evolucionar el sistema en desarrollo

- Iterativo e Incremental:

Maneja una serie de entregas ejecutables e integra continuamente la arquitectura para producir nuevas versiones mejoradas

- Conceptualmente amplio y diverso.
- Enfoque orientado a objetos.

### **1.6 Arquitectura cliente-servidor**

En el mundo de TCP/IP las comunicaciones entre computadoras se rigen básicamente por lo que se llama modelo Cliente-Servidor, éste es un modelo que intenta proveer usabilidad, flexibilidad, interoperabilidad y escalabilidad en las comunicaciones. [9]

Desde el punto de vista funcional, se puede definir la computación Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma.

En el modelo cliente servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor (hace una petición), y este envía uno o varios mensajes con la respuesta. En un sistema distribuido cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras.

Entre sus principales ventajas se observan: [9]

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema. Esta centralización también facilita la tarea de poner al día datos u otros recursos.
- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).

- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente). Esta independencia de los cambios también se conoce como encapsulación.
- Existen tecnologías, suficientemente desarrolladas, diseñadas para el paradigma cliente/servidor que aseguran la seguridad en las transacciones, la amigabilidad de la interfaz, y la facilidad de empleo.

La arquitectura cliente/servidor genérica tiene dos tipos de nodos en la red: clientes y servidores. Consecuentemente, estas arquitecturas se refieren a veces como arquitecturas de dos niveles o dos capas.

Algunas redes disponen de tres tipos de nodos:

- Clientes que interactúan con los usuarios finales.
- Servidores de aplicación que procesan los datos para los clientes.
- Servidores de la base de datos que almacenan los datos para los servidores de aplicación.

Esta configuración se llama una arquitectura de tres capas y es la utilizada para el desarrollo de la aplicación.

### **1.6.1 Arquitectura de N Capas**

En la actualidad, la tendencia más aceptada es la aplicación de patrones de diseño de arquitectura que dividen la responsabilidad en distintas capas (separación de incumbencias) que interaccionan unas con otras a través de sus interfaces. Se trata de los sistemas denominados multicapa o n-capas. Aplicados a los proyectos web, el modelo más básico es el de aplicaciones 3 capas:[10]

- Presentación

Como su nombre indica, se limita a la navegabilidad y a gestionar todos aquellos aspectos relacionados con la lógica de presentación de la aplicación, como comprobación de datos de entrada, formatos de salida, internacionalización de la aplicación, etc.

- Negocio o dominio

El resultado del análisis funcional de la aplicación viene a ser la identificación del conjunto de reglas de negocio que abstraen el problema real a tratar. Estas son las que realmente suponen el motor del sistema, dado que se basan en el funcionamiento del modelo real. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos el almacenamiento o la recuperación de los mismos.

- Acceso a datos

Esta capa es la encargada de persistir las entidades que se manejan en el negocio, el acceso a los datos almacenados, la actualización, etc., aunque puede ofrecer servicios relacionados con la persistencia o recuperación de información más complejos.

Todas estas capas pueden residir en un mismo equipo. Lo más usual es que haya una multitud de computadoras donde resida la capa de presentación, es decir los clientes de la arquitectura cliente/servidor, y una computadora central que ejerce la función de servidor en dicha arquitectura. Las capas de negocio y de datos pueden residir en un mismo servidor y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más servidores. Así mismo, si el tamaño o la complejidad de la base de datos aumentaran, se puede separar en varios servidores de datos, los cuales recibirán las peticiones del servidor donde reside la capa de negocio.[10]

### **1.7 Patrón de arquitectura de software**

Los patrones arquitectónicos o de arquitectura, son patrones de diseño de software que ofrecen soluciones a problemas de arquitectura de software. Dan una descripción

de los elementos y el tipo de relación que tienen junto a restricciones sobre cómo pueden ser usados. Expresan un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones.[11]

Aunque un patrón arquitectónico comunica una imagen de un sistema, no es una arquitectura como tal. Es más un concepto que captura elementos esenciales de una arquitectura de software. Muchas pueden implementar el mismo patrón y por lo tanto compartir las mismas características.[12]

Uno de los aspectos más importantes de los patrones, es que encarnan diferentes atributos de calidad. Por ejemplo, algunos representan soluciones a problemas de rendimiento y otros pueden ser utilizados con éxito en sistemas de alta disponibilidad.

En este caso, el patrón utilizado para el desarrollo del sistema informático es el Modelo Vista Controlador (MVC).

### **1.7.1 Patrón de Diseño MVC**

El patrón Modelo-Vista-Controlador (MVC) es una guía para el diseño de arquitecturas de aplicaciones que ofrezcan una fuerte interactividad con usuarios. Este patrón organiza la aplicación en tres modelos separados, el primero es un modelo que representa los datos de la aplicación y sus reglas de negocio, el segundo es un conjunto de vistas que representa los formularios de entrada y salida de información, el tercero es un conjunto de controladores que procesa las peticiones de los usuarios y controla el flujo de ejecución del sistema. [13]

Modelo: encapsula los datos de la aplicación, el flujo de datos y la lógica del negocio.

Vista: extrae datos del modelo y les aplica el formato necesario para la presentación.

Controlador: dirige el flujo de la aplicación y las entradas y salidas de datos entre la capa modelo y la vista.

Desarrollar una aplicación siguiendo este patrón de diseño tiene muchas ventajas:[14]

- La aplicación está implementada modularmente.
- Sus vistas muestran información actualizada siempre.
- El programador no debe preocuparse de solicitar que las vistas se actualicen, ya que este proceso es realizado automáticamente por el modelo de la aplicación.
- Si se desea hacer una modificación al modelo del dominio, como aumentar métodos o datos contenidos, sólo debe modificarse el modelo y las interfaces del mismo con las vistas, no todo el mecanismo de comunicación y de actualización entre modelos.
- Las modificaciones a las vistas no afectan en absoluto a los otros módulos de la aplicación.
- MVC es bastante utilizado en la actualidad en marcos de aplicación orientados a objeto desarrollados para construir aplicaciones de gran tamaño; Java Swing, Apache Struts, Microsoft ASP.NET, las transformaciones XSL o incluso los documentos LATEX siguen este patrón de diseño.
- MVC está demostrando ser un patrón de diseño bien elaborado pues las aplicaciones que lo implementan presentan una extensibilidad y una mantenibilidad únicas comparadas con otras aplicaciones basadas en otros patrones.

### **1.8 Sistema Gestor de Base de Datos**

Un Sistema Gestor de Bases de Datos (SGBD) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos. [15]

Las características de un SGBD son: [15]

- **Abstracción de la información:** Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario.
- **Independencia:** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.
- **Redundancia mínima:** Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias.
- **Consistencia:** En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.
- **Seguridad:** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra asegurada frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.
- **Integridad.** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.

- Respaldo y recuperación. Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.

Algunos ejemplos de SGBD son Oracle, DB2, PostgreSQL, MySQL, MS SQL Server, etc. Para el desarrollo de la solución propuesta se utilizará PostgreSQL.

### **1.8.1 PostgreSQL**

PostgreSQL es un Sistema Gestor de Bases de Datos (SGBD) objeto/relacional, que posee licencia BSD, la cual es una licencia original de una distribución de Software: Berkeley Software Distribution y permite el uso del código fuente en software no libre, por lo que el código fuente de PostgreSQL se encuentra disponible en la red, convirtiéndose en uno de los gestores de bases de datos con código abierto más potente en la actualidad. Es un sistema multiplataforma, pudiendo ser utilizado en la mayoría de los sistemas operativos actuales, consta con varias versiones, además de poseer el apoyo de una comunidad de usuarios que colabora activamente con su desarrollo. [16]

Sus características técnicas la hacen una de las bases de datos más potentes y robustas del mercado. Su desarrollo comenzó hace más de 15 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema. [16]

## **1.9 Lenguajes utilizados en el desarrollo del sistema informático**

### **UML**

El Lenguaje de Modelado Unificado (UML, Unified Modeling Language) es la sucesión de una serie de métodos de análisis y diseño orientado a objetos, los cuales aparecen a fines de los 80 y principios de los 90, siendo como su nombre lo indica un lenguaje de modelado y no un método. UML incrementa la capacidad de lo que se puede hacer

con otros métodos de análisis y diseño orientado a objetos. Los autores de UML apuntan también al modelado de sistemas distribuidos y concurrentes para asegurar que el lenguaje maneje adecuadamente estos dominios. [17]

El lenguaje de modelado es la notación (principalmente gráfica) que usan los métodos para expresar un diseño, indicando los pasos que se deben seguir para llegar a un diseño. La estandarización de un lenguaje de modelado es invaluable, ya que es la parte principal del proceso de comunicación que requieren todos los agentes involucrados en un proyecto informático. Si se quiere discutir un diseño con alguien más, ambos deben conocer el lenguaje de modelado y no así el proceso que se siguió para obtenerlo.[17]

### **Lenguaje de programación Java**

Java es un lenguaje orientado a objetos, creado por la empresa Sun Microsystems a principios de los años 90. Presenta varias semejanzas con lenguajes como C, C++ y C#, por lo que el programador que conozca estos lenguaje podrá entender Java con mayor rapidez. Con la programación en Java, se pueden crear los applets, aplicaciones especiales que se ejecutan dentro de un navegador, al ser cargada una página HTML en un servidor web. Por lo general los applets son programas pequeños y de propósitos específicos. [18]

Sun describe al lenguaje Java de la siguiente manera: simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutral, portable, de alta performance, multitarea y dinámico. [19]

Java es un lenguaje respaldado por una empresa como la Sun Microsystems, característica que garantiza la mantenibilidad en el tiempo, además de contar con un equipo de desarrollo y mantenimiento estable, por lo que para la elección de este lenguaje de programación fueron determinantes estas características.

### **1.10 Servidor de aplicaciones**

Un servidor de aplicaciones es una plataforma de middleware para el desarrollo y despliegue de software basado en componentes. Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las computadoras cliente. Un servidor de aplicaciones generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones. [20] Es por ello que fue el elegido como servidor para brindar servicios a la aplicación, JBoss.

JBoss Application Server es uno de los servidores de aplicaciones de código abierto más ampliamente desarrollado del mercado. Por ser una plataforma certificada J2EE, soporta todas las funcionalidades de J2EE 1.4, incluyendo servicios adicionales como clustering, caching y persistencia. JBoss es ideal para aplicaciones Java y aplicaciones basadas en la Web. También soporta Enterprise Java Beans (EJB) 3.0. [20]

### **1.11 Framework, librerías y componentes**

Cuando se hace uso del término framework, se refiere a una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.[14]

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Una vez elegido Java como lenguaje de programación, la elección del conjunto de tecnologías para desarrollar estuvo basada en la capa del negocio, pues en la capa

de presentación la especificación de Java es Java Server Faces, incluyendo librerías como Richfaces, Facelets y Ajax4jsf. En la capa de negocio los frameworks más reconocidos son Spring y Seam. Spring exige por parte del programador gran esfuerzo, pues no permite la comunicación con Hibernate a través de JPA (Java Persistence API) y es por ello que surge Seam, para permitir comunicar la capa de presentación con la capa de acceso a datos sin necesidad de configuración alguna.

Por lo anteriormente planteado el framework elegido fue Seam pues permite el desarrollo de aplicaciones de forma sencilla y ágil.

### **JBoss Seam**

JBoss Seam 2.2.1 es un framework que integra y unifica los distintos estándares de la plataforma Java EE 5.0, pudiendo trabajar con todos ellos siguiendo el mismo modelo de programación. Ha sido diseñado para simplificar al máximo el desarrollo de aplicaciones, basando el diseño en Plain Old Java Objects (POJOs) con anotaciones. Estos componentes se usan desde la capa de persistencia (Modelo), hasta la capa de presentación, poniendo todas las capas en comunicación directa. Seam es el encargado de conectar a Enterprise JavaBeans 3 (EJB3) y Java Server Faces (JSF), ya que estos framework no han sido concebidos para trabajar juntos, el mismo elimina la barrera existente entre estas tecnologías, permite usar EJBs directamente como "backing beans" de JSF y lanzar o escuchar eventos web. [21]

### **Java Server Faces**

Java Server Faces 1.2 (JSF), es un framework de desarrollo para aplicaciones Web en Java, basadas en el patrón MVC. Además es muy sencillo y flexible, siendo capaz de crear componentes propios o utilizar los componentes ya existentes de acuerdo a las necesidades del problema. [22]

Utiliza páginas JSP para generar las vistas, añadiendo una biblioteca de etiquetas propia para crear los elementos de los formularios HTML.

JSF permite desarrollar rápidamente aplicaciones de negocio dinámicas en las que toda la lógica de negocio se implementa en java, o es llamada desde java, creando páginas para las vistas muy sencillas. [22]

### **Richfaces**

RichFaces es una librería de componentes visuales para JSF, escrita en su origen por Exadel y adquirida por Jboss. Además, RichFaces posee un framework avanzado para la integración de funcionalidades Ajax en dichos componentes visuales, mediante el soporte de la librería Ajax4JSF. [23]

Son características de RichFaces las siguientes:[23]

- Se integra perfectamente en el ciclo de vida de JSF.
- Incluye funcionalidades Ajax, de modo que nunca vemos el JavaScript y tiene un contenedor Ajax propio.
- Contiene un set de componentes visuales, los más comunes para el desarrollo de una aplicación web rica (Rich Internet Application), con un número bastante amplio que cubren casi todas nuestras necesidades.
- Soporta facelets.
- Soporta css themes o skins.
- Los componentes de interfaz de usuario de Richfaces vienen preparados para su uso fuera del paquete, así los desarrolladores pueden ahorrar tiempo y disponer de las ventajas anteriormente mencionadas para la creación de aplicaciones Web.
- Es un proyecto open source, activo y con una comunidad también activa.

### **Ajax4jsf**

Ajax4jsf es una librería open source que se integra totalmente en la arquitectura de JSF y extiende la funcionalidad de sus etiquetas dotándolas con tecnología Ajax de forma limpia y sin añadir código Java script. Mediante este framework se puede variar el ciclo de vida de una petición JSF, recargar determinados componentes de la página sin necesidad de recargarla por completo, realizar peticiones al servidor automáticas, control de cualquier evento de usuario, etc. En definitiva Ajax4jsf permite dotar a una aplicación JSF de contenido mucho más profesional con muy poco esfuerzo. [24]

### **Facelets**

Facelets 1.1 es un framework ligero que permite el uso de plantillas en aplicaciones JSF. [25]

Facelets posee características tales como: [25]

- Trabajo basado en plantillas.
- Fácil composición de componentes.
- Creación de etiquetas lógicas a la medida.
- Funciones para expresiones.
- Desarrollo amigable para el diseñador gráfico.
- Creación de librerías de componentes.

### **Hibernate**

Hibernate 3.3 es un framework que realiza el mapeo entre el mundo orientado a objetos de las aplicaciones y el concepto asociado al modelo entidad - relación de las bases de datos utilizados en entornos Java. El término utilizado es ORM (Object Relational Mapping) y consiste en la técnica de realizar la transición de una representación de los datos de un modelo relacional, a un modelo orientado a objetos y viceversa. Hibernate no solo realiza esta transformación, sino que proporciona

capacidades para la obtención y almacenamiento de datos de la base de datos, reduciendo así el tiempo de desarrollo. [26]

## **1.12 Herramientas utilizadas**

### **Visual Paradigm**

Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora), son aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero, estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software. El principal objetivo que persiguen estas aplicaciones informáticas es mejorar la planificación de un proyecto. [27]

La herramienta seleccionada para sustentar el modelado visual de los artefactos fue Visual Paradigm for UML 6.4 Enterprise Edition, que es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, con un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y documentación.

### **Eclipse**

Para la programación y desarrollo del sistema fue seleccionado el Entorno de Desarrollo Integrado (IDE) Eclipse SDK en su versión 3.4.2, con la colección de plugins de JBoss Tools, necesarios para utilizar los frameworks de desarrollo que se proponen, cuenta además con facilidades de refactorización, completamiento de código y es una herramienta multiplataforma que permite la integración. Es una potente y completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. No es más que un entorno de desarrollo integrado (IDE) en el que se encuentran todas

las herramientas y funciones necesarias para trabajar, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar. [28]

### **iReport**

Para la generación de plantillas en la interfaz web, se empleó iReport. La herramienta iReport es un constructor / diseñador de informes visual, poderoso, intuitivo y fácil de usar para JasperReports escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos como cartas, imágenes, sub-informes, etc. Los datos para imprimir pueden ser recuperados por varios caminos, incluso múltiples uniones JDBC, TableModels, JavaBeans, XML, etc.[29]

### **ER/Studio (Embarcadero)**

Esta herramienta se destaca por ser sencilla de aprender y operar. Además, ofrece una interfaz intuitiva y ayuda en línea completa y ordenada. Permite la validación del modelo lógico y físico diseñados (validez de dependencias, relaciones, claves y límites soportados por la base de datos física). Como características avanzadas, se encuentra la generación del modelo físico a partir del modelo lógico y viceversa, además permite generación de código Java para el manejo a través de JDBC (genera el código necesario para realizar el mantenimiento básico de datos).[30]

### **PgAdmin**

PgAdmin es un administrador de base de datos PostgreSQL, una de los más populares en la función de administración de código abierto y plataforma de desarrollo de PostgreSQL, la base de datos Open Source. La aplicación se puede utilizar en Linux, Solaris, Mac OSX y Windows, así como versiones comerciales y derivados de PostgreSQL. PgAdmin es desarrollado por una comunidad de expertos PostgreSQL en todo el mundo y está disponible en más de una docena de idiomas. Es un software libre publicado bajo la licencia de PostgreSQL. [31]

### **Conclusiones del capítulo**

En este capítulo se estudiaron algunos de los sistemas existentes relacionados con la gestión de la información referente al servicio de alimentación y se definieron los principales conceptos asociados al objeto de estudio. Se analizaron los lenguajes, tecnologías, herramientas y metodologías necesarias para el desarrollo del sistema, concluyendo utilizar RUP como metodología de desarrollo del software, UML como lenguaje de modelado, Visual Paradigm como herramienta para modelar la aplicación, Java como lenguaje de programación, JBoss como servidor de aplicaciones y PostgreSQL como sistema gestor de base de datos, con una arquitectura cliente-servidor utilizando el patrón MVC.

## *Capítulo II: Descripción y construcción de la solución propuesta*

### **Introducción al capítulo**

Un buen proceso de desarrollo de software es un elemento crítico para el éxito de cualquier organización. La presencia de un proceso bien definido y bien gestionado, es fundamental para el entendimiento del problema a resolver. Para ello previamente se estableció, con el objetivo de definir los procesos asociados a la organización, la metodología de desarrollo de software RUP, apoyado en UML como lenguaje de modelado; lo cual permitirá obtener una descripción lo más exacta posible de las actividades que se desarrollan en la Residencia de Postgrado de la Universidad de Cienfuegos.

Además de abordar los aspectos anteriores, en este capítulo se describe y analiza el modelo del sistema; identificando los requerimientos funcionales y no funcionales, así como los actores que intervienen en el mismo. Se generan diagramas como: el diagrama de casos de uso del sistema, el diagrama de clases del diseño, el diagrama del modelo físico y lógico de datos y el diagrama de despliegue; los cuales sentarán las bases a la implementación del sistema.

### **2.1 Identificación de los procesos del negocio**

La Residencia de Postgrado para brindar el servicio de alimentación a sus huéspedes, lleva a cabo una serie de actividades que se describen a continuación.

El proceso comienza cuando el almacenero del almacén central de la UCF, se encarga de abastecer con productos el almacén de la Residencia, a través de un vale que controla información como: número del vale, fecha, código de los productos que entraron y la cantidad por cada uno de ellos. Una vez recibido los productos, el almacenero de la Residencia les da entrada al almacén.

La Residencia cuenta con diferentes tipos de clientes como son los huéspedes de primer, segundo y tercer nivel; los clientes que pagan en cuc y los trabajadores. A

cada tipo de cliente se le tiene asignado una norma de gramaje por producto en los platos que consume. El encargado de planificar el menú para desayuno, almuerzo y comida debe tener en cuenta detalles como la existencia de los productos en el almacén, así como que existen ingredientes no básicos, por lo que un plato puede ofertarse sin dichos ingredientes y que existen varios productos que pueden ser usados como un mismo ingrediente.

Es por ello que para planificar el menú, el encargado del proceso debe regirse por el “Compendio de Normas Técnicas para la Prestación de Servicios Gastronómicos”, documento establecido por la Dirección Provincial de Comercio, Gastronomía y Servicios. Este contiene el resultado de revisión práctica de las recetas que los conforman, así como una tabla de rendimiento de las materias primas durante su procesamiento. El Compendio de Normas Técnicas está organizado por grupos, cada cual correspondiente a un tipo específico de producto (alimento) o tipo de preparación (ejemplo, salsas, masas conformadas). Contiene consideraciones técnicas para alguno de estos grupos, como es el caso de los platos a base de huevo, los platos a base de carnes, pescados, aves y mariscos y aquellos cuya cocción fundamental es la freidura. Establece una flexibilización en los componentes no básicos de alimentos ligeros para toda la red de establecimientos gastronómicos de la provincia, y en caso de carecer de los mismos, indica los componentes que podrán sustituirlos.

Luego de planificado el menú el planificador emite una hoja de cálculo con las normas para la confección de los platos, según el tipo de cliente, que pasa al área de cocina. Se emite también un modelo llamado Inventario a Precio de Venta (IPV) que indica la cantidad de productos a extraer para la confección del menú y la existencia que debe quedar de estos en almacén.

Este modelo generado por el planificador sirve al almacenero de guía para efectuar la entrega de los productos para la confección del menú. Luego en la cocina se encargan de elaborar los platos siguiendo las normas indicadas en las hojas de cálculos, para luego ofertarlos a los clientes, finalizando así el proceso.

Teniendo en cuenta la descripción anterior se identifican los siguientes procesos del negocio:

- Recibir productos.
- Planificar menú.
- Realizar pedido.

## **2.2 Reglas del negocio a considerar**

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.[7]

En este caso se deben tener en cuenta las siguientes:

- El almacenero de la Residencia de Postgrado es el único encargado de recibir los productos que llegan del almacén del comedor de la universidad.
- El planificador del menú es el único encargado para la confección del mismo.
- Es necesario respetar para cada cliente la norma de gramaje por producto en los platos que se consuman.
- El planificador del menú es el único autorizado para solicitar el pedido de los productos que se necesitan extraer del almacén, así como el almacenero es el único autorizado para darle salida a dichos productos.
- Para confeccionar los platos que integran el menú se debe tener en cuenta que existen ingredientes no básicos, por lo que un plato puede ofertarse sin dichos ingredientes.
- Al concluir un día la información generada en el IPV debe ser correcta. Esto incluye que la existencia actual de cada producto debe ser igual a la existencia inicial más las

entradas recibidas menos las diferentes salidas, ya sean por movimientos o confección del menú.

- La información generada en las hojas de cálculo de productos normados, debe coincidir con las salidas de los productos a través de los menús según el tipo de cliente.

### **2.3 Modelo de casos de uso del negocio.**

El modelo de casos de uso del negocio describe los procesos de una empresa en términos de casos de uso y actores del negocio en correspondencia con los procesos del negocio y los clientes, respectivamente. El modelo de casos de uso presenta un sistema desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios. Este modelo permite a los modeladores comprender mejor qué valor proporciona el negocio a sus actores. [7]

#### **2.3.1 Actores del negocio.**

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.[7]

Los actores de este negocio se muestran a continuación:

**Tabla 1 Actores del negocio**

<b>Actor del negocio</b>	<b>Descripción</b>
Almacenero del comedor de la UCF	Es el encargado de abastecer a la Residencia de Postgrado con los productos necesarios para

	prestar el servicio de alimentación, obteniendo las ganancias requeridas.
Cliente	Persona a la cual se le ofrecen los servicios que brinda la Residencia de Postgrado.
Planificador	Es el encargado de recibir los productos que son entregados por el almacenero de la residencia para elaborar el menú.

### **2.3.2 Trabajadores del negocio.**

Un trabajador del negocio representa a un individuo, software o hardware que desempeña un rol dentro de las realizaciones de los casos de uso del negocio.[7]

Los trabajadores de este negocio se muestran a continuación:

**Tabla 2 Trabajadores del negocio**

<b>Trabajador del negocio</b>	<b>Descripción</b>
Almacenero de la Residencia.	Es el encargado de recibir los productos que entrega el almacenero del comedor de la universidad, darles entrada y posteriormente entregarlos al planificador del menú cuando este los solicite.
Planificador del menú.	Es el encargado de planificar el menú para los clientes, así como de confeccionar el pedido para

	solicitar los productos necesarios, al almacenero de la Residencia.
Directora de la Residencia	Es la encargada de proponer el menú del día y en caso de existir problemas, buscar una solución.
Carpetera	Es la encargada de elaborar un informe que muestra el tipo y la cantidad de comensales para los cuales se confeccionará el menú.

### 2.3.3 Diagrama de casos de uso del negocio.

El diagrama de casos de uso del negocio se construye para lograr una visión general de los procesos de negocio llevados a cabo en la organización; en él se representa cada uno de los procesos como un caso de uso, los cuales se relacionan con los actores del negocio.

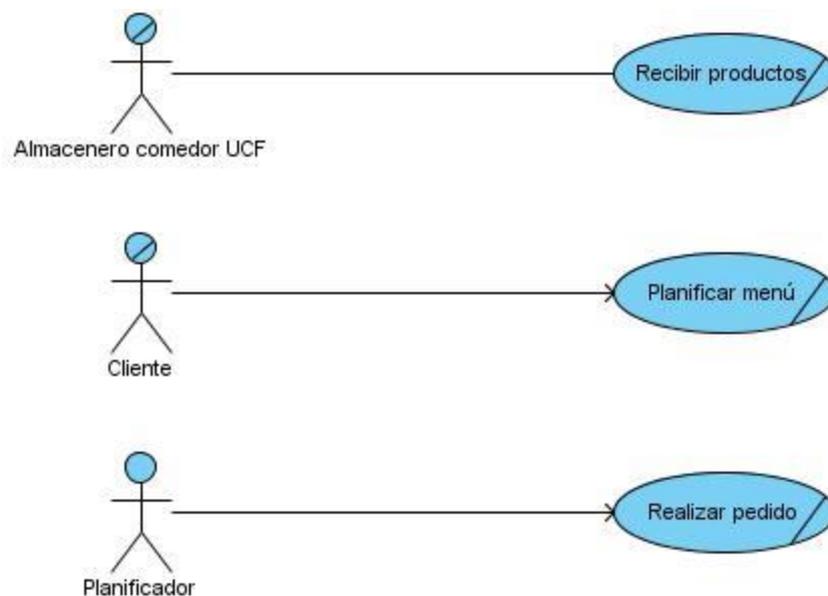


Figura 1 Diagrama de casos de uso del negocio

### **2.3.4 Descripciones de los casos de uso del negocio.**

Para comprender como intervienen los trabajadores en cada proceso no es suficiente con la representación gráfica del diagrama de casos de uso del negocio.

La descripción textual y el diagrama de actividad asociado a cada caso de uso facilitan el entendimiento del proceso. Ver Anexos A y B según muestra la siguiente tabla:

**Tabla 3 Casos de uso del negocio**

<b>Caso de uso del negocio</b>	<b>Descripción textual</b>	<b>Diagrama de actividad</b>
Recibir productos.	Anexo A.1	Anexo B.1
Planificar menú.	Anexo A.2	Anexo B.2
Realizar pedido.	Anexo A.3	Anexo B.3

### **2.4 Modelo de objeto del negocio.**

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo.[7]

En la figura se muestra el diagrama del modelo de objeto del negocio estudiado.

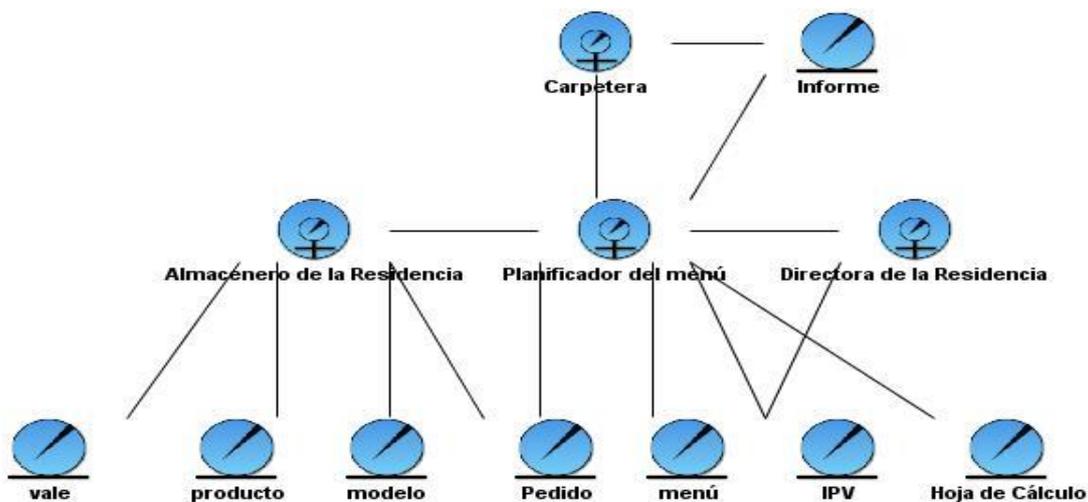


Figura 2 Modelo de objeto del negocio

## 2.5 Descripción del modelo del sistema.

El sistema informático propuesto estará constituido por tres módulos, uno para el control de las acciones referentes al menú, otro para aquellos procesos que se realizan en el almacén y el de administración. En su arquitectura será concebido como una aplicación Web de 3 capas.

El módulo Menú permitirá al planificador gestionar aquellas entidades necesarias para la confección de un menú, así como el propio menú. Además contiene reportes de los costos que implican la confección de un menú o varios.

El módulo Almacén permite dar entrada a los productos al almacén a través de la confección de vales, así como conocer la cantidad de estos para la confección de un menú. Contiene las opciones que permiten generar tanto el IPV como las hojas de cálculo que muestran las normas de los productos por platos para cada nivel.

Por último, el módulo Administración, permitirá el inicio y cierre de una nueva fecha de trabajo, así como el control de usuarios y la asignación de funcionalidades.

### **2.5.1 Requerimientos funcionales.**

Tener una comunicación positiva entre los usuarios y el equipo de desarrollo del software que propicie un entendimiento completo de lo que se tiene que hacer, es la clave del éxito en la producción de software. Durante años muchas aplicaciones han fracasado debido a la existencia de desacuerdos entre lo que el usuario quería, lo que necesitaba, lo que interpretaba cada miembro del equipo de proyecto y lo que realmente se obtenía. Aquí radica la importancia que tiene la identificación de los requerimientos como parte del proceso de desarrollo del software.

Los requerimientos funcionales son declaraciones de los servicios o funciones que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares.[32]

Los requerimientos funcionales se muestran a continuación.

- 1- Mostrar la existencia de los productos en el almacén.
- 2- Exportar a pdf el reporte de la existencia de los productos en el almacén.
- 3- Exportar a Excel el reporte de la existencia de los productos en el almacén.
- 4- Confeccionar vale para la recepción de mercancía.
- 5- Crear vales provisionales para aquellos productos que son confeccionados con ingredientes existentes en el almacén.
- 6- Mostrar dado un menú los productos que deben ser sacados del almacén.
- 7- Exportar a pdf el reporte de los productos que deben ser sacados del almacén dado un menú.
- 8- Exportar a Excel el reporte de los productos que deben ser sacados del almacén dado un menú.
- 9- Mostrar dada una fecha los productos que deben ser sacados del almacén.

## *Capítulo II: Descripción y construcción de la solución propuesta*

---

- 10- Exportar a pdf el reporte de los productos que deben ser sacados del almacén dado una fecha.
- 11- Exportar a Excel el reporte de los productos que deben ser sacados del almacén dado una fecha.
- 12- Dar salida a los productos del almacén teniendo en cuenta el tipo de movimiento.
- 13- Determinar el costo diario de los menús confeccionados.
- 14- Exportar a pdf el reporte del costo diario de los menús confeccionados.
- 15- Exportar a Excel el reporte del costo diario de los menús confeccionados
- 16- Determinar el costo de un menú determinado.
- 17- Exportar a pdf el reporte del costo de un menú determinado.
- 18- Exportar a Excel el reporte del costo de un menú determinado.
- 19- Determinar el costo de los menús confeccionados para un período de tiempo.
- 20- Exportar a pdf el reporte del costo de los menús confeccionados en un período de tiempo.
- 21- Exportar a Excel el reporte del costo de los menús confeccionados en un período de tiempo.
- 22- Determinar el costo de los menús confeccionados según la moneda.
- 23- Exportar a pdf el reporte del costo de los menús confeccionados según la moneda.
- 24- Exportar a Excel el reporte del costo de los menús confeccionados según la moneda.

- 25-       Mostrar IPV.
- 26-       Exportar a Excel el IPV.
- 27-       Exportar a pdf el IPV.
- 28-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los clientes que pertenecen al nivel 1.
- 29-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los clientes que pertenecen al nivel 2.
- 30-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los clientes que pertenecen al nivel 3.
- 31-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los clientes que pagan en CUC.
- 32-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los trabajadores.
- 33-       Mostrar una hoja de cálculo con las normas para la elaboración de los platos de los menús confeccionados para los menús de tipo Testigo.
- 34-       Iniciar fecha.
- 35-       Cerrar fecha.
- 36-       Elaborar menú.
- 37-       Mostrar menús confeccionados en una fecha.
- 38-       Mostrar menús del día.
- 39-       Exportar a pdf la propuesta de menú.
  
- 40-       Cancelar menú.
  
- 41-       Listar productos.
  
- 42-       Insertar productos.
  
- 43-       Modificar productos.
  
- 44-       Eliminar productos.
  
- 45-       Buscar productos.

- 46- Listar platos.
- 47- Insertar plato.
- 48- Eliminar plato.
- 49- Buscar plato.
- 50- Listar ingredientes.
- 51- Insertar ingrediente.
- 52- Modificar ingrediente.
- 53- Eliminar ingrediente.
- 54- Buscar ingrediente.
- 55- Asociar productos a ingredientes
- 56- Listar grupos.
- 57- Insertar grupo.
- 58- Modificar grupo.
- 59- Eliminar grupo.
- 60- Buscar grupo.
- 61- Listar niveles.
- 62- Insertar nivel.
- 63- Modificar nivel.
- 64- Eliminar nivel.
- 65- Buscar nivel.

- 66- Listar usuarios.
- 67- Insertar usuario.
- 68- Eliminar usuario.
- 69- Modificar usuario.
- 70- Buscar usuario.
- 71- Asignar a cada usuario las funcionalidades a las cuales va a tener acceso.
- 72- Cambiar contraseña
- 73- Cerrar sesión.
- 74- Autenticarse.

### **2.5.2 Requerimientos no funcionales.**

Los requisitos no funcionales especifican los criterios que se deben usar para juzgar el funcionamiento de un sistema, en lugar de un comportamiento específico. En general, los requisitos funcionales definen lo que el sistema debería hacer, mientras que los no funcionales verifican cómo un sistema debería de ser. Una buena definición de requerimientos no funcionales es tan importante como los funcionales. Deben de ser apropiadamente definidos, analizados y trazados. Los requisitos no funcionales pueden marcar la diferencia entre un producto bien aceptado y uno con poca aceptación.[32]

Existen múltiples categorías para clasificar a los requerimientos no funcionales, siendo las siguientes representativas de un conjunto de aspectos que se deben tener en cuenta.

**Apariencia:** El software debe tener una interfaz agradable y de fácil comprensión, dirigida a agilizar el proceso con tan solo dominar elementos básicos de informática. Se utilizará el color azul en varias tonalidades en el diseño del sistema, así como la

letra Forte. El logo estará en correspondencia con el de la Universidad de Cienfuegos pues el Hotel Universitario pertenece a dicha entidad. Todos los textos y mensajes en pantalla aparecerán en el idioma correspondiente al país.

**Usabilidad:** El software estará dirigido a los usuarios registrados. Tendrá bien delimitado los privilegios para cada uno de ellos.

**Hardware:** Para utilizar el sistema se necesitará del lado del cliente, como mínimo, máquinas con procesador Pentium, 256 MB de RAM (recomendado 512 MB), 1GB de disco duro (recomendado 5GB). Del lado del servidor se necesitan máquinas con procesador Dual Core, 1 GB de RAM como mínimo, 10GB de disco duro.

**Requerimientos de software:** El sistema debe correr en sistemas operativos Windows, Unix y Linux, utilizando la plataforma JAVA (Java Virtual Machine, JBoss AS y PostgreSQL). Deberá disponer de un navegador web, estos pueden ser Google chrome 1 y Firefox 2.0 o superior.

**Seguridad:** Se garantizará control estricto sobre la seguridad de la información teniendo en cuenta el establecimiento de niveles de acceso. Serán denegados accesos no autorizados. Se implementará en las interfaces de usuario el uso de campos obligatorios y validaciones garantizando la integridad de la información introducida. La entrada al sistema a través de la copia de la dirección de la url en otro navegador no será válida.

El sistema procurará gran importancia al tiempo de autenticación de un usuario, limitando y controlando así la entrada y/o uso por personas no autorizadas. Para mayor seguridad las contraseñas de acceso serán encriptadas en la base de datos a través del mecanismo de cifrado MD5, impidiendo de esta forma, que al lograr entrar se tenga acceso a las contraseñas de los usuarios.

Cada usuario solo tendrá acceso a las funcionalidades asignadas por el administrador.

**Fiabilidad:** El sistema deberá estar disponible 24 horas durante los 365 días del año, excepto el tiempo imprescindible para dar mantenimiento a las aplicaciones desplegadas.

**Rendimiento:** El sistema deberá procesar las transacciones en el menor tiempo posible, además de ejecutar de manera exitosa las operaciones que impliquen un elevado nivel de procesamiento y retornen una cantidad elevada de datos.

## **2.6 Modelo de caso de uso del sistema**

El modelo de casos de uso es la técnica más efectiva y a la vez la más simple que emplean los desarrolladores de software para modelar los requisitos del sistema desde la perspectiva del usuario. Consiste en actores (que representan usuarios y otros sistemas que interactúan con el software) y casos de uso (que representan el comportamiento del sistema, los escenarios que el sistema atraviesa en respuesta a un estímulo desde un actor).[7]

### **2.6.1 Actores del sistema.**

Un actor es aquel que interactúa con el sistema, sin ser parte de él y puede asumir el rol que juega una o varias personas, un equipo o un sistema automatizado.[7]

Los actores del sistema propuesto se definen a continuación.

**Tabla 4 Actores del sistema**

<b>Actor del sistema</b>	<b>Descripción</b>
Almacenero	Es el encargado de controlar a través del sistema los datos referentes a la entrada y salida de productos en el almacén, así como de conocer la existencia de los mismos.

Planificador	A través del sistema elabora el menú a ofertar a los clientes así como los reportes generados por el flujo de datos que esta acción implica.
Administrador	Es el encargado de gestionar todo lo relacionado con los usuarios y dar inicio o cierre a una fecha en el sistema.
Usuario	El usuario es un actor genérico que tendrá acceso a aquellas funciones comunes a todos los actores del sistema.

### **2.6.2 Casos de uso del sistema.**

Los actores interactúan y usan el sistema a través de casos de uso. Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. [7]

Los casos de uso del sistema quedan representados por:

**Tabla 5 Casos de uso del sistema**

<b>Caso de Uso</b>	<b>Actor del sistema</b>	<b>Requerimiento al que responde</b>
Mostrar existencia.	Almacenero.	1-3
Entrada de productos.	Almacenero.	4-5

*Capítulo II: Descripción y construcción de la solución propuesta*

Salida de productos.	Almacenero.	6-12
Determinar costos.	Planificador.	13-24
Mostrar IPV	Planificador.	25-27
Generar HC.	Planificador.	28-33
Cierre de fecha.	Administrador.	34
Inicio de fecha.	Administrador.	35
Gestionar menú.	Planificador.	36-40
Gestionar plato.	Planificador.	46-49
Gestionar producto.	Almacenero.	41-45
Gestionar ingrediente.	Planificador.	50-54
Asociar productos a un ingrediente.	Planificador.	55
Gestionar grupo.	Planificador.	56-60
Gestionar nivel.	Planificador.	61-65
Gestionar usuario.	Administrador.	66-70

Asignar funcionalidades por usuario	Administrador	71
Cambiar contraseña.	Usuario.	72
Cerrar sesión.	Usuario.	73
Autenticarse	Usuario.	74

### 2.6.3 Diagrama de casos de uso del sistema.

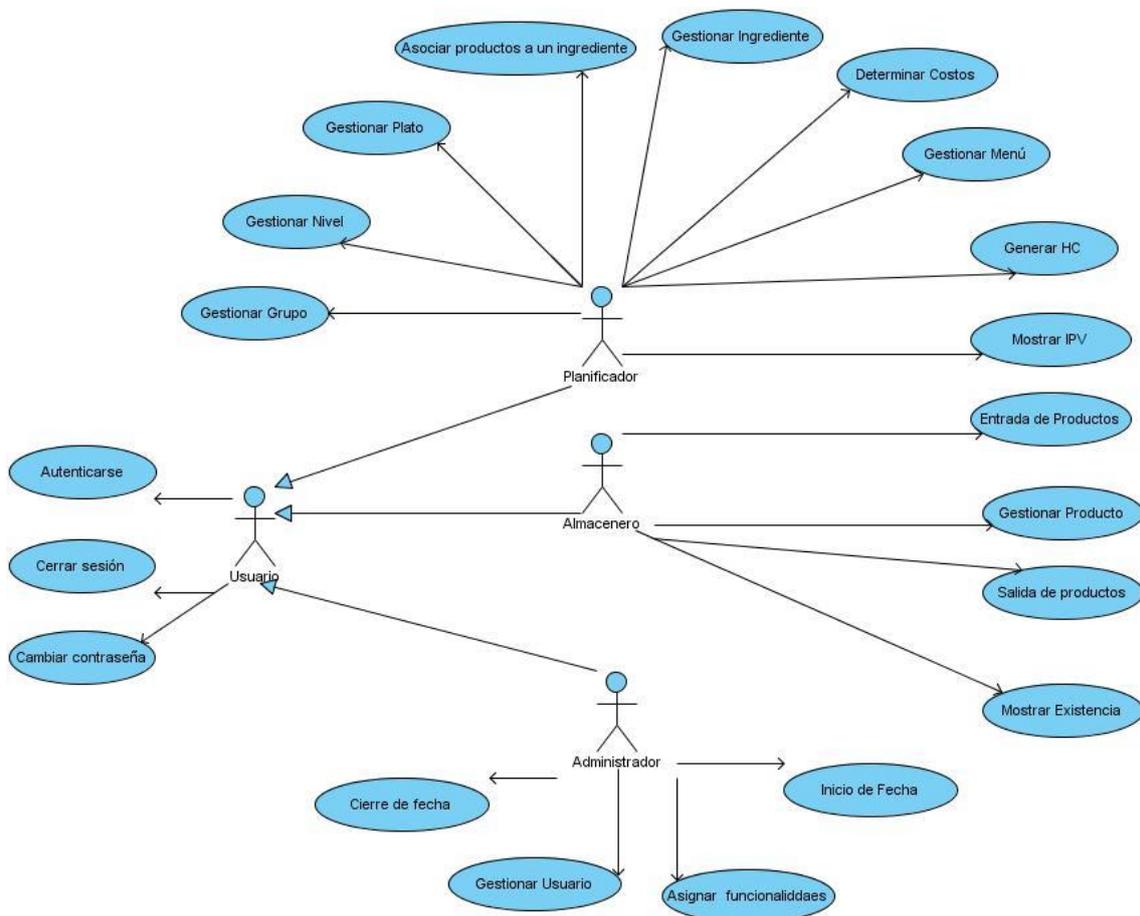


Figura 3 Diagrama de casos de uso del sistema

## 2.6.4 Descripción de los casos de uso del sistema

**Tabla 6 Descripción de los casos de uso del sistema**

<b>Caso de uso del sistema</b>	<b>Descripción</b>
Mostrar existencia.	Anexo C1
Entrada de productos.	Anexo C2
Salida de productos.	Anexo C3
Determinar costos.	Anexo C4
Mostrar IPV.	Anexo C5
Generar HC.	Anexo C6
Cierre de fecha.	Anexo C7
Inicio de fecha.	Anexo C8
Gestionar menú.	Anexo C9
Gestionar plato.	Anexo C10
Gestionar producto.	Anexo C11

Gestionar ingrediente.	Anexo C12
Asociar productos a un ingrediente.	Anexo C13
Gestionar grupo.	Anexo C14
Gestionar nivel.	Anexo C15
Gestionar usuario.	Anexo C16
Asignar funcionalidades por usuario	Anexo C17
Cambiar contraseña.	Anexo C18
Cerrar sesión.	Anexo C19
Autenticarse	Anexo C20

## **2.7 Construcción del sistema.**

Para describir la construcción de la solución propuesta se utilizan los diagramas de clases del diseño, artefacto propuesto por la metodología de desarrollo de software RUP. Otro diagrama que muestra el funcionamiento del sistema es el de despliegue, así como el diseño del modelo lógico y físico de la base de datos. Es por ello que a continuación se detallará cada uno de ellos.

### **2.7.1 Diagrama de clases del diseño.**

Un diagrama de clases de diseño presenta las clases del sistema con sus relaciones estructurales y de herencia correspondientes. En el caso de las aplicaciones web,

representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase.[7]

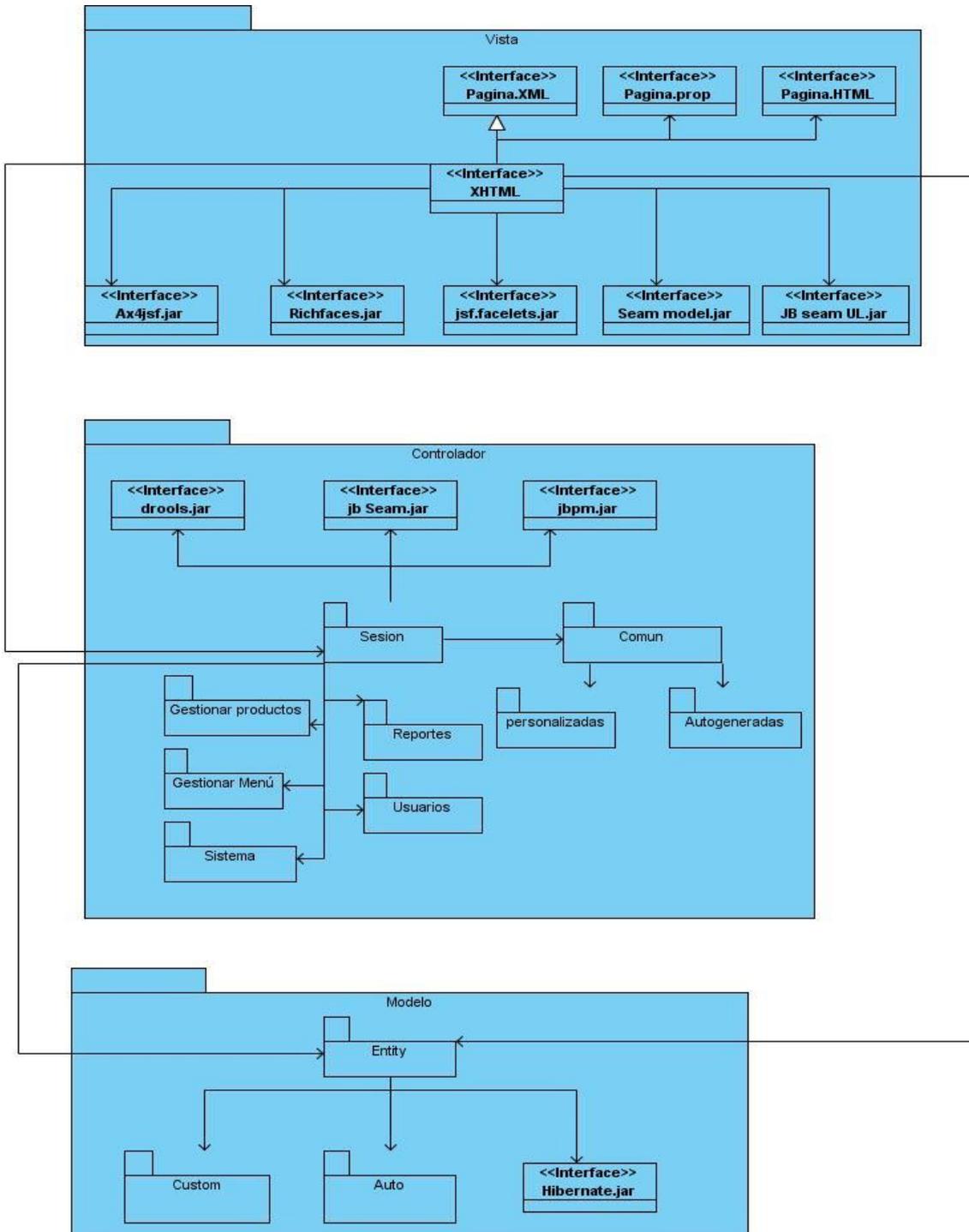


Figura 4 Diagrama de funcionamiento del framework Seam

**Tabla 7 Diagrama de clases web**

<b>Caso de uso</b>	<b>Diagrama de clases web</b>
Mostrar existencia.	Anexo D1
Entrada de productos.	Anexo D2
Salida de productos.	Anexo D3
Determinar costos.	Anexo D4
Mostrar IPV.	Anexo D5
Generar HC.	Anexo D6
Cierre de fecha.	Anexo D7
Inicio de fecha.	Anexo D8
Gestionar menú.	Anexo D9
Gestionar plato.	Anexo D10
Gestionar producto.	Anexo D11

Gestionar ingrediente.	Anexo D12
Asociar productos a un ingrediente.	Anexo D13
Gestionar grupo.	Anexo D14
Gestionar nivel.	Anexo D15
Gestionar usuario.	Anexo D16
Asignar funcionalidades por usuario	Anexo D17
Cambiar contraseña.	Anexo D18
Cerrar sesión.	Anexo D19
Autenticarse	Anexo D20

### **2.7.2 Diseño de la base de datos.**

El sistema maneja datos de gran importancia para la Residencia de Postgrado de la Universidad de Cienfuegos, por lo que la realización de un buen diseño de la base de datos para poder almacenar toda la información referente al servicio de alimentación es imprescindible.

### **2.7.2.1 Diagrama del modelo lógico de datos.**

El modelo lógico de la base de datos determina cómo se estructuran los datos de forma lógica mediante tablas y relaciones. Este diseño puede tener también una gran repercusión en el rendimiento de la aplicación.[33] Ver Anexo E1.

### **2.7.2.2 Diagrama del modelo físico de datos.**

El modelo físico de datos incluye todos los aspectos de diseño de un modelo de base de datos que se pueden modificar sin cambiar los componentes de la aplicación.[33] Ver Anexo E2.

### **2.7.3 Modelo de despliegue.**

El modelo de despliegue captura la configuración de los elementos de procesamiento, y las conexiones entre estos elementos en el sistema. Consiste en uno o más nodos (elementos de procesamiento con al menos un procesador, memoria, y posiblemente otros dispositivos), dispositivos (nodos estereotipados con una capacidad de procesamiento en el nivel modelado de abstracción), y conectores, entre nodos, y entre nodos y dispositivos. Mapea procesos dentro de estos elementos de procesamiento, permitiendo la distribución del comportamiento a través de los nodos que son representados. [34]

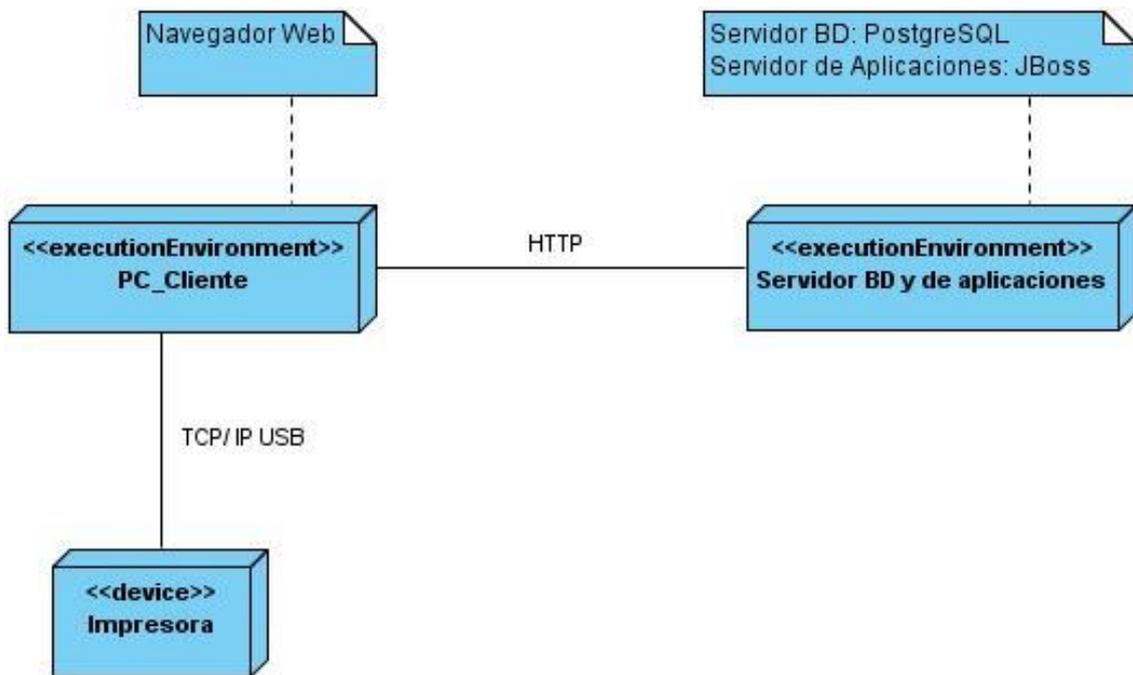


Figura 5 Diagrama de despliegue

## 2.8 Principios de diseño del sistema.

Para desarrollar un sistema de software con calidad se debe prestar cuidadosa atención al diseño, que tiene como objetivo modelar el sistema y encontrar su forma para que soporte todos los requisitos, tanto funcionales como no funcionales. Además permite al equipo de desarrollo de software evaluar la calidad del software antes de implementarlo.

### 2.8.1 Patrones de diseño

Un patrón de diseño es una abstracción de una solución en un nivel alto. Los patrones solucionan problemas que existen en muchos niveles de abstracción. Hay patrones que abarcan las distintas etapas del desarrollo; desde el análisis hasta el diseño y desde la arquitectura hasta la implementación.[35]

Los patrones de diseño que se utilizan en el desarrollo del sistema son los denominados Patrones Generales de Software para Asignación de Responsabilidades (GRASP (General Responsibility Assignment Software Patterns))

Los patrones GRASP describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y explicable. [36]

De todos ellos se destacan principalmente por su utilización en el diseño los patrones:[36]

- Experto: es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto debe recaer sobre la clase que conoce toda la información necesaria para crearlo.
- Creador: el patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que tiene la información necesaria para realizar la creación del objeto, usa directamente las instancias creadas del objeto, almacena o maneja varias instancias de la clase.
- Alta cohesión: expresa que la información que almacena una clase debe ser coherente y está en la mayor medida de lo posible relacionada con la clase.
- Bajo acoplamiento: es la idea de tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

Estos patrones se aplicaron en la definición de cada clase con el objetivo de lograr que las clases realicen las funciones relacionadas con la información que maneja para

que no se sobrecarguen. Además las clases se diseñaron de forma tal que estén lo menos relacionadas entre sí para que los cambios que puedan ocurrir en una clase repercutan lo menos posibles en las otras. Se definieron las clases controladoras, las cuales son las responsables de gestionar las operaciones del sistema. Se le asignó la responsabilidad de instanciar a una clase a aquellas clases que contengan a la misma.

### **2.8.2 Estándares en la interfaz de la aplicación.**

La mayoría de las aplicaciones web desarrolladas usando la plataforma Java Enterprise Edition versión 5.0 hacen uso de los módulos EJB para la lógica de negocio y los módulos o componentes JSF para las interfaces web del sistema (vistas).

EL framework JSF interactúa además con la librería de componentes Richfaces, la cual incluye conversión y validación de campos, establecimiento de reglas de navegación declarativas, la internacionalización y accesibilidad de la interfaz de usuario, un modelo orientado a eventos y combinado con Facelets, lo cual elimina la necesidad de dos motores de renderización mejorando el rendimiento en general además de que provee la capacidad añadida de la tecnología de plantillas de Facelets.

Estos frameworks y librerías fueron los empleados para la creación de la interfaz del sistema, pues Seam, tiene la responsabilidad de unirlos y hacerlos ver como un todo.

Para el diseño de la interfaz se emplearon los colores azules en varias tonalidades. Además cuenta con un menú en la parte superior que permite al usuario el acceso a todas las funcionalidades en todo momento, una parte central donde se muestran todos los formularios siguiendo el mismo principio de diseño y un banner en la parte superior. La entrada de información por parte de los usuarios se realiza a través de los componentes del formulario. Las fuentes utilizadas para los textos son de tipo Arial y tamaño variado según el contexto.

### **2.8.3 Tratamiento de excepciones.**

En una aplicación las excepciones son situaciones que se presentan en tiempo de ejecución afectando el correcto funcionamiento de la misma. Para el tratamiento de

las excepciones en el sistema propuesto se hizo uso de todas las facilidades que brinda la plataforma utilizada.

Igualmente el framework Seam brinda un conjunto de excepciones predefinidas, que permiten capturar estas situaciones desde las clases controladoras correspondientes y mostrarlas mediante la clase FacesMessages. Seam permite además mediante el fichero de configuración page.xml, todo un flujo de navegación basado en excepciones.

#### **2.8.4 Formato de los reportes.**

Para el diseño de los reportes fue empleado el formato de letra Courier New, tamaño 10, pues es una letra legible y de fácil entendimiento. Fueron empleados colores como el blanco y el negro para no recargar y hacer engorrosa su visualización, siempre teniendo en cuenta además los medios con que cuenta la entidad para la impresión de los mismos. Cada reporte tiene un encabezado que lo identifica. En el caso de mostrar números, estos tendrán dos lugares decimales luego de la coma.

#### **2.8.5 Estándares de codificación.**

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código. Para ello se utilizaron en todo momento nombres descriptivos y comentarios informativos. Los nombres de las variables, métodos, consultas y objetos son cortos, claros, y describen su propósito, en idioma español.

Para la documentación del código se utilizó Javadoc, estándar centrado en la interfaz (API - Application Programming Interface) de las clases y paquetes Java.

Documentar el código de un programa es añadir suficiente información como para explicar lo que hace, punto por punto, de forma que no sólo los ordenadores sepan qué hacer, sino que además los humanos entiendan qué están haciendo y por qué.

El paquete de desarrollo Java incluye una herramienta, javadoc, para generar un conjunto de páginas web a partir de los ficheros de código. Esta herramienta toma en consideración algunos comentarios para generar una documentación bien presentada de clases y componentes de clases (variables y métodos).

Aunque javadoc no ayuda a la comprensión de los detalles de código, si ayuda a la comprensión de la arquitectura de la solución, lo que no es poco.

### **Conclusiones del capítulo.**

En este capítulo se describió el flujo del proceso que tiene lugar en la Residencia de Postgrado en la Universidad de Cienfuegos, con el fin de brindar el servicio de alimentación a sus huéspedes, identificando de esta forma los procesos fundamentales para el cumplimiento del servicio. Se identificaron los principales actores y trabajadores que interactúan con el negocio y se elaboraron los modelos de casos de uso y de actividades correspondientes, lo cual posibilitó una comprensión más clara con respecto al problema a resolver, dando paso al modelado del sistema.

Una vez definidos los requerimientos funcionales y no funcionales, se identificaron y describieron los actores del sistema así como los casos de uso a implementar.

Se realizaron los diagramas de clases web, el diagrama del modelo físico y lógico de datos y el diagrama de despliegue. Se definieron además, los principios de diseño del sistema; permitiendo un análisis completo del mismo y creando una guía para la implementación.

## *Capítulo III: Estudio de factibilidad y diseño de los casos de prueba funcional*

### **Introducción al capítulo.**

En este capítulo se realiza el estudio de factibilidad al producto de software, obteniendo una estimación del esfuerzo humano y el tiempo de desarrollo que se requieren para su elaboración, así como los costos del mismo. Se elabora un análisis entre costos y beneficios; con el objetivo de concluir si es o no factible su desarrollo. Para ello se emplea el análisis de planificación por casos de uso. Además se diseñan los casos de prueba funcional en función de lograr el control de la calidad en el desarrollo del software.

### **3.1 Estimación por puntos de casos de uso.**

Existen diferentes métodos de estimación para calcular la factibilidad de un proyecto, como son: Puntos de Función y Puntos por Casos de Uso, del modelo de COCOMO II, sin embargo se decidió utilizar el método de estimación basada en casos de uso, ya que es muy útil en proyectos pequeños, con pocos casos de uso del sistema y es recomendable para su uso que el sistema no cuente con más de 50 casos de uso.

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.[7]

#### **3.1.1 Planificación basada en casos de uso.**

##### **3.1.1.1 Cálculo de los puntos de casos de uso sin ajustar.**

Se calcula a partir de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

Donde,

UUCP: Puntos de casos de uso sin ajustar

UAW: Factor de peso de los actores sin ajustar

UUCW: Factor de peso de los casos de uso sin ajustar

Factor de peso de los casos de uso sin ajustar:

**Tabla 8 Clasificación de los casos de uso**

<b>Caso de uso del sistema</b>	<b>Clasificación</b>
Mostrar existencia.	Simple
Entrada de productos.	Simple
Salida de productos.	Simple
Determinar costos.	Medio
Mostrar IPV.	Medio
Generar HC.	Medio
Cierre de fecha.	Simple
Inicio de fecha.	Simple

Gestionar menú.	Medio
Gestionar plato.	Medio
Gestionar producto.	Medio
Gestionar ingrediente.	Medio
Asociar productos a un ingrediente.	Simple
Gestionar grupo.	Medio
Gestionar nivel.	Medio
Gestionar usuario.	Medio
Asignar funcionalidades a un usuario	Simple
Cambiar contraseña.	Simple
Cerrar sesión.	Simple
Autenticarse	Simple

En la tabla antes mostrada se tienen 10 casos de uso de clasificación simple y 10 de clasificación media por lo que se le aplican como factores de peso 5 y 10 respectivamente.

Por tanto:

$$UUCW = 10*5 + 10*10 = 50 + 100$$

$$UUCW = 150$$

Factor de peso de los actores sin ajustar:

Los actores del sistema son de tipo complejo ya que son personas que interactúan con el mismo mediante una interfaz gráfica, por lo que se les asigna a cada uno de ellos un peso equivalente a 3.

Por tanto:

$$UAW = (\text{Cantidad de actores}) * \text{Peso}$$

$$UAW = 4*3 = 12$$

Como ya se dispone de los valores de factor de peso de actores y casos de uso sin ajustar, es posible obtener el valor de los puntos de caso de uso sin ajustar:

$$UUCP = UAW + UUCW$$

$$UUCP = 12 + 150$$

$$UUCP = 162$$

### **3.1.1.2 Cálculo de Puntos de Casos de Uso ajustados.**

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar éste valor mediante la siguiente ecuación:

$$UCP = UUCP \times TCF \times EF$$

Donde,

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de casos de uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

Factor de Complejidad Técnica (TCF):

El TCF se calcula a través de la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor desde 0 hasta 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

En la tabla que se muestra a continuación se muestra el significado, el peso, el valor asignado y el total:

**Tabla 9 Factor de complejidad técnica**

<b>Factor:</b>	<b>Descripción:</b>	<b>Peso:</b>	<b>Valor:</b>	<b>Peso* Valori:</b>
T1	Sistema distribuido.	2	3	6
T2	Objetivos de performance o tiempo de respuesta.	1	4	4
T3	Eficiencia del usuario final.	1	3	3
T4	Procesamiento interno complejo.	1	2	2
T5	El código debe ser reutilizable.	1	3	3
T6	Facilidad de instalación.	0.5	3	1.5
T7	Facilidad de uso.	0.5	4	2
T8	Portabilidad.	2	4	8
T9	Facilidad de cambio.	1	3	3
T10	Concurrencia.	1	3	3

T11	Incluye objetivos especiales de seguridad.	1	3	3
T12	Provee acceso directo a terceras partes.	1	3	3
T13	Se requieren facilidades especiales entrenamiento a usuarios	1	3	3

$$\Sigma (\text{Peso}_i * \text{Valor}_i) = 6+4+3+2+3+1.5+2+8+3+3+3+3+3 = 44.5$$

El Factor de Complejidad Técnica resulta:

$$\text{TCF} = 0.6 + 0.01 * \Sigma (\text{Peso}_i * \text{Valor}_i)$$

$$\text{TCF} = 0.6 + 0.01 * 44.5$$

$$\text{TCF} = 1.045$$

Factor de ambiente (FE):

El Factor de ambiente se calcula atendiendo a las habilidades y el entrenamiento del grupo involucrado. El procedimiento para su cálculo es similar al cálculo del Factor de complejidad técnica.

**Tabla 10 Factor de ambiente**

<b>Factor:</b>	<b>Descripción:</b>	<b>Peso:</b>	<b>Valor:</b>	<b>Peso<sub>i</sub>* Valor<sub>i</sub>:</b>
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	4	6
E2	Experiencia en la aplicación.	0.5	3	1.5
E3	Experiencia en orientación a objetos.	1	4	4
E4	Capacidad del analista líder.	0.5	4	2
E5	Motivación.	1	5	5

---

E6	Estabilidad de los requerimientos.	2	4	8
E7	Personal part-time.	-1	3	-3
E8	Dificultad del lenguaje de programación.	-1	3	-3

$$\Sigma (\text{Pesoi} * \text{Valori}) = 6+1.5+4+2+5+8-3-3 = 20.5$$

$$EF = 1.4 - 0.03 * \Sigma (\text{Pesoi} * \text{Valor asignadoi})$$

$$EF = 1.4 - 0.03 * 20.5$$

$$EF = 0.785$$

Los casos de uso ajustados resultan:

$$UCP = UUCP * TCF * EF$$

$$UCP = 162 * 1.045 * 0.785$$

$$UCP = 132.89$$

### **3.1.1.3 Estimación del esfuerzo:**

CF: Factor de Conversión

$$CF = 20 \text{ Horas/Hombre}$$

El esfuerzo en horas hombres está dado por:

$$E = UCP * CF$$

$$E = 132.89 * 20 = 2657.8 \text{ Horas/Hombre}$$

Duración:

Trabajando los 25 días al mes y 10 horas al día como promedio, se obtiene: Duración (días) = Total de horas / hombre entre 10 horas al día = 2657.8 / 10

=265.78 días

Duración (meses) = Total de días / 25 días por mes = 265.78 / 25

=10.63 meses ≈ 11 meses

El proyecto se realiza en 11 meses.

**Tabla 11 Esfuerzo de desarrollo del proyecto**

<b>Actividad</b>	<b>% Esfuerzo</b>	<b>Valor Esfuerzo</b>
Análisis	10	265.78
Diseño	20	531.56
Implementación	40	1063.12
Prueba	15	398.6755
Sobrecarga	15	398.67
Total	100	2657.8

#### **3.1.1.4 Cálculo de costos**

Tomando como salario promedio mensual \$315.00

Costo = 11 meses \* \$315 mensual = \$3465.00

#### **3.2 Análisis de los costos y beneficios.**

El sistema informático desarrollado, como resultado del presente trabajo de diploma, no implica costo alguno para la entidad donde se pretende implantar, sin embargo, al desarrollo de todo producto informático va asociado un costo y su justificación económica viene dado por los beneficios tangibles e intangibles que este produce.

Utilizar este nuevo software permite a parte del personal encargado de brindar el servicio de alimentación en la Residencia de Postgrado, gestionar la información referente a los menús ofertados así como todo el flujo de información que surge al crear estos, tal es el caso de la entrada y salida de productos en el almacén. Posibilita conocer el gramaje necesario para la confección de los menús teniendo en cuenta los tipos de huéspedes, todo esto de manera rápida y confiable. Para el desarrollo de este sistema no fue necesario invertir en medios técnicos. Estos beneficios implican un ahorro considerable del tiempo invertido en la gestión y control de esta información.

### **3.3 Diseño de pruebas funcionales**

Un aspecto crucial en el control de calidad del desarrollo de software son las pruebas y, dentro de estas, las pruebas funcionales, en las cuales se hace una verificación dinámica del comportamiento de un sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba. [37]

Para hacer pruebas funcionales se requiere una planificación que consiste en definir los aspectos a chequear y la forma de verificar su correcto funcionamiento, punto en el cual adquieren sentido los casos de prueba.[37]

Para el diseño de los casos de prueba del sistema fueron elegidos los Casos de uso que requirieron mayor entrada de datos por parte del usuario.

#### **3.3.1 Caso de prueba para el Caso de Uso Autenticarse**

Página de ingreso al sistema

Pantalla que valida con un nombre de Usuario y Contraseña el usuario, con el fin de permitir el ingreso al sistema.

**Sistema Informático para la Gestión del Servicio de Alimentación en la Residencia de Postgrado**

Introduzca los datos para ingresar en el sistema.

ⓘ Usuario o contraseña incorrecta. Inténtelo de nuevo.



Nombre Usuario:

Contraseña:

Acceso

**Figura 6 Caso de prueba Autenticarse**

Validaciones:

La Validación se hace en el Evento “Onclick” del Botón Acceso, con las siguientes reglas:

- Usuario: Es una cadena con un mínimo de 5 caracteres. (Obligatorio)
- Contraseña: Cadena de Caracteres con un mínimo de 8. Al menos un carácter numérico, uno especial e incluir mayúsculas y minúsculas (Obligatorio)

Si la validación no tuvo éxito saldrá el mensaje “Usuario o contraseña incorrecta. Inténtelo de nuevo” en la Zona de errores.

Zona de Errores:

Estará ubicada en el centro de la Pantalla encima del recuadro de ingreso.

### 3.3.2 Caso de prueba para el Caso de Uso Entrada de Productos

**!** El número del vale debe ser mayor que 0.

**Entrada de productos**

**Tipo de entrada:**

Vale     Interno     Expediente

Fecha: 05-03-2015

Tipo:

Número

**Buscar**

Productos	Moneda
Sal	MN
Mermelada de Mango	CUC
Papa Prefrita	CUC
Jugo Concentrado Naranja	CUC
Caldo sustancioso huesos cerdo - s...	MN
Arroz	MN
Frijoles blancos	MN
Aceite	MN
Pastas largas	MN
Harina trigo	MN
Condimento	MN
Frijoles negros	MN

Cantidad:

Nombre	Moneda	Cantidad
--------	--------	----------

**Figura 7** Caso de prueba Entrada de productos

La entrada de productos permite actualizar la existencia de estos en el almacén. Para ello se debe especificar el tipo de entrada y atendiendo a esta, introducir los datos en el formulario. El próximo paso sería asignarle a cada producto a entrar, la cantidad y luego presionar el botón Aceptar.

Validaciones:

- La selección de los radiobuttons es obligatoria, estos son excluyentes. Si no se efectúa la selección se mostrará un “\* “al lado del componente.

- El campo Número para el tipo de entrada es un número entero positivo el cual identifica al vale. Si la validación no tuvo éxito saldrá el mensaje “El número del vale es un número entero mayor que 0” en la Zona de errores. (Obligatorio al ejecutarse el evento del botón Aceptar)
- El combobox Tipo, en caso de ser visible, es Obligatoria la elección de uno u otro tipo. Si la validación no tuvo éxito se mostrará un “\*” “al lado del campo.
- El campo cantidad es un número real de doble precisión pero debe cumplir con la condición de ser mayor que 0. Si la validación no tuvo éxito saldrá el mensaje “La cantidad del producto debe ser mayor que 0” en la Zona de errores. (Obligatorio al ejecutarse el evento del botón que contiene el caracter > )
- La lista que contiene el producto con la cantidad debe contener al menos un elemento. Si la validación no tuvo éxito saldrá el mensaje “Por favor inserte cada producto con la cantidad recibida” en la Zona de errores.

En caso de que el usuario deje algún campo en blanco o introduzca letras en aquellos que representan un número, se mostrará un “\*” “al lado del campo.

En el evento del botón Aceptar se muestra un mensaje de éxito o de error según sea el caso en la Zona de Errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.3.3 Caso de prueba para el Caso de Uso Gestionar menú

Acción: Insertar Menú

Por favor, inserte los platos con sus respectivas cantidades antes de continuar.

**Elaborar Menú**

Fecha: 2015-03-05    Tipo: Desayuno    Nivel: CUC

Orden de los platos:  
 Grupo     Oferta     Alfabético

Grupo: --Seleccione--

Buscar

Platos	Plato	Cantidad
Agua natural embotellada		
tasajo2		
Ajiaco		
Alas de Pollo Empanadas		
Alas de Pollo Fritas		
Calabaza Hervida		
Escalope de Filete Empanado		
Jugo de Mandarina		
Papas Fritas		
Parguito Frito		
Refresco de sirope		
tasajo		

Cantidad:

< >

Mostrar existencia

Cancelar    Siguiente

Figura 8 Caso de prueba Gestionar Menú

Crear un menú determina cuál será la oferta de platos para un tipo de cliente y la selección de aquellos productos a utilizar para la confección de los mismos.

Validaciones:

- En los combobox Tipo y Nivel es obligatoria la elección. En caso de no seleccionar alguna de las opciones el sistema muestra un “\*” al lado del campo.
- El campo Cantidad para el plato es un número entero positivo. Si la validación no tuvo éxito saldrá el mensaje “La cantidad del plato debe ser un número entero

mayor que 0” en la Zona de errores.. (Obligatorio al ejecutarse el evento del botón que contiene el caracter > )

- La lista que contiene el plato con la cantidad debe contener al menos un elemento. Si la validación no tuvo éxito saldrá el mensaje “Por favor inserte los platos con sus respectivas cantidades antes de continuar” en la Zona de errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

Al presionar el botón siguiente, si no existieron errores, el sistema muestra la siguiente vista.

The screenshot shows a window titled "Selección de Productos" with a header bar. Below the header, it displays "Fecha: 05/03/2015 Nivel: Nivel 3 Tipo: Almuerzo". The window is divided into three main sections:

- Platos del Menú:** A table with columns "Plato", "Cantidad", and "Listo".

Plato	Cantidad	Listo
Alas de Pollo	3	Si
Empanadas	3	Si
Calabaza Hervida	3	Si
- Ingredientes:** A table with columns "Ingrediente" and "Listo".

Ingrediente	Listo
Harina de trigo	Si
Pan rallado	Si
Huevo	Si
Naranja agria	Si
Alas de pollo	Si
Sal	Si
Grasa	Si
Pimienta molida	Si
Ajo en cabeza	Si
- Productos:** A table with columns "Producto" and "Cantidad".

Producto	Cantidad
Pan MN	<input checked="" type="checkbox"/>
Pan CUC	<input type="checkbox"/>

At the bottom right of the window, there are two buttons: "Cancelar" and "Aceptar".

Figura 9 Caso de prueba Gestionar Menú Siguiete

En esta vista el usuario selecciona para cada plato, en cada ingrediente, el producto a utilizar.

Validaciones:

- Si el menú se planifica contra existencia, es necesario validar que la cantidad a extraer del producto sea menor que la cantidad existente en el almacén. Si sucede lo contrario el sistema modifica el campo Listo y muestra No para el ingrediente y el plato

en que no será posible la extracción. Además muestra un mensaje informando la no existencia del producto.

En el evento del botón Aceptar se muestra un mensaje de éxito o de error según sea el caso en la Zona de Errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.3.4 Caso de prueba para el Caso de Uso Salida de productos

Acción: Insertar movimiento

El número del vale debe ser mayor que 0.

Salida de productos por movimiento

Tipo de movimiento:

Fuera de Norma  Interno  Centro de Costo  Merma

No. Vale:

Justificación:

Productos	Moneda	Cantidad	UM
Mermelada de Mango	CUC	0.0	g
Papa Prefrita	CUC	0.0	g
Jugo Concentrado Naranja	CUC	45.0	g
Caldo sustancioso huesos cerdo - sub...	MN	0.0	g
Arroz	MN	655.52	g
Frijoles blancos	MN	130.0	g
Aceite	MN	1969.5	g
Pastas largas	MN	0.0	g
Harina trigo	MN	377.6	g
Condimento	MN	0.0	g
Frijoles negros	MN	30.0	g
Col	MN	0.0	g

Cantidad:

Nombre	Moneda	Cantidad	UM
Sal	MN	23.0	g

Figura 10 Caso de prueba Salida de productos

Crear un movimiento permite dar salida a los productos del almacén.

Validaciones:

- La selección de los radiobuttons es obligatoria, estos son excluyentes. Si no se efectúa la selección se mostrará un “\* “al lado del componente.
- El campo No. Vale para el tipo de movimiento Interno es un número entero positivo el cual identifica al movimiento. Si la validación no tuvo éxito saldrá el mensaje “El número del vale debe ser un número entero mayor que 0” en la Zona de errores. (Obligatorio al ejecutarse el evento del botón Aceptar)
- El campo cantidad es un número real de doble precisión pero debe cumplir con la condición de ser mayor que 0. Si la validación no tuvo éxito saldrá el mensaje “La cantidad del producto debe ser un número mayor que 0” en la Zona de errores. (Obligatorio al ejecutarse el evento del botón que contiene el caracter > )
- En caso que la cantidad a extraer exceda la cantidad real mostrada en la primera tabla, el sistema mostrará un mensaje de alerta y no permitirá la inserción del producto en la lista de productos a extraer.
- La lista que contiene el producto con la cantidad debe contener al menos un elemento. Si la validación no tuvo éxito saldrá el mensaje “Por favor inserte cada producto con la cantidad a extraer” en la Zona de errores.

En caso de que el usuario deje algún campo en blanco o introduzca letras en aquellos que representan un número, se mostrará un “\* “al lado del campo.

En el evento del botón Aceptar se muestra un mensaje de éxito o de error según sea el caso en la Zona de Errores.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

### 3.3.5 Caso de prueba para el Caso de Uso Gestionar Producto

#### Acción: Insertar producto

The screenshot shows a web form titled "Adicionar Producto". It contains several input fields with asterisks indicating they are required. The fields and their values are: "Código:" with "P001", "Nombre:" with "pimientos", "U medida:" with "U", "Moneda:" with "MN", and "Tipo peso:" with "B". The "Peso unidad:" field contains "-3" and has a red error message: "El peso por unidad del producto debe ser mayor que 0". The "Costo:" field contains "fff" and has a red error message: "El valor debe ser un número mayor que 0". At the bottom left, there is a note: "\* Campos requeridos". At the bottom of the form, there are two buttons: "Aceptar" and "Cancelar".

Figura 11 Caso de prueba Gestionar Producto Acción Insertar producto

#### Validaciones:

- Todos los campos son requeridos en caso de omitir datos en alguno de estos, el sistema mostrará el mensaje: Valor requerido.
- El campo Peso unidad es un número entero positivo. En caso de que el dato insertado no esté incluido en este conjunto numérico, el sistema mostrará al lado derecho del campo el mensaje: El peso por unidad del producto debe ser un número entero positivo.
- El campo Costo es un número real de doble precisión, además debe cumplir con la condición de ser mayor que 0. En caso de que el dato insertado no esté incluido en el conjunto, el sistema mostrará el mensaje: El costo del producto debe ser un número mayor que 0.
- Si el código del nuevo producto a insertar ya existe, el sistema muestra el mensaje: Existe un producto con ese código. La operación ha fallado.

## Acción: Buscar producto

**Buscar Producto**

Código:

Nombre:

U medida:

Moneda:

Tipo peso:

**Listado de Productos**

Código	Nombre	U medida	Moneda	Tipo peso	Peso unidad	Costo	Acción
P001	Sal	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P002	Arroz	g	MN	N	1	0.3	<a href="#">Ver Editar</a>
P003	Azúcar	g	MN	N	1	0.5	<a href="#">Ver Editar</a>
P004	Frijoles blancos	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P005	Aceite	g	MN	N	1	0.6	<a href="#">Ver Editar</a>
P006	Pastas largas	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P007	Harina trigo	g	MN	N	1	0.3	<a href="#">Ver Editar</a>
P008	Condimento	g	MN	N	1	0.5	<a href="#">Ver Editar</a>
P009	Frijoles negros	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P010	Col	g	MN	B	1	0.6	<a href="#">Ver Editar</a>

[Siguiete >](#) [Última >>](#)

Figura 12 Caso de prueba Gestionar Producto Acción Buscar producto

### Validaciones

- Si al usuario introducir el criterio para buscar, el sistema no encuentra el o los productos con los datos introducidos en el formulario, muestra en el panel Listado de productos el mensaje: El producto que busca no existe.

### 3.3.6 Caso de prueba para el Caso de Uso Asociar productos a un ingrediente



Figura 13 Caso de prueba Asociar productos a un ingrediente

Validaciones:

- El combobox ingrediente es requerido, en caso de no elegir algún ingrediente, el sistema muestra el carácter “\*” al lado izquierdo del campo.
- La lista de productos asociados debe tener al menos un elemento para efectuar la asociación. En caso de no cumplir la condición, el sistema muestra en la zona de errores el mensaje: Elija los productos a asociar para el ingrediente seleccionado.

Zona de Errores y de Éxitos: Ubicada en la parte superior de la vista.

#### Conclusiones del capítulo

La realización del cálculo de factibilidad económica para el sistema informático propuesto, estimó un tiempo de 11 meses para su desarrollo y un costo de \$3465 aproximadamente. Se definieron los beneficios que trae consigo la implementación del sistema para la entidad, concluyendo de esta forma que es factible su desarrollo.

Se diseñaron los casos de pruebas funcionales, permitiendo obtener una verificación dinámica del comportamiento del sistema, garantizando de esta forma el control de la calidad en el desarrollo del software.

### Conclusiones

Tomando en consideración los objetivos planteados al inicio del trabajo, se arriban a las conclusiones siguientes:

- Durante el análisis del objeto y campo de acción se identificaron peculiaridades en el proceso de gestión del servicio de alimentación en la Residencia de Postgrado que ameritó el desarrollo de un sistema informático dirigido a agilizar este proceso, así como proveer un manejo de datos seguro.
- Las herramientas, lenguajes y la metodología a utilizar, fueron elegidas en función de garantizar la mantenibilidad y estabilidad en el tiempo, así como proveer una documentación detallada durante el análisis, diseño e implementación del sistema informático.
- Se diseñó e implementó un sistema informático con funcionalidades que se adecuan a las necesidades de la institución, proporcionándole una herramienta que reduce el tiempo empleado en el proceso de gestión del servicio de alimentación, dando cumplimiento de esta forma a los requerimientos planteados.
- El diseño de los casos de prueba funcional permitió obtener una verificación dinámica del comportamiento del sistema, garantizando de esta forma el control de la calidad en el desarrollo del software.
- Utilizando como método de estimación Puntos de Casos de Uso para determinar la factibilidad, se concluyó que fue factible desarrollar el sistema informático.

## *Recomendaciones*

A pesar de que la investigación realizada cumplió con los objetivos trazados, se recomienda:

- Poner a prueba el sistema durante un período de tiempo significativo, para comprobar su desempeño.
- Generalizar su uso al comedor obrero estudiantil y a Residencias de Postgrados de otras universidades en el país.
- Continuar el estudio de los procesos asociados a la entidad con el objetivo de añadirle nuevas funcionalidades al sistema, en función de las transformaciones estructurales que tiene la institución actualmente.

Referencias bibliográficas

- [1] Colectivo de autores, «Reglamento Orgánico-UCF».
- [2] M. A. B. Osmany Aguilera Almaguer y Eduardo Peña Cruz, «Integración de la enseñanza de las TIC en la educación», 03-jul-2011. [En línea]. Disponible en: <http://www.eumed.net/rev/ced/29/abc.htm>. [Accedido: 11-dic-2014].
- [3] Manuel Mundaca, «Lo que se cocina en las TIC | A un Clic de las TIC». [En línea]. Disponible en: <http://www.aunclicdelastic.com/lo-que-se-cocina-en-las-tic/>. [Accedido: 11-dic-2014].
- [4] Next Computer Services, S. A., «Solución Integral de Control de la Producción». [En línea]. Disponible en: <https://drive.google.com/viewerng/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxz2x1Y2lvbmVzcGFyYWNhdGVyaW5nfGd4OjczY2JjMzVhNWZiYzBIZGE>. [Accedido: 24-nov-2014].
- [5] Teknisa Software, «Tecfood Servicios de alimentación». [En línea]. Disponible en: <http://www.teknisa.com/es/productos/tecfood-servicios-de-alimentaci%C3%B3n>. [Accedido: 24-nov-2014].
- [6] G. C. M. Alexander González Domínguez, «Diseño y Desarrollo de una Aplicación Informática para la Elaboración y Evaluación de Menús Hospitalarios», *Rev Cub Aliment Nutr*.
- [7] Ivar Jacobson, Grady Booch, y James Rumbaugh, *El Proceso Unificado de Desarrollo de Software*. .
- [8] «Procesos de Software - Metodologías para el Desarrollo del Software». [En línea]. Disponible en:

- <http://procesosdesoftware.wikispaces.com/METODOLOGIAS+PARA+DESARROLLO+DE+SOFTWARE>. [Accedido: 01-dic-2014].
- [9] Bertha Mariel Márquez Avendaño y José Manuel Zulaica Rugarcía, «Implementación de un reconocedor de voz gratuito a el sistema de ayuda a invidentes Dos-Vox en español», Universidad de las Américas Puebla, Cholula, Puebla, México.
- [10] Daniel Fernández Lanvin, «Definición de una arquitectura software para el diseño de aplicaciones web basadas en tecnología Java-J2EE», Estudios de Doctorado Avances en Informática, Universidad de Oviedo.
- [11] Hanna Oktaba, «Introducción a Patrones», Facultad de Ciencias, UNAM. [En línea]. Disponible en: <http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/patrones.html>. [Accedido: 11-dic-2014].
- [12] N. K. Carlos Reynoso, «Estilos y Patrones en la Estrategia de Arquitectura de Microsoft», mar-2004. [En línea]. Disponible en: <http://carlosreynoso.com.ar/archivos/arquitectura/Estilos.PDF>. [Accedido: 11-dic-2014].
- [13] E. Bascón Pantoja, «El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing», *Rev. Acta Nova*, vol. 2, n.º 4, 2011.
- [14] J. J. Gutiérrez, *¿Qué es un framework web?* 2006.
- [15] Katty Ávila, « ¿Qué es un Sistema Gestor de Bases de Datos o SGBD?». [En línea]. Disponible en: <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>. [Accedido: 01-dic-2014].
- [16] Marc Gibert Ginesta y Oscar Pérez Mora, «Bases de datos en PostgreSQL». [En línea]. Disponible en:

- [http://informatica.gonzalonazareno.org/plataforma/file.php/3/Apuntes/UOC\\_Postgres.pdf](http://informatica.gonzalonazareno.org/plataforma/file.php/3/Apuntes/UOC_Postgres.pdf). [Accedido: 03-feb-2015].
- [17] José Enrique González Cornejo, «El Lenguaje de Modelado Unificado (UML)», ene-2008. [En línea]. Disponible en: <http://www.docirs.cl/uml.htm>. [Accedido: 24-nov-2014].
- [18] John Zukowski, *Java\_2\_J2SE\_1.4*. España: Anaya Multimedia (Grupo Anaya, S.A.), 2003.
- [19] Iñigo Mingo, José Ignacio Rodríguez, y Javier García de Jalón, *Aprenda Java como si estuviera en primero*. Universidad de Navarra, 1999.
- [20] Ana Chévez, «Servidor de aplicaciones JBoss». [En línea]. Disponible en: <http://asteriscus.com/presentacion/presentacion%20A71922.pdf>. [Accedido: 02-dic-2014].
- [21] Michael Yuan y Thomas Heute, *JBoss Seam Simplicity and Power Beyond Java EE*. .
- [22] Cristóbal González Almirón, «Introducción a JSF Java», 2013. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=IntroduccionJSFJava>. [Accedido: 11-dic-2014].
- [23] José Manuel Sánchez Suárez, «Introducción a RichFaces.», 2010. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsfIntro>. [Accedido: 11-dic-2014].
- [24] Juan Alonso Ramos, «Introducción a Ajax4Jsf», 2008. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=Ajax4Jsf>. [Accedido: 24-nov-2014].

- [25] José Manuel Sánchez Suárez, «Migración de JSP a Facelets». [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=migrateJSF2Facelets>. [Accedido: 11-dic-2014].
- [26] Cesar Crespo Martín, «Introducción a Hibernate», 2008. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=hibernate>. [Accedido: 24-nov-2014].
- [27] Luis Ángel Gallego Villa, «Ensayo, Uso de Herramientas CASE». Tecnología en Análisis y Desarrollo SENA. Puerto Berrío, Antioquia., 2011.
- [28] Mike Milinkovich, «Eclipse: The Open Innovation Network», abr-2007. [En línea]. Disponible en: [http://lomboz.ow2.org/eclipsist/2007/presentations/Mike\\_Millinkovich\\_Open\\_Innovation.pdf](http://lomboz.ow2.org/eclipsist/2007/presentations/Mike_Millinkovich_Open_Innovation.pdf). [Accedido: 03-feb-2015].
- [29] Giulio Toffoli, *The Definitive Guide to iReport*. JasperSoft Corporation, 2007.
- [30] Cecilia Belletti y Regina Motz, «Análisis de Herramientas CASE para uso didáctico en Diseño de Bases de Datos», Facultad de Ingeniería, Universidad ORT Uruguay.
- [31] Fredy Marcelo Padrón Ortiz, «PGADMIN III: Administrador de Base de Datos Open Source PostgreSQL», Universidad Académica de Ingeniería de Sistemas, Eléctrica y Electrónica, Cuenca-Ecuador, 2013.
- [32] E. M. Bennatan, «Software Project Management: A Practitioner's Approach». E.U: Editorial McGraw Hill, 1992.
- [33] R. M. Matos, «Introducción al trabajo con Base de Datos». 2004.

- [34] «Diagrama de despliegue». [En línea]. Disponible en: [http://www.ecured.cu/index.php/Diagrama\\_de\\_despliegue](http://www.ecured.cu/index.php/Diagrama_de_despliegue). [Accedido: 01-dic-2014].
- [35] Ciberula, «Patrones de Diseño en aplicaciones Web con Java J2EE». [En línea]. Disponible en: [http://java.ciberaula.com/articulo/disenio\\_patrones\\_j2ee](http://java.ciberaula.com/articulo/disenio_patrones_j2ee). [Accedido: 24-nov-2014].
- [36] Gutiérrez, Jorge A. Saavedra, «El Mundo Informático. Patrones GRASP (Patrones de Software para la asignación General de Responsabilidad).Parte II». [En línea]. Disponible en: <http://jorgesaavedra.wordpress.com/>. [Accedido: 24-nov-2014].
- [37] Liliana González Palacio, «Método para generar Casos de prueba funcional en el desarrollo del software», *Revista Ingenierías Universidad de Medellín*, dic. 2009.

*Bibliografía*

[1] Cecilia Belletti y Regina Motz, «Análisis de Herramientas CASE para uso didáctico en Diseño de Bases de Datos», Facultad de Ingeniería, Universidad ORT Uruguay.

[2] Iñigo Mingo, José Ignacio Rodríguez, y Javier García de Jalón, Aprenda Java como si estuviera en primero. Universidad de Navarra, 1999.

[3] Yeimile López Almejín, «Arquitectura - Ensayos de Calidad». [En línea]. Disponible en: <http://www.clubensayos.com/Tecnolog%C3%ADa/Arquitectura/1809239.html>. [Accedido: 16-dic-2014].

[4] Marc Gilbert Ginesta y Oscar Pérez Mora, «Bases de datos en PostgreSQL». [En línea]. Disponible en: [http://informatica.gonzalonazareno.org/plataforma/file.php/3/Apuntes/UOC\\_Postgres.pdf](http://informatica.gonzalonazareno.org/plataforma/file.php/3/Apuntes/UOC_Postgres.pdf). [Accedido: 03-feb-2015].

[5] Daniel Fernández Lanvin, «Definición de una arquitectura software para el diseño de aplicaciones web basadas en tecnología Java-J2EE», Estudios de Doctorado Avances en Informática, Universidad de Oviedo.

[6] Pedro Cuesta y Morales, «Desarrollo de Aplicaciones Distribuidas basadas en Tecnologías Web». [En línea]. Disponible en: <http://trevinca.ei.uvigo.es/~pcuesta/publicaciones/TecWeb.pdf>. [Accedido: 11-dic-2014].

[7] Jayson Falkner, Ben Galbraith, y Romin Irani, «Desarrollo Web con JSP - Dialnet». [En línea]. Disponible en: <http://dialnet.unirioja.es/servlet/libro?codigo=314753>. [Accedido: 12-ene-2015].

[8] «Diagrama de despliegue». [En línea]. Disponible en: [http://www.ecured.cu/index.php/Diagrama\\_de\\_despliegue](http://www.ecured.cu/index.php/Diagrama_de_despliegue). [Accedido: 01-dic-2014].

[9] G. C. M. Alexander González Domínguez, «Diseño y Desarrollo de una Aplicación Informática para la Elaboración y Evaluación de Menús Hospitalarios», Rev Cub Aliment Nutr.

[10] «Eclipse, entorno de desarrollo integrado - EcuRed». [En línea]. Disponible en: [http://www.ecured.cu/index.php/Eclipse,\\_entorno\\_de\\_desarrollo\\_integrado](http://www.ecured.cu/index.php/Eclipse,_entorno_de_desarrollo_integrado). [Accedido: 11-dic-2014].

[11] Maija Mustonen, «Eclipse RCP and Eclipse Java IDE», mar-2007.

[12] Mike Milinkovich, «Eclipse: The Open Innovation Network», abr-2007. [En línea]. Disponible en: [http://lomboz.ow2.org/eclipsist/2007/presentations/Mike\\_Millinkovich\\_Open\\_Innovation.pdf](http://lomboz.ow2.org/eclipsist/2007/presentations/Mike_Millinkovich_Open_Innovation.pdf). [Accedido: 03-feb-2015].

[13] Bibiana Apolonia del Brutto, «El impacto de las tecnologías de información y comunicación en las sociedades latinoamericanas - Monografias.com». [En línea]. Disponible en: <http://www.monografias.com/trabajos901/impacto-tic-sociedades-latinoamericanas/impacto-tic-sociedades-latinoamericanas.shtml>. [Accedido: 24-nov-2014].

[14] José Enrique González Cornejo, «El Lenguaje de Modelado Unificado (UML)», ene-2008. [En línea]. Disponible en: <http://www.docirs.cl/uml.htm>. [Accedido: 24-nov-2014].

[15] Gutiérrez, Jorge A. Saavedra, «El Mundo Informático. Patrones GRASP (Patrones de Software para la asignación General de Responsabilidad).Parte II». [En línea]. Disponible en: <http://jorgesaavedra.wordpress.com/>. [Accedido: 24-nov-2014].

[16] E. Bascón Pantoja, «El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing», Revista Acta Nova, vol. 2, n. ° 4, 2011.

[17] Ivar Jacobson, Grady Booch, y James Rumbaugh, El Proceso Unificado de Desarrollo de Software. .

[18] Luis Ángel Gallego Villa, «Ensayo, Uso de Herramientas CASE». Tecnología en Análisis y Desarrollo SENA. Puerto Berrío, Antioquia., 2011.

[19] N. K. Carlos Reynoso, «Estilos y Patrones en la Estrategia de Arquitectura de Microsoft», mar-2004. [En línea]. Disponible en: <http://carlosreynoso.com.ar/archivos/arquitectura/Estilos.PDF>. [Accedido: 11-dic-2014].

[20] Bertha Mariel Márquez Avendaño y José Manuel Zulaica Rugarcía, «Implementación de un reconocedor de voz gratuito a el sistema de ayuda a invidentes Dos-Vox en español», Universidad de las Américas Puebla, Cholula, Puebla, México.

[21] M. A. B. Osmany Aguilera Almaguer y Eduardo Peña Cruz, «Integración de la enseñanza de las TIC en la educación», 03-jul-2011. [En línea]. Disponible en: <http://www.eumed.net/rev/ced/29/abc.htm>. [Accedido: 11-dic-2014].

[22] Juan Alonso Ramos, «Introducción a Ajax4Jsf», 2008. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=Ajax4Jsf>. [Accedido: 24-nov-2014].

[23] César Crespo Martín, «Introducción a Hibernate», 2008. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=hibernate>. [Accedido: 24-nov-2014].

[24] Cristóbal González Almirón, «Introducción a JSF Java», 2013. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=IntroduccionJSFJava>. [Accedido: 11-dic-2014].

[25] Alejandro Teruel, «Introducción a la arquitectura de capas». [En línea]. Disponible en: <http://ldc.usb.ve/~teruel/ci3715/clases/arqCapas.html>. [Accedido: 14-ene-2015].

[26] Richard Lou Torrijos, «Introducción a la Tecnología JavaServer Faces. Programación en Castellano.» [En línea]. Disponible en:

[http://programacion.net/articulo/introduccion\\_a\\_la\\_tecnologia\\_javascript\\_faces\\_233](http://programacion.net/articulo/introduccion_a_la_tecnologia_javascript_faces_233).  
[Accedido: 11-dic-2014].

[27] R. M. Matos, «Introducción al trabajo con Base de Datos». 2004.

[28] Hanna Oktaba, «Introducción a Patrones», Facultad de Ciencias, UNAM. [En línea]. Disponible en: <http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/patrones.html>. [Accedido: 11-dic-2014].

[29] José Manuel Sánchez Suárez, «Introducción a RichFaces.», 2010. [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=richFacesJsfIntro>. [Accedido: 11-dic-2014].

[30] «Introduction to JBoss Seam», 22-oct-2014. [En línea]. Disponible en: <http://docs.jboss.org/seam/2.3.0.Final/reference/en-US/html/Book-Preface.html>. [Accedido: 22-oct-2014].

[31] John Zukowski, Java\_2\_J2SE\_1.4. España: Anaya Multimedia (Grupo Anaya, S.A.), 2003.

[32] Michael Yuan y Thomas Heute, JBoss Seam Simplicity and Power Beyond Java EE. .

[33] «Lenguajes de programación, programación Java». [En línea]. Disponible en: <http://www.lenguajes-de-programacion.com/programacion-java.shtml>. [Accedido: 24-nov-2014].

[34] Manuel Mundaca, «Lo que se cocina en las TIC | A un Clic de las TIC». [En línea]. Disponible en: <http://www.aunclidelastic.com/lo-que-se-cocina-en-las-tic/>. [Accedido: 11-dic-2014].

[35] «Marco de Desarrollo de la Junta de Andalucía | MADEJA». [En línea]. Disponible en: <http://www.juntadeandalucia.es/servicios/madeja/>. [Accedido: 11-dic-2014].

[36] Liliana González Palacio, «Método para generar Casos de prueba funcional en el desarrollo del software», Revista Ingenierías Universidad de Medellín, dic. 2009.

[37] José Manuel Sánchez Suárez, «Migración de JSP a Facelets». [En línea]. Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=migrateJSF2Facelets>. [Accedido: 11-dic-2014].

[38] «Modelo RUP | Ing. en Software». [En línea]. Disponible en: <http://softwarerecompilation.wordpress.com/modelo-rup/>. [Accedido: 09-dic-2014].

[39] H. Merlino, «Patrón de Diseño de Vistas Adaptables», Reportes Técnicos en Ingeniería del Software, vol. 8, n. ° 2, pp. 30–35, jul. 2013.

[40] Ciberula, «Patrones de Diseño en aplicaciones Web con Java J2EE». [En línea]. Disponible en: [http://java.ciberaula.com/articulo/disenio\\_patrones\\_j2ee](http://java.ciberaula.com/articulo/disenio_patrones_j2ee). [Accedido: 24-nov-2014].

[41] Fredy Marcelo Padrón Ortiz, «PGADMIN III: Administrador de Base de Datos Open Source PostgreSQL», Universidad Académica de Ingeniería de Sistemas, Eléctrica y Electrónica, Cuenca-Ecuador, 2013.

[42] Yasel Ramos Medina y Dalian Romero Gómez, «Planificación dinámica de actividades asistenciales del Sistema Integral para la Atención Primaria de Salud», Universidad de las Ciencias Informáticas (UCI), La Habana, Mayo de 2012.

[43] Daniel Pecos Martínez, «PostgreSQL vs. MySQL - geekWare». [En línea]. Disponible en: <http://danielpecos.com/documents/postgresql-vs-mysql/>. [Accedido: 24-nov-2014].

[44] «Procesos de Software - Metodologías para el Desarrollo del Software». [En línea]. Disponible en: <http://procesosdesoftware.wikispaces.com/METODOLOGIAS+PARA+DESARROLLO+DE+SOFTWARE>. [Accedido: 01-dic-2014].

- [45] J. J. Gutiérrez, ¿Qué es un framework web? 2006.
- [46] Katty Ávila, « ¿Qué es un Sistema Gestor de Bases de Datos o SGBD?». [En línea]. Disponible en: <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>. [Accedido: 01-dic-2014].
- [47] Colectivo de autores, «Reglamento Orgánico-UCF». .
- [48] J. A. L.-G. Jacqueline Marín-Sánchez y Pedro Yobanis Piñero-Pérez, «Revista Cubana de Ciencias Informáticas - Proceso para la planificación y control de proyectos de software utilizando Xedro-GESPRO». [En línea]. Disponible en: [http://scielo.sld.cu/scielo.php?pid=S2227-18992014000200010&script=sci\\_arttext](http://scielo.sld.cu/scielo.php?pid=S2227-18992014000200010&script=sci_arttext). [Accedido: 01-dic-2014].
- [49] C. R. Juan Palacio, «Scrum Manager: Proyectos – Formación.»
- [50] «Seam | Marco de Desarrollo de la Junta de Andalucía». [En línea]. Disponible en: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/94>. [Accedido: 11-dic-2014].
- [51] Ana Chévez, «Servidor de aplicaciones JBoss». [En línea]. Disponible en: <http://asteriscus.com/presentacion/presentacion%20A71922.pdf>. [Accedido: 02-dic-2014].
- [52] «Servidores Web - Monografias.com». [En línea]. Disponible en: <http://www.monografias.com/trabajos75/servidores-web/servidores-web.shtml>. [Accedido: 01-dic-2014].
- [53] «SIPRO-Solución integral de control de producción.pdf». [En línea]. Disponible en: <https://drive.google.com/viewerng/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbW Fpbnxzb2x1Y2lvbmVzcGFyYWNhdGVyaW5nfGd4OjczY2JmZVhNWZiYzBIZGE>. [Accedido: 24-nov-2014].

[54] Arasay Rodríguez Bastida, «Sistema informático para la gestión de la información referente a las ventas del Departamento Comercial de la Sucursal Caracol Sancti Spíritus.», Universidad de Cienfuegos, Cienfuegos, Cuba.

[55] Raquel Zambrano Ramírez, «Sistemas Gestores de Base de Datos», Revista Digital «Innovación y Experiencias», n.º Nro. 14-Enero de 2008.

[56] Sara Álvarez, «Sistemas gestores de bases de datos». [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>. [Accedido: 01-dic-2014].

[57] E. M. Bennatan, «Software Project Management: A Practitioner's Approach». E.U: Editorial McGraw Hill, 1992.

[58] Next Computer Services, S. A., «Solución Integral de Control de la Producción». [En línea]. Disponible en: <https://drive.google.com/viewerng/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbW Fpbnxzb2x1Y2lvbmVzcGFyYWNhdGVyaW5nfGd4OjczY2JjMzVhNWZiYzBIZGE>. [Accedido: 24-nov-2014].

[59] F. J. Díaz, C. A. Queiruga, y L. Fava, «Struts y JavaServer Faces, cara a cara», 2007.

[60] Teknisa Software, «Tecfood Servicios de alimentación». [En línea]. Disponible en: <http://www.teknisa.com/es/productos/tecfood-servicios-de-alimentaci%C3%B3n>. [Accedido: 24-nov-2014].

[61] Félix Rodríguez Álvarez, «Tecnologías de la información y la comunicación», 15-jul-2013. [En línea]. Disponible en: <http://www.monografias.com/tecnologia-informacion-comunicacion>. [Accedido: 15-jul-2013].

[62] Giulio Toffoli, The Definitive Guide to iReport. JasperSoft Corporation, 2007.

[63] Bruce Eckel, Thinking\_In\_Java\_4th\_Edition. January, 2006.

## *Glosario de términos*

**Muestras testigo:** Las muestras testigo son porciones de cada plato de un menú que se conserva temporalmente para ser examinado ante cualquier eventualidad ocurrida con los comensales que digirieron el menú.

**IPV:** Inventario a Precio de Venta. Nombre que recibe el documento que refleja todo el flujo de entrada y salida de los productos, con sus respectivas cantidades.

**Crowdsourcing:** es un tipo de actividad en línea participativa en la que una persona, institución, organización sin ánimo de lucro o empresa, propone a un grupo de individuos mediante una convocatoria abierta flexible, la realización libre y voluntaria de una tarea.

**UCF:** Universidad de Cienfuegos.

**SGBD:** Sistema Gestor de Base de Datos.

**JEE:** Plataforma Java Enterprise Edition.

**IDE:** Entorno de Desarrollo Integrado.

**JSF :** Java Server Face.

**POJOs :** Plain Old Java Objects.

**EJB :** Enterprise Java Beans.

**JPA :** Java Persistence Api

Anexos

**Anexo A: Descripción de los casos de uso del negocio.**

Anexo A 1 Descripción textual Recibir productos.

<b>Nombre del caso de uso del negocio:</b>	Recibir productos	
<b>Actores del negocio:</b>	Almacenero del comedor de la UCF.	
<b>Propósito:</b>	Abastecer al almacenero de la Residencia de Postgrado con los productos necesarios.	
<b>Resumen:</b>		
<p>El caso de uso inicia cuando el almacenero del comedor de la universidad hace entrega de los productos al almacenero de la residencia o cuando este último los solicite. La entrega de los productos se efectúa a través de un vale. El caso de uso finaliza cuando el almacenero de la residencia les da entrada a los productos en el almacén.</p>		
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
1- El almacenero del comedor de la universidad entrega los productos y un vale al almacenero de la residencia.	1.1- El almacenero recibe los productos y el vale.	

<p>2- El almacenero recibe el vale firmado.</p>	<p>1.2- El almacenero verifica mercancía contra vale.</p> <p>1.3- El almacenero firma el vale, guarda la copia y entrega el original al almacenero del comedor.</p> <p>2.1- El almacenero actualiza información de la existencia de los productos en el almacén, dándole entrada a los que acaban de llegar.</p>
<p><b>Cursos alternos:</b></p> <p>En la actividad 1, pudiese ser el almacenero de la residencia quien iniciase el proceso solicitando los productos.</p> <p>En la actividad 1.2 si el almacenero al verificar encuentra algún error se lo comunicará al almacenero del comedor.</p>	
<p><b>Prioridad:</b></p>	<p>Crítica.</p>
<p><b>Mejoras:</b></p>	<p>Optimizar el tiempo que demora el almacenero de la residencia en dar entrada a los productos, además de limitar la posibilidad de errores de cálculo.</p>

Anexo A 2 Descripción textual Planificar menú.

<b>Nombre del caso de uso del negocio:</b>	Planificar menú
<b>Actores del negocio:</b>	Cliente.
<b>Propósito:</b>	Mostrar el menú al cliente y determinar la cantidad de productos necesarios para extraer del almacén según los menús confeccionado.
<b>Resumen:</b>	
<p>El caso de uso inicia cuando el cliente solicita el menú. El planificador tomando como guía el menú entregado por la directora, planifica el menú, verificando la existencia en el almacén. El caso de uso finaliza cuando se determina la cantidad necesaria de productos a extraer a través del IPV y se muestra el menú a los clientes.</p>	
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1- El cliente solicita el menú del día.	<p>1.1- La carpetera elabora el informe de los huéspedes y entrega al planificador.</p> <p>1.2- El planificador solicita la propuesta del menú elaborada por la directora.</p>

<p>2- El cliente recibe el menú y finaliza el caso de uso.</p>	<p>1.3- La directora entrega la propuesta del menú.</p> <p>1.4- El planificador recibe el menú y determina la cantidad a extraer del almacén para confeccionarlo.</p> <p>1.5- El planificador verifica la cantidad contra la existencia en el almacén.</p> <p>1.6- El planificador confirma el menú y emite el IPV y las hojas de cálculo que indican las normas.</p> <p>1.7- El planificador muestra el menú a los clientes.</p>
<p><b>Cursos alternos:</b></p> <p>En la actividad 1.4 si el planificador al verificar cantidad contra existencia en el almacén, nota que no se puede hacer toda la extracción, deberá plantearlo a la directora y esta determinará qué hacer.</p>	
<p><b>Prioridad:</b></p>	<p>Crítica.</p>
<p><b>Mejoras:</b></p>	<p>Optimizar el tiempo que demora el planificador en calcular la cantidad de productos necesarios para confeccionar el menú, así como la elaboración del IPV y las hojas de cálculo.</p>



Anexo A 3 Descripción textual Realizar pedido.

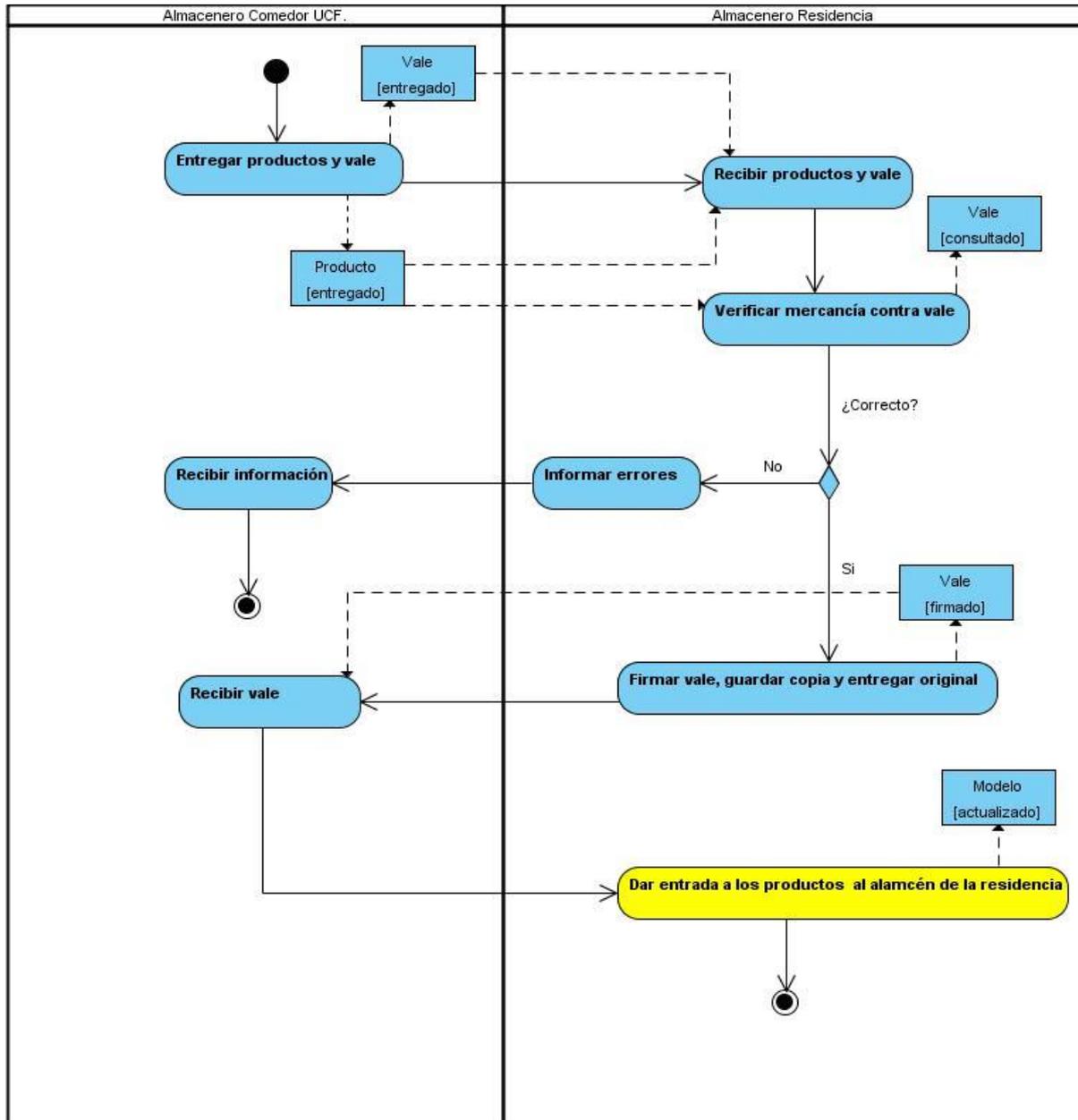
<b>Nombre del caso de uso del negocio:</b>	Realizar pedido	
<b>Actores del negocio:</b>	Planificador	
<b>Propósito:</b>	Extraer los productos del almacén para elaborar el menú y darle baja a los mismos.	
<b>Resumen:</b>		
El caso de uso inicia cuando el planificador entrega el IPV al almacenero de la residencia. El almacenero entrega los productos. El caso de uso finaliza cuando el almacenero les da baja del almacén a los productos entregados.		
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del negocio</b>	
1- El planificador entrega el IPV confeccionado al almacenero.	1.1- El almacenero toma el pedido y comienza a hacer entrega de los productos.	
2- El planificador recibe los productos.	2.1- El almacenero les da baja a los productos del almacén.	

---

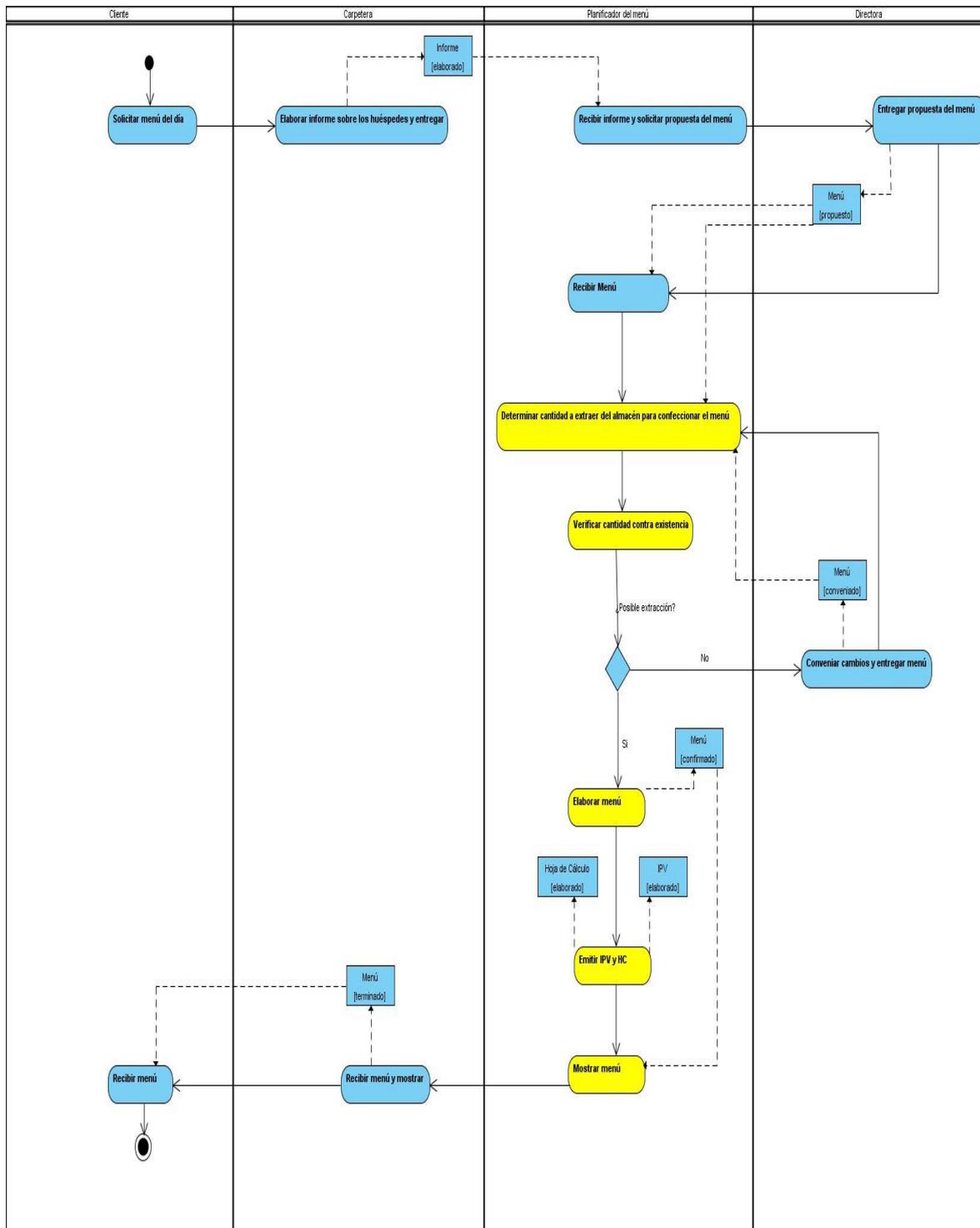
<b>Cursos alternos:</b>  En la actividad 1.1 si cuando el almacenero toma el pedido nota problemas lo comenta al planificador.	
<b>Prioridad:</b>	Crítica.
<b>Mejoras:</b>	Optimizar el tiempo que demora el almacenero en dar baja a los productos, así como limitar la posibilidad de errores de cálculo.

Anexo B: Diagramas de actividades del negocio.

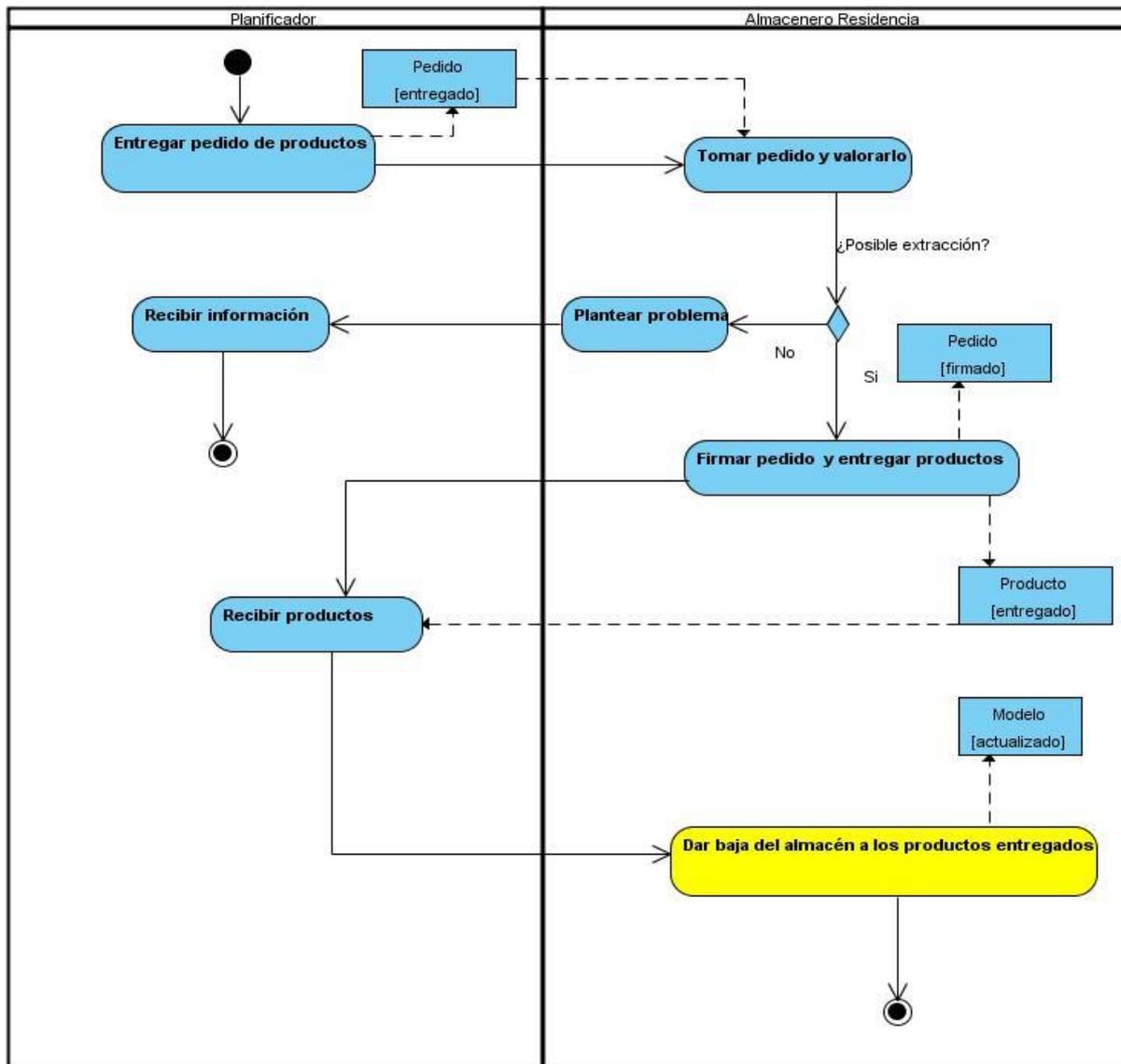
Anexo B 1 Recibir productos



Anexo B 2 Planificar Menú



Anexo B 3 Recibir productos



**Anexo C: Descripción de los casos de uso del sistema.**

Anexo C 1 Descripción textual Mostrar existencia.

<b>Caso de Uso</b>	Mostrar existencia.
<b>Actores</b>	Almacenero.
<b>Resumen</b>	El caso de uso inicia cuando el almacenero necesita conocer la existencia de los productos en el almacén; para lo cual presiona el botón Almacén y luego Real en Almacén en la interfaz principal. El caso de uso finaliza cuando el sistema muestra los productos existentes.
<b>Responsabilidades</b>	Brindar información al almacenero acerca de la existencia de los productos en el almacén.
<b>CU asociados</b>	No hay.
<b>Precondiciones</b>	Inicialmente el almacenero deberá autenticarse con éxito en el sistema, para luego poder conocer la existencia de los productos en el almacén.
<b>Descripción</b>	
<b>Acción del Actor (es)</b> 1-El almacenero presiona el botón Almacén en la interfaz principal.  2- El almacenero elige la opción Real en Almacén.	<b>Respuesta del Sistema</b> 1.1-El sistema despliega un menú.  2.1- El sistema muestra una lista con todos los productos existentes en el almacén, donde el almacenero si desea puede

	exportar el reporte obtenido a formato pdf o Excel.
<b>Prototipo de Interfaz</b>	
<b>Sección</b> No existe	
<b>Requisitos funcionales</b>	R1,R2,R3
<b>Postcondiciones</b>	Queda visualizado para el almacenero los productos existentes en el almacén.

Anexo C 2 Descripción textual Entrada de productos.

<b>Caso de Uso</b>	Entrada de productos.
<b>Actores</b>	Almacenero.
<b>Resumen</b>	El caso de uso inicia cuando el almacenero necesita dar entrada a los productos que llegan al almacén. El caso de uso finaliza cuando el sistema registra en la base de datos los productos.
<b>Responsabilidades</b>	Permitir dar entrada al almacén de los productos que llegan.
<b>CU asociados</b>	No hay.
<b>Precondiciones</b>	Inicialmente el almacenero deberá autenticarse con éxito en el sistema.

Descripción	
<p><b>Acción del Actor (es)</b></p> <ol style="list-style-type: none"> <li>1- El almacenero presiona el botón Almacén en la interfaz principal.</li> <li>2- El almacenero elige la opción Recepción de Mercancía.</li> <li>3- El almacenero introduce los datos y presiona la opción Aceptar.</li> </ol>	<p><b>Respuesta del Sistema</b></p> <ol style="list-style-type: none"> <li>1.1-El sistema despliega un menú.</li> <li>2.1- El sistema muestra una interfaz que contiene un formulario para especificar el tipo de entrada que se efectuará y los productos a entrar.</li> <li>3.1- El sistema valida los datos, los registra en la base de datos y muestra un mensaje de confirmación, finalizando el caso de uso.</li> </ol>

**Prototipo de Interfaz**

The screenshot displays the 'Entrada de productos' interface. At the top, there's a title bar. Below it, a form titled 'Tipo de entrada:' contains three radio buttons: 'Vale' (selected), 'Interno', and 'Expediente'. Below the form, there are input fields for 'Fecha:' (03-03-2015), 'Tipo:' (Numerado), and 'Número:' (5). A 'Buscar' button is located below the form. Below the search bar is a list of products with columns for 'Productos', 'Moneda', and 'Cantidad'. The list includes items like 'Frijoles Colorados', 'Garbanzo', 'Masa cerdo', 'Frijoles Colorados', 'Mayonesa', 'Pan', 'Pollo Troceado', 'Natilla', 'Mezcla polvo batido', 'Pulpa Guayaba', 'Calabaza', 'Huevo', and 'Cerdo Banda'. To the right of the list is a 'Cantidad:' input field with the value '5' and navigation buttons '<' and '>'. Below the list is a 'Nuevo Producto' button. To the right of the list is a table with columns 'Nombre', 'Moneda', and 'Cantidad'. The table contains two rows: 'Galletas Dulces' with 'MN' and '3.0', and 'Pasta de cebolla condimentada' with 'MN' and '5.0'.

**Sección No existe**

**Cursos Alternos**

En la línea 3.1 si al sistema validar los datos introducidos por el almacenero encuentra errores o campos vacíos, le hace un señalamiento en los campos

<p>haciéndoselo saber. El almacenero vuelve a introducir los datos y continúa con la acción 3.</p>	
<b>Requisitos funcionales</b>	R4, R5.
<b>Postcondiciones</b>	Queda registrado en la Base de Datos, los nuevos productos que entraron al almacén.

Anexo C 3 Descripción textual Salida de productos.

<b>Caso de Uso</b>	Salida de Productos.
<b>Actores</b>	Almacenero.
<b>Resumen</b>	El caso de uso inicia cuando el almacenero le va a dar salida a los productos para la confección de los menús. El caso de uso finaliza cuando el sistema muestra un mensaje de confirmación y da la opción de exportar a formato pdf o Excel la salida de los productos.
<b>Responsabilidades</b>	Dar salida a los productos del almacén por menú, por fecha o según el tipo de movimiento.
<b>CU asociados</b>	
<b>Precondiciones</b>	Inicialmente el almacenero deberá autenticarse con éxito en el sistema.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>

<p>1 El almacenero presiona el botón Almacén en la interfaz principal.</p> <p>2 El almacenero necesita dar salida a los productos.</p> <p>3 El almacenero elige la opción para exportar a formato pdf o Excel los reportes obtenidos.</p>	<p>1.1- El sistema despliega un menú.</p> <p>2.1-El sistema ejecuta alguna de las siguientes acciones:</p> <p>a- Si el almacenero decide dar salida a los productos por fecha ir a la sección “Fecha”.</p> <p>b- Si el almacenero decide dar salida a los productos por menú ir a la sección “Menú”.</p> <p>c- Si el almacenero decide dar salida a los productos por movimiento ir a la sección “Movimiento”.</p> <p>3.1- El sistema muestra un archivo pdf o Excel con los datos obtenidos y finaliza el caso de uso.</p>
<p><b>Sección “Menú”</b></p>	

## Prototipo de Interfaz

### Seleccione los parámetros

Fecha:  Tipo:  Nivel:

### Vista previa

#### Salida de productos por menú

Fecha: 06/03/2015

Tipo: Desayuno

Nivel: Nivel 1

Código	Producto	Moneda	UM	Cantidad
P003	Azúcar	MN	g	2.00
P048	Mortadella	MN	g	2.00
P019	Pan	MN	U	2.00
P079	Aqua Natural	MN	U	2.00
P034	Guayaba	MN	g	2.00
P001	Sal	MN	g	2.00

### Acción del actor

- 1- El almacenero presiona la opción Por menú.
  
- 2- El almacenero elige los datos y presiona el botón Aceptar.

### Respuesta del sistema

- 1.1- El sistema le muestra una interfaz con un formulario para elegir los datos del menú.
  
- 2.1- El sistema muestra un reporte con los productos a sacar para la confección del menú seleccionado.

### Cursos Alternos

En la línea 2.1 si el sistema no encuentra un menú con los datos elegidos muestra un mensaje.

### Sección "Fecha"

<b>Prototipo de Interfaz</b>																																				
 <p>The screenshot shows a web interface with a header 'Seleccione los parámetros' containing a date input field set to 'Mar 3, 2015' and an 'Aceptar' button. Below is a 'Reporte' section with the title 'Salida de productos diaria' and a table of products for the date '03/03/2015'.</p> <table border="1" data-bbox="427 645 917 824"> <thead> <tr> <th>Código</th> <th>Producto</th> <th>Moneda</th> <th>UM</th> <th>Cantidad</th> </tr> </thead> <tbody> <tr> <td>P002</td> <td>Arroz</td> <td>MN</td> <td>q</td> <td>15,00</td> </tr> <tr> <td>P025</td> <td>Calabaza</td> <td>MN</td> <td>q</td> <td>8,00</td> </tr> <tr> <td>P001</td> <td>Sal</td> <td>MN</td> <td>q</td> <td>33,00</td> </tr> <tr> <td>P005</td> <td>Aceite</td> <td>MN</td> <td>q</td> <td>21,00</td> </tr> <tr> <td>P079</td> <td>Aqua Natural</td> <td>MN</td> <td>U</td> <td>17,00</td> </tr> <tr> <td>P116</td> <td>Yuca</td> <td>CUC</td> <td>q</td> <td>8,00</td> </tr> </tbody> </table>		Código	Producto	Moneda	UM	Cantidad	P002	Arroz	MN	q	15,00	P025	Calabaza	MN	q	8,00	P001	Sal	MN	q	33,00	P005	Aceite	MN	q	21,00	P079	Aqua Natural	MN	U	17,00	P116	Yuca	CUC	q	8,00
Código	Producto	Moneda	UM	Cantidad																																
P002	Arroz	MN	q	15,00																																
P025	Calabaza	MN	q	8,00																																
P001	Sal	MN	q	33,00																																
P005	Aceite	MN	q	21,00																																
P079	Aqua Natural	MN	U	17,00																																
P116	Yuca	CUC	q	8,00																																
<p><b>Acción del actor</b></p> <p>1- El almacenero selecciona la opción Por fecha.</p> <p>2- El almacenero elige la fecha y presiona el botón Aceptar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1- El sistema le muestra una interfaz con un formulario para introducir una fecha.</p> <p>2.1- El sistema muestra un reporte con los productos a sacar para la confección de los menús en la fecha especificada.</p>																																			
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si el sistema no encuentra menús en la fecha elegida muestra un mensaje.</p>																																				
<p><b>Sección "Movimiento"</b></p>																																				

## Prototipo de Interfaz

Salida de productos por movimiento

Tipo de movimiento:

Fuera de Norma   
  Interno   
  Centro de Costo   
  Merma

Justificación

Productos	Moneda	Cantidad
Sal	MN	4806.25
oooooooooooooooo	MN	0.0
oooooooooooooooo	MN	0.0
Mermelada de Mango	CUC	52.0
Papa Prefrita	CUC	6.0
Jugo Concentrado Naranja	CUC	55.0
eeeeeeee	MN	0.0
oooooooooooooooo	MN	0.0
Caldo sustancioso huesos cerdo - sub...	MN	3.0
Arroz	MN	2349.0
eeeeeeee	MN	0.0
oooooooooooooooo	CUC	0.0

Cantidad

< >

Nombre	Moneda	Cantidad

### Acción del actor

- 1- El almacenero selecciona la opción Movimiento.
- 2- El almacenero introduce los datos del movimiento y presiona el botón aceptar.

### Respuesta del sistema

- 1.1-El sistema le muestra una interfaz con un formulario para introducir que tipo de movimiento desea hacer y los productos a mover.
- 2.1- El sistema valida los datos y crea un movimiento, mostrando un mensaje de confirmación.

### Cursos Alternos

En la línea 2.1 si al sistema validar los datos introducidos por el almacenero encuentra errores o campos vacíos, le hace un señalamiento en los campos

<p>haciéndoselo saber. El almacenero vuelve a introducir los datos y continúa con la acción 2.</p>	
<p><b>Requisitos funcionales</b></p>	<p>R6, R7, R8, R9, R10, R11, R12.</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos con un nuevo movimiento.</p>

Anexo C 4 Descripción textual Determinar costos.

<p><b>Caso de Uso</b></p>	<p>Determinar costos.</p>
<p><b>Actores</b></p>	<p>Planificador</p>
<p><b>Resumen</b></p>	<p>El caso de uso inicia cuando el planificador necesita determinar los costos de los menús confeccionados. El caso de uso finaliza cuando el planificador en caso de ser necesario exporta a formato pdf o Excel los reportes obtenidos.</p>
<p><b>Responsabilidades</b></p>	<p>Determinar costos por menú, por fecha o por período.</p>
<p><b>CU asociados</b></p>	
<p><b>Precondiciones</b></p>	<p>Inicialmente el Planificador deberá autenticarse con éxito en el sistema.</p>
<p><b>Descripción del flujo normal de eventos</b></p>	
<p><b>Acción del Actor (es)</b></p>	<p><b>Respuesta del Sistema</b></p>

<p>1- El planificador presiona el botón Menú en la interfaz principal.</p> <p>2- El planificador necesita determinar los costos.</p> <p>3- El planificador elige la opción para exportar a formato pdf o Excel los reportes obtenidos.</p>	<p>1.2- El sistema despliega un menú.</p> <p>2.1-El sistema ejecuta alguna de las siguientes acciones:</p> <p>a- Si el planificador decide determinar los costos por menú, ir a la sección “Menú”.</p> <p>b- Si el planificador decide determinar los costos diarios, ir a la sección “Diario”.</p> <p>c- Si el planificador decide determinar los costos por período, ir a la sección “Período”.</p> <p>d- Si el planificador decide determinar los costos por moneda, ir a la sección “Moneda”.</p> <p>3.1- El sistema muestra un archivo pdf o Excel con los datos obtenidos y finaliza el caso de uso.</p>
<p><b>Sección “Menú”</b></p>	

### Prototipo de Interfaz

**Seleccione los parámetros**

Fecha:  Tipo:  Nivel:

---

**Vista previa**

## Costo del Menú

Fecha: 03/03/2015  
 Tipo: Almuerzo  
 Nivel: Nivel 3

Código	Producto	Moneda	UM	Cantidad	Costo	SubTotal
P001	Sal	MN	g	4,00	0,20	0,80
P002	Arroz	MN	g	4,00	0,30	1,20
P005	Aceite	MN	g	4,00	0,60	2,40

#### Acción del actor

- 1- El planificador presiona la opción Por menú.
  
- 2- El planificador elige los datos y presiona el botón Aceptar.

#### Respuesta del sistema

- 1.1- El sistema le muestra una interfaz con un formulario para elegir los datos del menú.
  
- 2.1- El sistema muestra un reporte con los costos del menú seleccionado.

#### Cursos Alternos

En la línea 2.1 si el sistema no encuentra un menú con los datos elegidos muestra un mensaje.

#### Sección "Diario"

#### Prototipo de Interfaz

**Seleccione los parámetros**

Fecha:

**Reporte**

### Costo diario del menú

Fecha: 06/03/2015

Código	Producto	Moneda	UM	Cantidad	Costo	SubTotal
P011	Muslo y contramuslo Pollo	CUC	g	2,00	0,20	0,40
P007	Harina trigo	MN	g	2,00	0,30	0,60
P245	Papa frita	MN	g	2,00	0,60	1,20
P028	Huevo	MN	U	2,00	0,20	0,40
P001	Sal	MN	g	10,00	0,20	2,00
P034	Guayaba	MN	g	2,00	0,20	0,40
P002	Arroz	MN	g	4,00	0,30	1,20
P003	Azúcar	MN	g	2,00	0,50	1,00
P079	Agua Natural	MN	U	4,00	0,20	0,80
P019	Pan	MN	U	2,00	0,20	0,40
P005	Aceite	MN	g	6,00	0,60	3,60
P048	Mortadella	MN	g	2,00	0,50	1,00

**Acción del actor**

- 1- El planificador selecciona la opción Diario.
  
- 2- El planificador elige la fecha y presiona el botón Aceptar.

**Respuesta del sistema**

- 1.1- El sistema le muestra una interfaz con un formulario para introducir una fecha.
  
- 2.1- El sistema muestra un reporte con los costos de los menús encontrados en la fecha elegida

**Cursos Alternos**

En la línea 2.1 si el sistema no encuentra menús en la fecha elegida muestra un mensaje.

**Sección "Período"**

## Prototipo de Interfaz

**Seleccione los parámetros**

Fecha 1:   Fecha 2:

---

**Vista previa**

### Costo del Menú por periodo

Del 06/03/2015 al 08/05/2015

Código	Producto	Moneda	UM	Cantidad	Costo	SubTotal
P307	Mortadella Novel	MN	g	3,00	0,30	0,90
P245	Papa frita	MN	g	2,00	0,60	1,20
P026	Huevo	MN	U	2,00	0,20	0,40
P039	Papa	MN	g	3,00	0,20	0,60
P274	Salsa criolla	MN	g	1,00	0,20	0,20
P003	Azúcar	MN	g	17,00	0,50	8,50
P079	Agua Natural	MN	U	18,00	0,20	3,60
P027	Cerdo Banda	MN	g	1,00	0,30	0,30
P019	Pan	MN	U	27,00	0,20	5,40

### Acción del actor

- 1- El planificador selecciona la opción Período.
  
- 2- El planificador selecciona las fechas y presiona el botón Aceptar.

### Respuesta del sistema

- 1.2- El sistema le muestra una interfaz con un formulario para introducir una un período de fecha.
  
- 2.1- El sistema muestra un reporte con los costos de los menús encontrados en el período de fecha elegido.

### Cursos Alternos

En la línea 2.1 si el sistema no encuentra un menú con los datos elegidos muestra un mensaje.

### Sección "Moneda"

### Prototipo de Interfaz

**Seleccione los parámetros**

Fecha:   Moneda:

---

**Reporte**

### Costo del Menú por moneda

Fecha: 06/03/2015    Moneda: CUC

Código	Producto	UM	Cantidad	Costo	SubTotal
P011	Muslo y contramuslo Pollo	g	2,00	0,20	0,40

**Descripción del flujo normal de eventos**

<b>Acción del actor</b>	<b>Respuesta del sistema</b>
<p>1- El planificador selecciona la opción Por Moneda.</p> <p>2- El planificador selecciona la fecha, la moneda y presiona el botón Aceptar.</p>	<p>1.1 El sistema le muestra una interfaz con un formulario para elegir el tipo de moneda y la fecha.</p> <p>2.1- El sistema muestra un reporte con los costos de los menús en la moneda seleccionada.</p>
<b>Requisitos funcionales</b>	R13, R14, R15, R16, R17, R18, R19, R20, R21, R22, R23, R24
<b>Postcondiciones</b>	El sistema muestra reportes tanto en formato pdf como en Excel de los costos de los menús seleccionados.

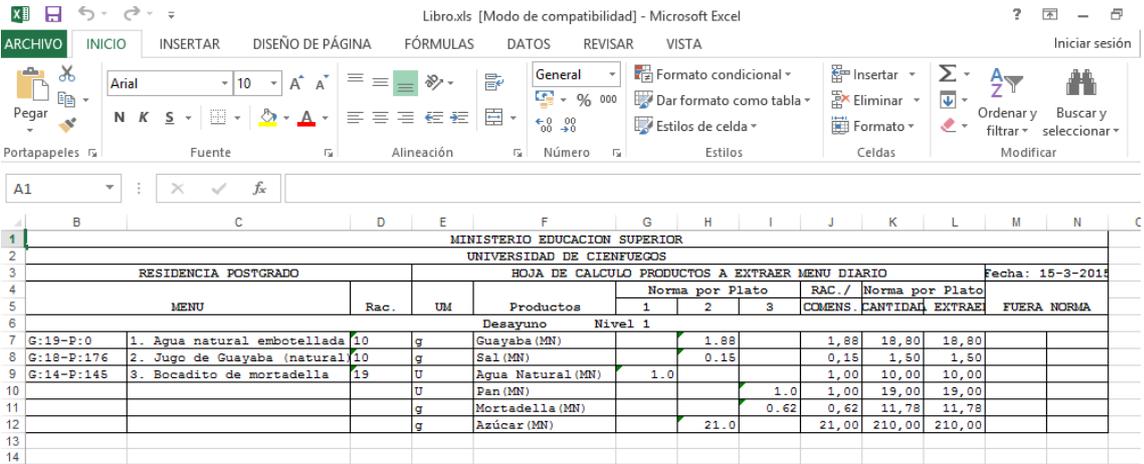
Anexo C 5 Descripción textual Mostrar IPV.

<b>Caso de Uso</b>	Mostrar IPV	
<b>Actores</b>	Planificador	
<b>Resumen</b>	El caso de uso inicia cuando el planificador necesita conocer todo el flujo de información de la fecha en que se trabaja. El caso de uso finaliza cuando el sistema muestra un reporte con todos los datos.	
<b>Responsabilidades</b>	Brindar información al planificador de todas las operaciones de entrada y salida de productos, la cantidad real en existencia y otros datos.	
<b>CU asociados</b>	No hay	
<b>Precondiciones</b>	Inicialmente el planificador deberá autenticarse con éxito en el sistema.	
<b>Descripción</b>		
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>	
1-El planificador presiona el botón Almacén en la interfaz principal.	1.1-El sistema despliega un menú.	
2- El planificador elige la opción IPV.	2.1- El sistema muestra un documento Excel con toda la información que requiere un IPV.	
<b>Prototipo de Interfaz</b>		

MINISTERIO EDUCACION SUPERIOR										
UNIVERSIDAD DE CIENFUEGOS										
RESIDENCIA POSTGRADO		CONTROL INVENTARIO A PRODUCTOS NORMADOS					Fecha			Hoja No.
Área: Despensa							D	M	A	
UM	Productos	Inicio	Entrada	No. Vale Sal. Alm.	ev centr costo	Disp	Salida			Final
							19	4	14	
g	Aceite (MN)	26082,06				26082,06				26082,06
U	Agua Natural (CUC)	65,00				65,00				65,00
g	Aguja salada (MN)	4803,33				4803,33				4803,33
g	Arroz (MN)	34834,42				34834,42				34834,42
<b>Requisitos funcionales</b>			R25							
<b>Postcondiciones</b>			Queda visualizado para el planificador toda la información de las operaciones efectuadas en el día.							

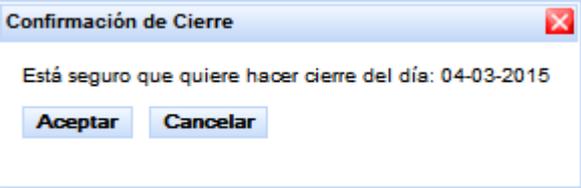
Anexo C 6 Descripción textual Generar HC.

Caso de Uso	Generar HC
<b>Actores</b>	Planificador
<b>Resumen</b>	El caso de uso inicia cuando el planificador necesita crear hojas de cálculo en Excel que recogen la información de las normas por cliente en los platos confeccionados. El caso de uso finaliza cuando el sistema muestra un reporte con todos los datos.
<b>Responsabilidades</b>	Crear hojas de cálculo Excel que muestran las normas por cliente en los platos que conforman el menú del día.
<b>CU asociados</b>	No hay

<p><b>Precondiciones</b></p>	<p>Inicialmente el planificador deberá autenticarse con éxito en el sistema.</p>
<p><b>Descripción</b></p>	
<p><b>Acción del Actor (es)</b></p> <p>1-El planificador presiona el botón Almacén en la interfaz principal.</p> <p>2- El planificador puede elegir el tipo de hoja de cálculo que desea, para cada nivel, para trabajador o muestras testigos.</p>	<p><b>Respuesta del Sistema</b></p> <p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema muestra un documento Excel con toda la información que requiere la opción seleccionada.</p>
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Requisitos funcionales</b></p>	<p>R26-R31</p>
<p><b>Postcondiciones</b></p>	<p>Queda visualizado para el planificador toda la información de las normas para cada cliente.</p>

Anexo C 7 Descripción textual Cierre de Fecha.

Caso de Uso	Cierre de Fecha
<b>Actores</b>	Administrador
<b>Resumen</b>	El caso de uso inicia cuando el administrador una vez que ha terminado de trabajar con una fecha procede a cerrarla. El caso de uso finaliza cuando el sistema cierra la fecha y pasa toda la información al historial.
<b>Responsabilidades</b>	Cerrar una fecha de trabajo.
<b>Precondiciones</b>	Debe existir una fecha iniciada.
<b>Descripción</b>	
<p><b>Acción del Actor (es)</b></p> <p>1- El administrador presiona el botón Sistema en la interfaz principal.</p> <p>2- El administrador elige la opción Cierre.</p> <p>3- El administrador presiona el botón Aceptar.</p>	<p><b>Respuesta del Sistema</b></p> <p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema muestra un mensaje de advertencia.</p> <p>3.1- El sistema cierra la fecha actual de trabajo y actualiza la base datos.</p> <p>3.2- El sistema muestra un mensaje de confirmación.</p>

<b>Prototipo de Interfaz</b> 	
<b>Cursos Alternos</b> En la línea 3 si el administrador decide cancelar la operación el sistema sigue trabajando con la última fecha abierta.	
<b>Requisitos funcionales</b>	R32
<b>Postcondiciones</b>	Queda cerrada la fecha con que se estaba trabajando en el sistema.

Anexo C 8 Descripción textual Inicio de Fecha.

Caso de Uso	Inicio de Fecha
<b>Actores</b>	Administrador
<b>Resumen</b>	El caso de uso inicia cuando el administrador cierra una fecha de trabajo y para comenzar otra necesita abrir una nueva. Esto sucede ya que el administrador no puede trabajar con menús ya confeccionados, de fechas anteriores. El caso de uso finaliza cuando el sistema abre una nueva fecha de trabajo.
<b>Responsabilidades</b>	Abrir una fecha de trabajo.

<p><b>Precondiciones</b></p>	<p>El sistema no puede tener una fecha abierta, o sea, el administrador debe cerrar una fecha para iniciar otra.</p>
<p><b>Descripción</b></p>	
<p><b>Acción del Actor (es)</b></p> <p>1-El administrador presiona el botón Sistema en la interfaz principal.</p> <p>2- El administrador elige la opción Inicio.</p>	<p><b>Respuesta del Sistema</b></p> <p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema verifica que no exista una fecha abierta y abre una nueva.</p>
<p><b>Prototipo de Interfaz</b></p> <div style="display: flex; justify-content: space-around;"> <div data-bbox="204 1104 746 1368"> </div> <div data-bbox="810 1104 1305 1368"> </div> </div>	
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si el sistema encuentra una fecha abierta, lo comunica al administrador a través de un mensaje. El administrador cierra la fecha abierta y continúa con la acción 2.</p>	
<p><b>Requisitos funcionales</b></p>	<p>R33</p>
<p><b>Postcondiciones</b></p>	<p>Queda abierta una nueva fecha de trabajo en el sistema.</p>

Anexo C 9 Descripción textual Gestionar Menú

Caso de Uso	Gestionar Menú.
<b>Actores</b>	Planificador.
<b>Resumen</b>	El caso de uso inicia cuando el planificador del menú necesita elaborar un nuevo menú, conocer los platos de un menú ya elaborado o dado una fecha los tipos de menú que han sido confeccionados. El caso de uso finaliza cuando el sistema muestra los datos solicitados.
<b>Responsabilidades</b>	Permitir al planificador del menú elaborar los menús; así como conocer información de aquellos que han sido elaborados previamente.
<b>Precondiciones</b>	En caso de ser necesario deben existir menús elaborados previamente.
<b>Descripción</b>	
<p><b>Acción del Actor (es)</b></p> <p>1-El planificador presiona el botón Menú en la interfaz principal.</p> <p>2- El planificador necesita elaborar, mostrar o cancelar un menú.</p>	<p><b>Respuesta del Sistema</b></p> <p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema ejecuta alguna de las siguientes acciones:</p> <p>a- Si el planificador decide elaborar un nuevo menú, ir a la sección “Elaborar”.</p>

	<p>b- Si el planificador decide mostrar un menú, ir a la sección “Mostrar”.</p> <p>c- Si el planificador decide mostrar los menús confeccionados en el día, ir a la sección “Confeccionados”.</p> <p>d- Si el planificador decide cancelar un menú, ir a la sección “Cancelar”.</p>
--	---

**Sección “Elaborar”**

**Prototipo de Interfaz**

The screenshot shows a web application interface titled "Elaborar Menú". At the top, there are input fields for "Fecha: 2015-03-04", "Tipo: Almuerzo", and "Nivel: CUC". Below this, there is a section for "Orden de los platos:" with three radio buttons: "Grupo", "Oferta", and "Alfabético". A "Grupo:" dropdown menu is set to "--Seleccione--". A search bar with a "Buscar" button is positioned above a list of menu items. The list includes items like "Agua natural embotellada", "Ajíaco", "Alas de Pollo Empanadas", "Calabaza Hervida", "Escalope de Filete Empanado", "Papas Fritas", "Parguito Frito", "Refresco de sirope", "Alones de Pavo", "Aporreado de cerdo con pan", "Aporreado de pescado con pan", and "Aporreado de Tasajo". To the right of the list is a "Cantidad:" input field and navigation arrows. A table on the right shows the selected items: "Jugo de Mandarina" with a quantity of 2, and "Alas de Pollo Fritas" with a quantity of 4. At the bottom left, there is a "Mostrar existencia" button, and at the bottom right, there are "Cancelar" and "Siguiente" buttons.

**Selección de Productos**

Fecha: 04/03/2015 Nivel: CUC Tipo: Almuerzo

Platos del Menú		
Plato	Cantidad	Listo
Jugo de Mandarina	2	Si
Alas de Pollo Fritas	4	Si

Ingredientes	
Ingrediente	Listo
Naranja agria	Si
Harina de trigo	Si
Ajo en cabeza	Si
Alas de pollo	Si
Grasa	Si
Sal	Si
Pimienta molida	Si

Productos	
Producto	Cantidad
Pasta de Ajo MN	<input checked="" type="checkbox"/>
Ajo MN	<input type="checkbox"/>

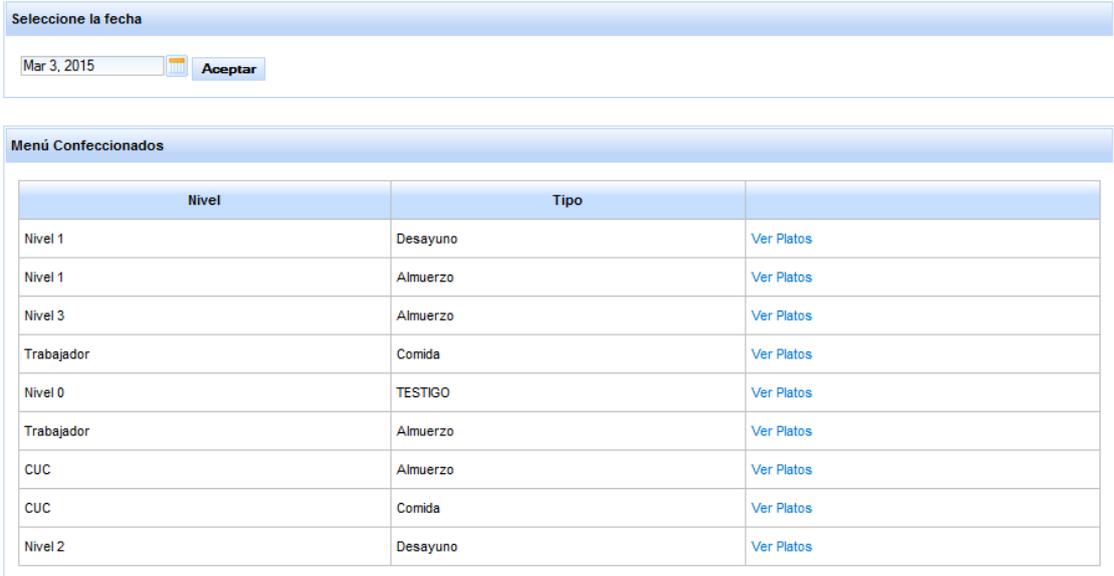
**Acción del actor**

- 1- El planificador presiona la opción para elaborar un nuevo menú.
  
- 2- El planificador introduce los datos y presiona el botón siguiente.
  
- 3- El planificador introduce las cantidades para cada producto en cada ingrediente y presiona el botón aceptar.

**Respuesta del sistema**

- 1.2- El sistema le muestra una interfaz con un formulario para introducir los datos del menú.
  
- 2.1- El sistema verifica que los datos estén correctos.
  
- 2.2- El sistema muestra una nueva interfaz para seleccionar por cada plato los ingredientes y productos necesarios para elaborarlo.
  
- 3.1 El sistema valida que las cantidades se correspondan con la cantidad de platos e inserta en la base de datos los datos del nuevo menú.

	<p>3.2 El sistema muestra un mensaje de éxito.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores, le hace un señalamiento en los campos donde fueron detectados los errores o la falta de datos. El planificador vuelve a introducir los datos y continúa con la acción 2.</p> <p>En la línea 4.1 si al sistema verificar las cantidades no coinciden con la cantidad del plato, el sistema hace un señalamiento en el campo del error. El planificador introduce una nueva cantidad y continúa con la acción 4.</p>	
<p><b>Sección “Mostrar”</b></p>	
<p><b>Prototipo de Interfaz</b></p>  <p>The screenshot shows a web browser window titled 'Vista previa'. The main content is a menu titled 'Menú' with the date 'Fecha: 03/03/2015'. There are two sections: one for 'Desayuno' (Nivel: Nivel 1) with an offer of 'Agua natural embotellada', and one for 'Almuerzo' (Nivel: Nivel 1) with an offer of 'Calabaza Hervida'. A vertical scrollbar is visible on the right side of the content area.</p>	
<p><b>Acción del actor</b></p> <p>1- El planificador elige la opción mostrar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1- El sistema muestra los menús confeccionados en el día con sus respectivos platos.</p>

<b>Cursos Alternos</b>	
En la línea 1.2 si el sistema no encuentra menús en el día, muestra un mensaje.	
<b>Sección “Confeccionados”</b>	
<b>Prototipo de Interfaz</b>	
	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
<p>1- El planificador elige la opción Confeccionar.</p> <p>2- El planificador introduce la fecha.</p>	<p>1.1- El sistema muestra un formulario para introducir la fecha.</p> <p>2.1- El sistema muestra un reporte con todos los menús elaborados en la fecha y sus respectivos platos.</p>
<b>Sección “Cancelar”</b>	
<b>Prototipo de Interfaz</b>	

Eliminar Menú		
Menús planificados:		
Tipo	Nivel	Eliminar
Desayuno	Nivel 1	
Almuerzo	Nivel 1	
Almuerzo	Nivel 3	
Comida	Trabajador	
TESTIGO	Nivel 0	
Almuerzo	Trabajador	
Almuerzo	CUC	
Comida	CUC	
Desayuno	Nivel 2	
Descripción del flujo normal de eventos		
Acción del actor	Respuesta del sistema	
<p>1- El planificador elige la opción Cancelar.</p> <p>2- El planificador elige el menú a cancelar.</p>	<p>1.2- El sistema muestra los menús confeccionados en el día.</p> <p>2.1- El sistema cancela en la base de datos el menú seleccionado y lo elimina de la lista mostrada.</p>	
Requisitos funcionales	R34-R37.	
Postcondiciones	Queda actualizada la base de datos del sistema con un nuevo menú, eliminado en caso de ver elegido esta opción o mostrados.	

Anexo C 10 Descripción textual Gestionar Plato

<b>Caso de Uso</b>	Gestionar Plato.
<b>Actores</b>	Planificador.
<b>Resumen</b>	El caso de uso se inicia cuando el planificador necesita insertar, eliminar o modificar un plato. El caso de uso culmina con la actualización de los datos.
<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un plato.
<b>Precondiciones</b>	
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
1-El planificador presiona el botón Gestionar en la interfaz principal.	1.1-El sistema despliega un menú.
2- El planificador elige la opción Platos.	2.1- El sistema muestra una lista de todos los platos.
3- El planificador necesita insertar, modificar, buscar o eliminar un plato.	3.1- El sistema ejecuta alguna de las siguientes acciones:  a- Si el planificador decide insertar un nuevo plato, ir a la sección "Insertar".

<p>4- El planificador termina el trabajo con los platos.</p>	<p>b- Si el planificador decide eliminar un plato, ir a la sección “Eliminar”.</p> <p>c- Si el planificador decide buscar un plato, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
--	---

**Sección “Insertar”**

**Prototipo de Interfaz**

**Crear nuevo Plato**

Nombre:  Grupo:

**Asentamiento:**

Manual  Carta

**A servir:**

U.M.

**Buscar**

Ingredientes	Norma Huésped	Norma Trabajador
Ajo	Neto: <input type="text"/>	Neto: <input type="text"/>
Azúcar	Bruto: <input type="text"/>	Bruto: <input type="text"/>
Apio		
Azúcar refinado		
Bacón		
Bicarbonato de sodio		
Bistec de hígado de res		
Boniato		
Calabaza		
Caldo de ave		
Caldo sustancioso		
Camarones (cola)		
Carne de Res 1ra	Obligatorio: <input type="checkbox"/>	
Catchup ó puré de tomate		
Cebolla		
Chicharrones (CE)		
Concentrado de mango		
Corazón		

Nombre	Neto H.	Bruto H.	Neto T.	Bruto T.	Oblig.

<p><b>Acción del actor</b></p>	<p><b>Respuesta del sistema</b></p>
--------------------------------	-------------------------------------

<p>1 El planificador presiona la opción para insertar un nuevo plato.</p> <p>2 El planificador introduce los datos y presiona el botón insertar.</p>	<p>1.1 El sistema le muestra una interfaz con un formulario para introducir los datos del plato.</p> <p>2.1 El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema inserta el nuevo plato en la base de datos y muestra un mensaje de éxito.</p>
--	---

**Cursos Alternos**

En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción 2.

En la línea 2.2 si al sistema tratar de insertar el plato, este ya existe, se lo comunica a través de un mensaje al planificador y finaliza el caso de uso.

**Sección “Buscar”**

**Prototipo de Interfaz**

**Buscar Plato**

Nombre:

**Listado de Platos**

Nombre	Peso ración H	Peso ración T	Um	Página manual	Manual	Consecutivo Trab	Número carta	Grupo	Acción
Agua natural embotellada	500	0	ml	0		0	0	19	<a href="#">Ver</a>
Ajiaco	464	0	g	27		0	0	01	<a href="#">Ver</a>
Alas de Pollo Empanadas	200	0	g	64		0	0	05	<a href="#">Ver</a>
Alas de Pollo Fritas	174	0	g	64		0	0	05	<a href="#">Ver</a>
Albóndiga de ternera a la catalana	348	0	g	70		0	0	06	<a href="#">Ver</a>
Aliño para Ensaladas	1000	0	g	170		0	0	17	<a href="#">Ver</a>
Alones de Pavo	464	0	g	65		0	0	05	<a href="#">Ver</a>
Aporreado de cerdo con pan	120	0	g	148		0	0	14	<a href="#">Ver</a>
Aporreado de pescado con pan	90	0	g	152		0	0	14	<a href="#">Ver</a>
Aporreado de Tasajo	261	75	g	71		0	0	06	<a href="#">Ver</a>

[Siguiente >](#) [Última >>](#)

**Acción del actor**

- 1 El planificador introduce el dato del plato en el cuadro de texto que aparece en la interfaz y presiona el botón buscar.

**Respuesta del sistema**

- 1.1 El sistema selecciona en la lista mostrada de todos los platos, el que concuerda con el criterio introducido en el cuadro de texto.

**Cursos Alternos**

En la línea 1.1 si el sistema no encuentra el plato, muestra un mensaje informando la no existencia del plato.

**Sección “Eliminar”**

**Acción del actor**

- 1 El planificador selecciona el plato y presiona la opción Editar.

**Respuesta del sistema**

- 1.1 El sistema muestra un formulario con los datos del plato y la opción Eliminar.

<p>2 El planificador presiona la opción Eliminar.</p>	<p>2.1 El sistema elimina el plato de la basa de datos y muestra al planificador la lista de platos actualizada.</p>
<p><b>Cursos Alternos</b> En la línea 2.1 si el sistema no puede eliminar el plato muestra un mensaje al planificador.</p>	
<p><b>Requisitos funcionales</b></p>	<p>R43-R46.</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos del sistema con un nuevo plato, guardados los cambios efectuados sobre alguno ya existente o eliminado en caso de ver elegido esta opción.</p>

Anexo C 11 Descripción textual Gestionar Producto.

<p><b>Caso de Uso</b></p>	<p>Gestionar Producto.</p>
<p><b>Actores</b></p>	<p>Almacenero.</p>
<p><b>Resumen</b></p>	<p>El caso de uso inicia cuando el almacenero presiona el botón Gestionar en la interfaz principal y luego de desplegarse un menú, selecciona la opción Productos. El caso de uso finaliza cuando el almacenero haya insertado, modificado o eliminado el producto deseado.</p>

<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un producto.
<b>CU asociados</b>	
<b>Precondiciones</b>	Inicialmente el almacenero deberá autenticarse con éxito en el sistema, para luego poder insertar, modificar, eliminar o buscar un producto.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
<p>1-El almacenero presiona el botón Gestionar en la interfaz principal.</p> <p>2- El almacenero elige la opción productos.</p> <p>3- El almacenero necesita insertar, modificar, buscar o eliminar un producto.</p>	<p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema muestra una lista de todos los productos.</p> <p>3.1- El sistema ejecuta alguna de las siguientes acciones:</p> <p>a- Si el almacenero decide insertar un nuevo producto, ir a la sección "Insertar".</p> <p>b- Si el almacenero decide modificar un producto, ir a la sección "Modificar".</p> <p>c- Si el almacenero decide eliminar un producto, ir a la sección "Eliminar".</p>

<p>4- El almacenero termina el trabajo con los productos.</p>	<p>d- Si el almacenero decide buscar un producto, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
---	---

**Prototipo de Interfaz**



**Sección “Insertar”**

**Prototipo de Interfaz**



<p><b>Acción del actor</b></p>	<p><b>Respuesta del sistema</b></p>
--------------------------------	-------------------------------------

<p>1- El almacenero presiona la opción para insertar un nuevo producto.</p> <p>2- El almacenero introduce los datos y presiona el botón insertar.</p>	<p>1.3- El sistema le muestra una interfaz con un formulario para introducir los datos del producto.</p> <p>2.1- El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema inserta el nuevo producto en la base de datos y muestra un mensaje de éxito.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si al sistema validar los datos introducidos por el almacenero encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El almacenero vuelve a introducir los datos y continúa con la acción2.</p> <p>En la línea 2.2 si al sistema tratar de insertar el producto, este ya existe, se lo comunica a través de un mensaje al almacenero y finaliza el caso de uso.</p>	
<p><b>Sección “Buscar”</b></p>	
<p><b>Prototipo de Interfaz</b></p>	

**Buscar Producto**

Código:

Nombre:

U medida:

Moneda:

Tipo peso:

**Listado de Productos**

Código	Nombre	U medida	Moneda	Tipo peso	Peso unidad	Costo	Acción
P001	Sal	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P002	Arroz	g	MN	N	1	0.3	<a href="#">Ver Editar</a>
P003	Azúcar	g	MN	N	1	0.5	<a href="#">Ver Editar</a>
P004	Frijoles blancos	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P005	Aceite	g	MN	N	1	0.6	<a href="#">Ver Editar</a>
P006	Pastas largas	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P007	Harina trigo	g	MN	N	1	0.3	<a href="#">Ver Editar</a>
P008	Condimento	g	MN	N	1	0.5	<a href="#">Ver Editar</a>
P009	Frijoles negros	g	MN	N	1	0.2	<a href="#">Ver Editar</a>
P010	Col	g	MN	B	1	0.6	<a href="#">Ver Editar</a>

**Acción del actor**

1- El almacenero introduce el dato del producto en el cuadro de texto que aparece en la interfaz y presiona el botón buscar.

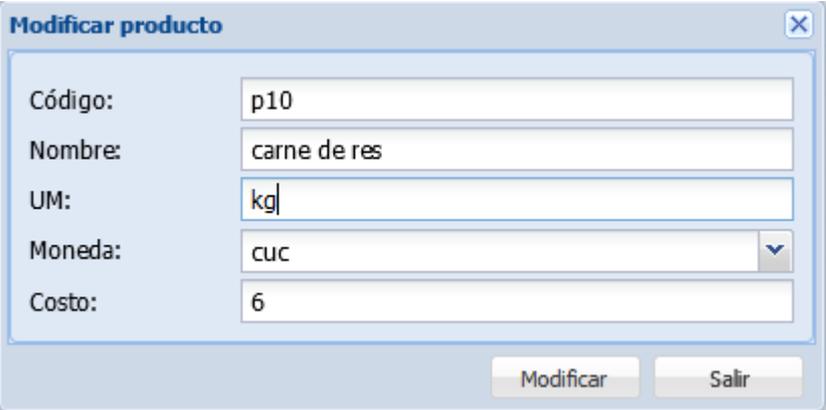
**Respuesta del sistema**

1.1 El sistema selecciona en la lista mostrada de todos los productos, el que concuerda con el criterio introducido en el cuadro de texto.

**Cursos Alternos**

En la línea 1.1 si el sistema no encuentra el producto, muestra un mensaje informando la no existencia del producto.

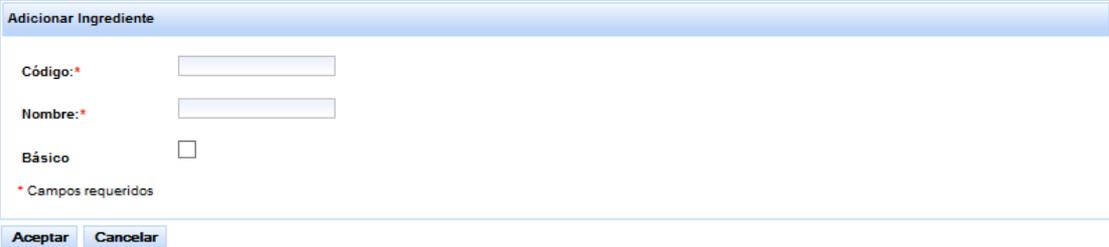
**Sección “Eliminar”**

<p><b>Acción del actor</b></p> <p>1 El almacenero selecciona el producto y presiona la opción Editar.</p> <p>2 El almacenero presiona la opción Eliminar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema muestra un formulario con los datos del producto y la opción Eliminar.</p> <p>2.1 El sistema elimina el producto de la basa de datos y muestra al almacenero la lista de productos actualizada.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 1.1 si el sistema no puede eliminar el producto muestra un mensaje al almacenero.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Descripción del flujo normal de eventos</b></p>	
<p>1 El almacenero selecciona la opción editar.</p>	<p>1.1- El sistema muestra una interfaz con los datos del producto.</p> <p>2.1- El sistema valida que los datos estén correctos y no hayan campos vacíos.</p>

<p>2 El almacenero modifica los datos y presiona la opción Aceptar.</p>	<p>2.2- El sistema actualiza la información existente en la base de datos con los cambios efectuados.</p> <p>2.3- El sistema muestra la interfaz que contiene los datos del producto actualizados.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el almacenero encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El almacenero vuelve a introducir los datos y continúa con la acción 2.</p> <p>En la actividad 2.2 si el sistema no puede actualizar los datos porque el producto ya existe le muestra un mensaje al almacenero.</p>
<p><b>Requisitos funcionales</b></p>	<p>R38-R42.</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos del sistema con un nuevo producto, guardados los cambios efectuados sobre alguno ya existente o eliminado en caso de ver elegido esta opción.</p>

Anexo C 12 Descripción textual Gestionar Ingrediente.

<b>Caso de Uso</b>	Gestionar Ingrediente.
<b>Actores</b>	Planificador.
<b>Resumen</b>	El caso de uso se inicia cuando el planificador necesita insertar, buscar, eliminar o modificar un ingrediente. El caso de uso culmina con la actualización de los datos.
<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un ingrediente.
<b>CU asociados</b>	
<b>Precondiciones</b>	
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b> 1-El planificador presiona el botón Gestionar en la interfaz principal.  2- El planificador elige la opción Ingredientes.  3- El planificador necesita insertar, modificar, buscar o eliminar un ingrediente.	<b>Respuesta del Sistema</b> 1.1-El sistema despliega un menú.  2.1- El sistema muestra una lista de todos los ingredientes.  3.1- El sistema ejecuta alguna de las siguientes acciones:  a- Si el planificador decide insertar un nuevo ingrediente, ir a la sección "Insertar".

<p>4- El planificador termina el trabajo con los ingredientes.</p>	<p>b- Si el planificador decide modificar un ingrediente, ir a la sección “Modificar”.</p> <p>c- Si el planificador decide eliminar un ingrediente, ir a la sección “Eliminar”.</p> <p>d- Si el planificador decide buscar un ingrediente, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
<p><b>Sección “Insertar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Acción del actor</b></p> <p>1 El planificador presiona la opción para insertar un nuevo ingrediente.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema le muestra una interfaz con un formulario para introducir los datos del ingrediente.</p>

<p>2 El planificador introduce los datos y presiona el botón insertar.</p>	<p>2.1 El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema inserta el nuevo ingrediente en la base de datos y muestra una interfaz con los datos del ingrediente.</p>
--	---

### Cursos Alternos

En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción 2.

En la línea 2.2 si al sistema tratar de insertar el ingrediente, este ya existe, se lo comunica a través de un mensaje al planificador y finaliza el caso de uso.

### Sección “Buscar”

#### Prototipo de Interfaz

**Buscar Ingrediente** «

Código:

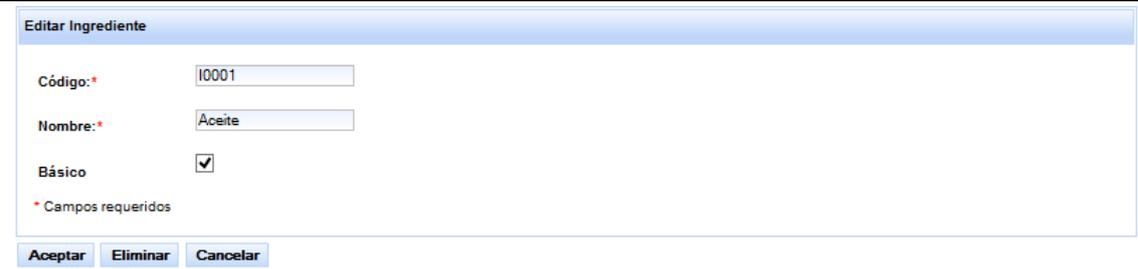
Nombre:

**Listado de Ingredientes**

Código	Nombre	Básico	Acción
I0001	Aceite	Si	<a href="#">Ver Editar</a>
I0002	Acido cítrico	No	<a href="#">Ver Editar</a>
I0003	Ají	No	<a href="#">Ver Editar</a>
I0004	Ajo	No	<a href="#">Ver Editar</a>
I0005	Ajo en cabeza	No	<a href="#">Ver Editar</a>
I0006	Ajo puerro	No	<a href="#">Ver Editar</a>
I0007	Ajo puerro o cebollino	No	<a href="#">Ver Editar</a>
I0008	Alas de pollo	No	<a href="#">Ver Editar</a>
I0010	Alones de pavo	No	<a href="#">Ver Editar</a>
I0011	Anón	No	<a href="#">Ver Editar</a>

[Siguiete >](#) [Última >>](#)

<p><b>Acción del actor</b></p> <p>1 El planificador introduce el dato del ingrediente en el cuadro de texto que aparece en la interfaz y presiona el botón buscar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema selecciona en la lista mostrada de todos los ingredientes, el que concuerda con el criterio introducido en el cuadro de texto.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 1.1 si el sistema no encuentra el ingrediente, muestra un mensaje informando la no existencia del ingrediente.</p>	
<p><b>Sección “Eliminar”</b></p>	
<p><b>Acción del actor</b></p> <p>1 El planificador selecciona el ingrediente y presiona la opción Editar.</p> <p>2 El planificador presiona la opción Eliminar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema muestra un formulario con los datos del ingrediente y la opción Eliminar.</p> <p>2.1 El sistema elimina el ingrediente de la basa de datos y muestra al planificador la lista de ingredientes actualizada.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si el sistema no puede eliminar el ingrediente muestra un mensaje al administrador.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Prototipo de Interfaz</b></p>	

	
<p><b>Descripción del flujo normal de eventos</b></p>	
<p>1 El planificador selecciona la opción editar.</p> <p>2 El planificador modifica los datos y presiona la opción Aceptar.</p>	<p>1.1 El sistema muestra una interfaz con los datos del ingrediente.</p> <p>2.1 El sistema valida que los datos estén correctos y no hayan campos vacíos.</p> <p>2.2 El sistema actualiza la información existente en la base de datos con los cambios efectuados.</p> <p>2.3 El sistema muestra la interfaz que contiene los datos del ingrediente actualizados.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción 2.</p> <p>En la actividad 2.2 si el sistema no puede actualizar los datos porque el ingrediente</p>

	ya existe le muestra un mensaje al planificador.
<b>Requisitos funcionales</b>	R47-R51.
<b>Postcondiciones</b>	Queda actualizada la base de datos del sistema con un nuevo ingrediente, guardados los cambios efectuados sobre alguno ya existente o eliminado en caso de ver elegido esta opción.

Anexo C 13 Descripción textual Asociar productos a un ingrediente.

<b>Caso de Uso</b>	Asociar productos a un ingrediente
<b>Actores</b>	Planificador
<b>Resumen</b>	El caso de uso inicia cuando el planificador necesita asociar productos existentes en el almacén a los ingredientes correspondientes. El caso de uso finaliza cuando el sistema muestra un mensaje de confirmación.
<b>Responsabilidades</b>	Asociar Productos a Ingredientes.
<b>Precondiciones</b>	El planificador debe estar autenticado en el sistema.
<b>Descripción</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
1- El planificador presiona el botón Menú en la interfaz principal.	1.1-El sistema despliega un menú.

<p>2- El planificador elige la opción Asociar Productos a Ingredientes.</p> <p>3- El planificador asocia los productos según su ingrediente y presiona el botón aceptar.</p>	<p>2.1- El sistema muestra un formulario.</p> <p>3.1- El sistema muestra un mensaje de confirmación.</p>
--	--

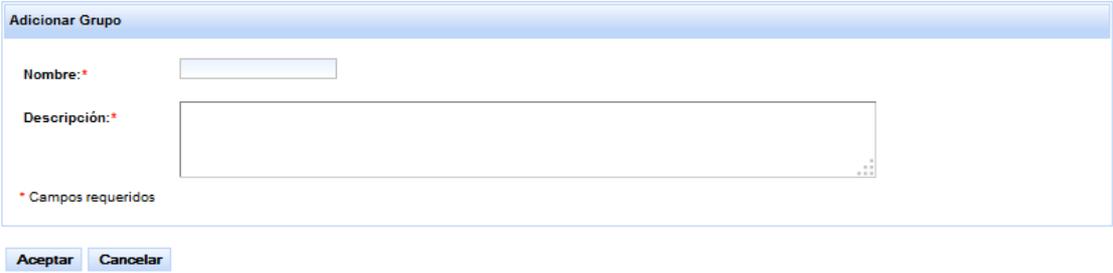
**Prototipo de Interfaz**



<p><b>Requisitos funcionales</b></p>	<p>R52</p>
<p><b>Postcondiciones</b></p>	<p>Queda almacenada en la base de datos una nueva asociación de productos por ingrediente.</p>

Anexo C 14 Descripción textual Gestionar grupo.

<b>Caso de Uso</b>	Gestionar Grupo.
<b>Actores</b>	Planificador.
<b>Resumen</b>	El caso de uso se inicia cuando el planificador necesita insertar, buscar, eliminar o modificar un grupo. El caso de uso culmina con la actualización de los datos.
<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un grupo.
<b>CU asociados</b>	
<b>Precondiciones</b>	
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
1-El planificador presiona el botón Gestionar en la interfaz principal.	1.1-El sistema despliega un menú.
2- El planificador elige la opción Grupos.	2.1- El sistema muestra una lista de todos los grupos.
3- El planificador necesita insertar, modificar, buscar o eliminar un grupo.	3.1- El sistema ejecuta alguna de las siguientes acciones:  a- Si el planificador decide insertar un nuevo grupo, ir a la sección "Insertar".

<p>4- El planificador termina el trabajo con los grupos.</p>	<p>b- Si el planificador decide modificar un grupo, ir a la sección “Modificar”.</p> <p>c- Si el planificador decide eliminar un grupo, ir a la sección “Eliminar”.</p> <p>d- Si el planificador decide buscar un grupo, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
<p><b>Sección “Insertar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Acción del actor</b></p> <ol style="list-style-type: none"> <li>1 El planificador presiona la opción para insertar un nuevo grupo.</li> <li>2 El planificador introduce los datos y presiona el botón insertar.</li> </ol>	<p><b>Respuesta del sistema</b></p> <ol style="list-style-type: none"> <li>1.2 El sistema le muestra una interfaz con un formulario para introducir los datos del grupo.</li> <li>2.1 El sistema verifica que los datos estén correctos.</li> </ol>

2.2- El sistema inserta el nuevo grupo en la base de datos y muestra una interfaz con los datos del grupo.

### Cursos Alternos

En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción2.

En la línea 2.2 si al sistema tratar de insertar el grupo, este ya existe, se lo comunica a través de un mensaje al planificador y finaliza el caso de uso.

### Sección “Buscar”

#### Prototipo de Interfaz

**Buscar Grupo**

Nombre:

Descripción:

**Buscar** **Cancelar**

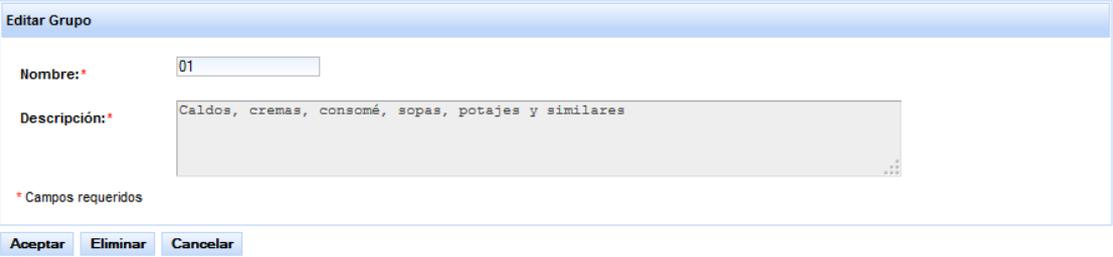
**Listado de Grupos**

Nombre	Descripción	Acción
01	Caldos, cremas, consomé, sopas, potajes y similares	<a href="#">Ver</a> <a href="#">Editar</a>
02	Huevos	<a href="#">Ver</a> <a href="#">Editar</a>
03	Pescados	<a href="#">Ver</a> <a href="#">Editar</a>
04	Mariscos	<a href="#">Ver</a> <a href="#">Editar</a>
05	Aves	<a href="#">Ver</a> <a href="#">Editar</a>
06	Carne de res	<a href="#">Ver</a> <a href="#">Editar</a>
07	Carne de cerdo, Jamón y embutidos	<a href="#">Ver</a> <a href="#">Editar</a>
08	Caprino y ovino	<a href="#">Ver</a> <a href="#">Editar</a>
09	Conejo	<a href="#">Ver</a> <a href="#">Editar</a>
10	Platos fríos	<a href="#">Ver</a> <a href="#">Editar</a>

[Nuevo Grupo](#) [Siguiete >](#) [Última >>](#)

### Acción del actor

### Respuesta del sistema

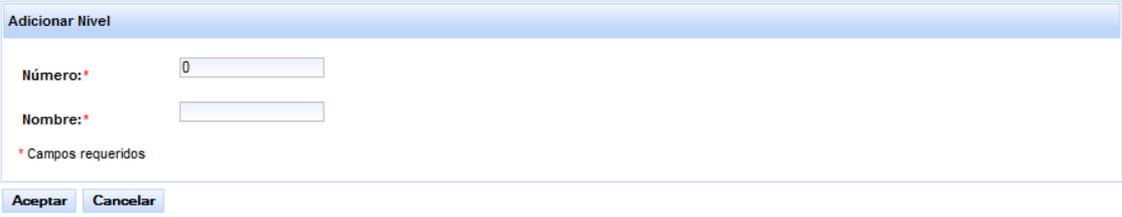
<p>2 El planificador introduce el dato del grupo en el cuadro de texto que aparece en la interfaz y presiona el botón buscar.</p>	<p>2.1 El sistema selecciona en la lista mostrada de todos los grupos, el que concuerda con el criterio introducido en el cuadro de texto.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 1.1 si el sistema no encuentra el grupo, muestra un mensaje informando la no existencia del grupo.</p>	
<p><b>Sección “Eliminar”</b></p>	
<p><b>Acción del actor</b></p> <p>2 El planificador selecciona el grupo y presiona la opción Editar.</p> <p>2 El planificador presiona la opción Eliminar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema muestra un formulario con los datos del grupo y la opción Eliminar.</p> <p>2.1 El sistema elimina el grupo de la base de datos y muestra al planificador la lista de grupos actualizada.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si el sistema no puede eliminar el grupo muestra un mensaje al administrador.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Descripción del flujo normal de eventos</b></p>	

<p>1 El planificador selecciona la opción editar.</p> <p>2 El planificador modifica los datos y presiona la opción Aceptar.</p>	<p>1.2 El sistema muestra una interfaz con los datos del grupo.</p> <p>2.1 El sistema valida que los datos estén correctos y no hayan campos vacíos.</p> <p>2.2 El sistema actualiza la información existente en la base de datos con los cambios efectuados.</p> <p>2.3 El sistema muestra la interfaz que contiene los datos del grupo actualizados.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción 2.</p> <p>En la actividad 2.2 si el sistema no puede actualizar los datos porque el grupo ya existe le muestra un mensaje al planificador.</p>
<p><b>Requisitos funcionales</b></p>	<p>R53-R57.</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos del sistema con un nuevo grupo, guardados los cambios efectuados sobre alguno ya</p>

	existente o eliminado en caso de ver elegido esta opción.
--	---

Anexo C 15 Descripción textual Gestionar Nivel.

<b>Caso de Uso</b>	Gestionar Nivel.
<b>Actores</b>	Planificador.
<b>Resumen</b>	El caso de uso se inicia cuando el planificador necesita insertar, eliminar o modificar un nivel. El caso de uso culmina con la actualización de los datos.
<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un nivel.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
1-El planificador presiona el botón Gestionar en la interfaz principal.	1.1-El sistema despliega un menú.
2- El planificador elige la opción Niveles.	2.1- El sistema muestra una lista de todos los niveles.
3- El planificador necesita insertar, modificar, buscar o eliminar un nivel.	3.1- El sistema ejecuta alguna de las siguientes acciones:  a- Si el planificador decide insertar un nuevo nivel, ir a la sección "Insertar".

<p>4- El planificador termina el trabajo con los niveles.</p>	<p>b- Si el planificador decide modificar un nivel, ir a la sección “Modificar”.</p> <p>c- Si el planificador decide eliminar un nivel, ir a la sección “Eliminar”.</p> <p>d- Si el planificador decide buscar un nivel, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
<p><b>Prototipo de Interfaz</b></p>	
<p><b>Sección “Insertar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Acción del actor</b></p> <p>1 El planificador presiona la opción para insertar un nuevo nivel.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema le muestra una interfaz con un formulario para introducir los datos del nivel.</p>

<p>2 El planificador introduce los datos y presiona el botón insertar.</p>	<p>2.1 El sistema verifica que los datos estén correctos.</p> <p>2.2- El sistema inserta el nuevo nivel en la base de datos y muestra un mensaje de éxito.</p>
--	--

**Cursos Alternos**

En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción2.

En la línea 2.2 si al sistema tratar de insertar el nivel, este ya existe, se lo comunica a través de un mensaje al planificador y finaliza el caso de uso.

**Sección “Buscar”**

**Prototipo de Interfaz**

Buscar Nivel

Nombre:

Buscar Cancelar

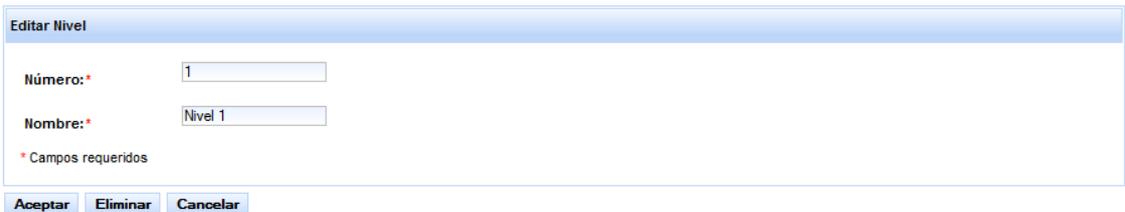
Listado de Niveles

Número	Nombre	Acción
1	Nivel 1	Ver Editar
2	Nivel 2	Ver Editar
3	Nivel 3	Ver Editar
4	CUC	Ver Editar
5	Trabajador	Ver Editar
0	Nivel 0	Ver Editar

Nuevo Nivel

<p><b>Acción del actor</b></p> <p>1 El planificador introduce el dato del nivel en el cuadro de</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema selecciona en la lista mostrada de todos los niveles,</p>
---	---

<p>texto que aparece en la interfaz y presiona el botón buscar.</p>	<p>el que concuerda con el criterio introducido en el cuadro de texto.</p>
<p><b>Cursos Alternos</b> En la línea 1.1 si el sistema no encuentra el nivel, muestra un mensaje informando la no existencia del nivel.</p>	
<p><b>Sección “Eliminar”</b></p>	
<p><b>Acción del actor</b></p> <ol style="list-style-type: none"> <li>1 El planificador selecciona el nivel y presiona la opción Editar.</li> <li>2 El planificador presiona la opción Eliminar.</li> </ol>	<p><b>Respuesta del sistema</b></p> <ol style="list-style-type: none"> <li>1.1 El sistema muestra un formulario con los datos del nivel y la opción Eliminar.</li> <li>2.1 El sistema elimina el nivel de la base de datos y muestra al planificador la lista de niveles actualizada.</li> </ol>
<p><b>Cursos Alternos</b> En la línea 2.1 si el sistema no puede eliminar el nivel muestra un mensaje al planificador.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Prototipo de Interfaz</b></p>	

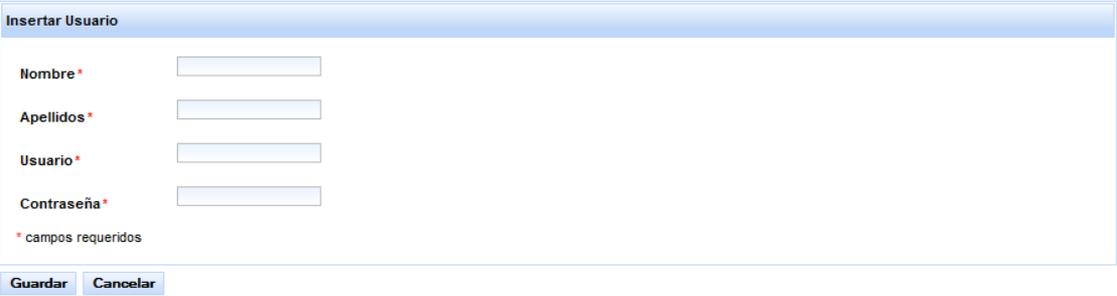
	
<p><b>Descripción del flujo normal de eventos</b></p>	
<p>1 El planificador selecciona la opción editar.</p> <p>2 El planificador modifica los datos y presiona la opción Aceptar.</p>	<p>1.1 El sistema muestra una interfaz con los datos del nivel.</p> <p>2.1 El sistema valida que los datos estén correctos y no hayan campos vacíos.</p> <p>2.2 El sistema actualiza la información existente en la base de datos con los cambios efectuados.</p> <p>2.3 El sistema muestra la interfaz que contiene los datos del nivel actualizados.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el planificador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El planificador vuelve a introducir los datos y continúa con la acción2.</p> <p>En la actividad 2.2 si el sistema no puede actualizar los datos porque el nivel ya existe le muestra un mensaje al planificador.</p>

---

<b>Requisitos funcionales</b>	R58 - R62.
<b>Postcondiciones</b>	Queda actualizada la base de datos del sistema con un nuevo nivel, guardados los cambios efectuados sobre alguno ya existente o eliminado en caso de ver elegido esta opción.

Anexo C 16 Descripción textual Gestionar Usuario.

<b>Caso de Uso</b>	Gestionar Usuario.
<b>Actores</b>	Administrador.
<b>Resumen</b>	El caso de uso se inicia cuando el administrador necesita insertar, eliminar o modificar un usuario. El caso de uso culmina con la actualización de los datos.
<b>Responsabilidades</b>	Insertar, modificar, eliminar o buscar un usuario.
<b>Descripción del flujo normal de eventos</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
1-El administrador presiona el botón Gestionar en la interfaz principal.	1.1-El sistema despliega un menú.
2- El administrador elige la opción Usuarios.	2.1- El sistema muestra una lista de todos los usuarios.
3- El administrador necesita insertar, modificar, buscar o eliminar un usuario.	3.1- El sistema ejecuta alguna de las siguientes acciones:  a- Si el administrador decide insertar un nuevo usuario, ir a la sección "Insertar".  b- Si el administrador decide modificar un usuario, ir a la sección "Modificar".

<p>4- El administrador termina el trabajo con los usuarios.</p>	<p>c- Si el administrador decide eliminar un usuario, ir a la sección “Eliminar”.</p> <p>d- Si el administrador decide buscar un usuario, ir a la sección “Buscar”.</p> <p>4.1- El sistema finaliza el caso de uso.</p>
<p><b>Sección “Insertar”</b></p>	
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Acción del actor</b></p> <p>3 El administrador presiona la opción para insertar un nuevo usuario.</p> <p>4 El administrador introduce los datos y presiona el botón insertar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.3 El sistema le muestra una interfaz con un formulario para introducir los datos del usuario.</p> <p>2.1 El sistema verifica que los datos estén correctos.</p>

	2.2- El sistema inserta el nuevo usuario en la base de datos y muestra un mensaje de éxito.
--	---

**Cursos Alternos**

En la línea 2.1 si al sistema validar los datos introducidos por el administrador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El administrador vuelve a introducir los datos y continúa con la acción 2.

En la línea 2.2 si al sistema tratar de insertar el usuario, este ya existe, se lo comunica a través de un mensaje al administrador y finaliza el caso de uso.

**Sección “Buscar”**

**Prototipo de Interfaz**

Nombre	Apellidos	Usuario	Acción
Pedro Antonio	Ramos Oropesa	planificador	<a href="#">Ver Modificar</a>
Jorgee	Torres Cabrera	almacenero	<a href="#">Ver Modificar</a>
Jose Arturo	Cabrera Torres	inf1007	<a href="#">Ver Modificar</a>
Yadian	Cabrera Torres	alm23	<a href="#">Ver Modificar</a>

**Acción del actor**

- 1 El administrador introduce el dato del usuario en el cuadro de texto que aparece en la interfaz y presiona el botón buscar.

**Respuesta del sistema**

- 1.1 El sistema selecciona en la lista mostrada de todos los usuarios, el que concuerda con el criterio introducido en el cuadro de texto.

<p><b>Cursos Alternos</b></p> <p>En la línea 1.1 si el sistema no encuentra el usuario, muestra un mensaje informando la no existencia del usuario.</p>	
<p><b>Sección “Eliminar”</b></p>	
<p><b>Acción del actor</b></p> <p>2 El administrador selecciona el usuario y presiona la opción Editar.</p> <p>2 El administrador presiona la opción Eliminar.</p>	<p><b>Respuesta del sistema</b></p> <p>1.1 El sistema muestra un formulario con los datos del usuario y la opción Eliminar.</p> <p>2.1 El sistema elimina el usuario de la base de datos y muestra al administrador la lista de usuarios actualizada.</p>
<p><b>Cursos Alternos</b></p> <p>En la línea 2.1 si el sistema no puede eliminar el usuario muestra un mensaje al administrador.</p>	
<p><b>Sección “Modificar”</b></p>	
<p><b>Descripción del flujo normal de eventos</b></p>	
<p>1 El administrador selecciona la opción editar.</p> <p>2 El administrador modifica los datos y presiona la opción Aceptar.</p>	<p>1.4 El sistema muestra una interfaz con los datos del usuario.</p> <p>2.1 El sistema valida que los datos estén correctos y no hayan campos vacíos.</p> <p>2.2 El sistema actualiza la información existente en la base de datos con los cambios efectuados.</p>

	<p>2.3 El sistema muestra la interfaz que contiene los datos del usuario actualizados.</p>
<p><b>Cursos alternos</b></p>	<p>En la línea 2.1 si al sistema validar los datos introducidos por el administrador encuentra errores o campos vacíos, le hace un señalamiento en los campos haciéndoselo saber. El administrador vuelve a introducir los datos y continúa con la acción2.</p> <p>En la actividad 2.2 si el sistema no puede actualizar los datos porque el usuario ya existe le muestra un mensaje al administrador.</p>
<p><b>Requisitos funcionales</b></p>	<p>R63 - R67.</p>
<p><b>Postcondiciones</b></p>	<p>Queda actualizada la base de datos del sistema con un nuevo usuario, guardados los cambios efectuados sobre alguno ya existente o eliminado en caso de ver elegido esta opción.</p>

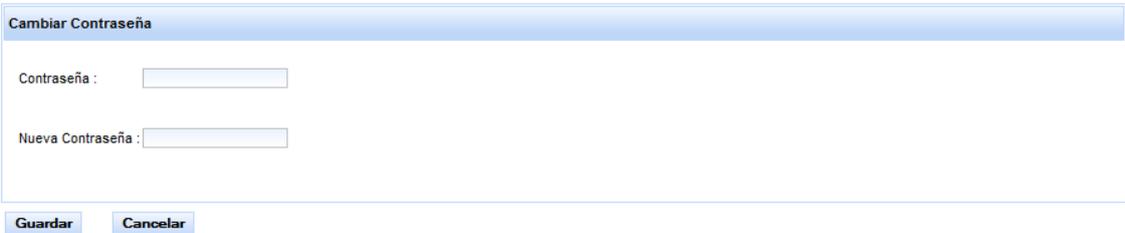
Anexo C 17 Descripción textual Asignar funcionalidades a un usuario.

<b>Caso de Uso</b>	Asignar funcionalidades a un usuario.
<b>Actores</b>	Administrador
<b>Resumen</b>	El caso de uso inicia cuando el administrador necesita asignarle funcionalidades a un usuario o modificar las ya existentes. El caso de uso finaliza el sistema guarda los datos satisfactoriamente.
<b>Responsabilidades</b>	Asignar funcionalidades a un usuario.
<b>Precondiciones</b>	El usuario debe ser creado con anterioridad.
<b>Descripción</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>
<p>1- El administrador presiona el botón Sistema en la interfaz principal.</p> <p>2- El administrador elige la opción Asignación de funcionalidades a un usuario.</p> <p>3- El administrador selecciona las funcionalidades y presiona el botón aceptar.</p>	<p>1.1-El sistema despliega un menú.</p> <p>2.1- El sistema muestra un formulario con el nombre del usuario y sus funcionalidades.</p> <p>3.1- El sistema muestra un mensaje de confirmación.</p>
<b>Prototipo de Interfaz</b>	

<b>Usuario</b> planificador	
<b>Funcionalidades</b>	
Almacen	<input type="checkbox"/>
Gestionar	<input checked="" type="checkbox"/>
Sistema	<input checked="" type="checkbox"/>
ActualizarValesProvisionales	<input type="checkbox"/>
ElaborarHojaCalculo	<input type="checkbox"/>
<b>Requisitos funcionales</b>	R68
<b>Postcondiciones</b>	Son asignadas nuevas funcionalidades a un usuario o eliminadas las ya existentes.

Anexo C 18 Descripción textual Cambiar contraseña.

<b>Caso de Uso</b>	Cambiar contraseña
<b>Actores</b>	Usuario
<b>Resumen</b>	El caso de uso se inicia cuando el usuario desea cambiar su contraseña de entrada al sistema. Para ello debe insertar su, contraseña y la nueva contraseña. El caso de uso finaliza cuando son guardados los datos.
<b>Precondiciones</b>	El usuario debe estar logueado en el sistema.
<b>Descripción</b>	
<b>Acción del Actor (es)</b>	<b>Respuesta del Sistema</b>

<p>1- El usuario presiona la opción Sistema en la interfaz principal.</p> <p>2- El usuario presiona la opción Cambiar contraseña.</p> <p>3- El usuario introduce su contraseña, la nueva y presiona la opción aceptar.</p>	<p>1.1-El sistema muestra un menú.</p> <p>2.1- El sistema muestra un formulario.</p> <p>3.2- El sistema valida los campos y guarda los datos.</p>
<p><b>Prototipo de Interfaz</b></p> 	
<p><b>Cursos Alternos</b></p> <p>En la acción 3.2 si la contraseña actual no es la correcta o la nueva contraseña tiene menos de 8 caracteres, el sistema muestra un mensaje al usuario.</p>	
<p><b>Requisitos funcionales</b></p>	<p>R69</p>
<p><b>Postcondiciones</b></p>	<p>Queda cerrada la sesión para el usuario logueado.</p>

Anexo C 19 Descripción textual Cerrar sesión.

<p><b>Caso de Uso</b></p>	<p>Cerrar sesión</p>
---------------------------	----------------------

<b>Actores</b>	Usuario
<b>Resumen</b>	El caso de uso se inicia cuando el usuario desea salir del sistema. Se brinda la opción de cerrar la sesión del usuario que esté registrado. Terminando así el caso de uso.
<b>Responsabilidades</b>	Cerrar sesión.
<b>CU asociados</b>	No hay
<b>Precondiciones</b>	El usuario debe estar logueado en el sistema.
<b>Descripción</b>	
<b>Acción del Actor (es)</b> 1- El usuario presiona la opción para cerrar la sesión en la interfaz principal.	<b>Respuesta del Sistema</b> 1.1-El sistema cierra la sesión del usuario logueado.
<b>Requisitos funcionales</b>	R71
<b>Postcondiciones</b>	Queda cerrada la sesión para el usuario logueado.

Anexo C 20 Descripción textual Autenticarse.

<b>Caso de Uso</b>	Autenticarse
<b>Actores</b>	Usuario
<b>Resumen</b>	El caso de uso se inicia cuando el usuario desea ingresar al sistema. Para ello debe introducir su usuario y contraseña, a continuación se chequea. Si los datos son correctos el usuario podrá acceder a las

	opciones del sistema que le corresponden, en el caso de que no lo sean se mostrará un mensaje de error, denegando el acceso.
<b>Responsabilidades</b>	Permitir el acceso a las funcionalidades del sistema, teniendo en cuenta el rol de cada usuario.
<b>CU asociados</b>	No hay
<b>Precondiciones</b>	El Usuario debe estar registrado en el sistema, teniendo un identificador y contraseña.
<b>Descripción</b>	
<b>Acción del Actor (es)</b> 1- El usuario necesita entrar al sistema e introduce sus datos.	<b>Respuesta del Sistema</b> 1.1-El sistema chequea los datos y da entrada según el rol del usuario registrado.
<b>Prototipo de Interfaz</b>	

## Sistema Informático para la Gestión del Servicio de Alimentación en la Residencia de Postgrado

Introduzca los datos para ingresar en el sistema.



Nombre Usuario:

Contraseña:

Acceso

### Cursos Alternos

En la acción 1.1 el sistema mostrará un mensaje de error, denegando el acceso.

### Requisitos funcionales

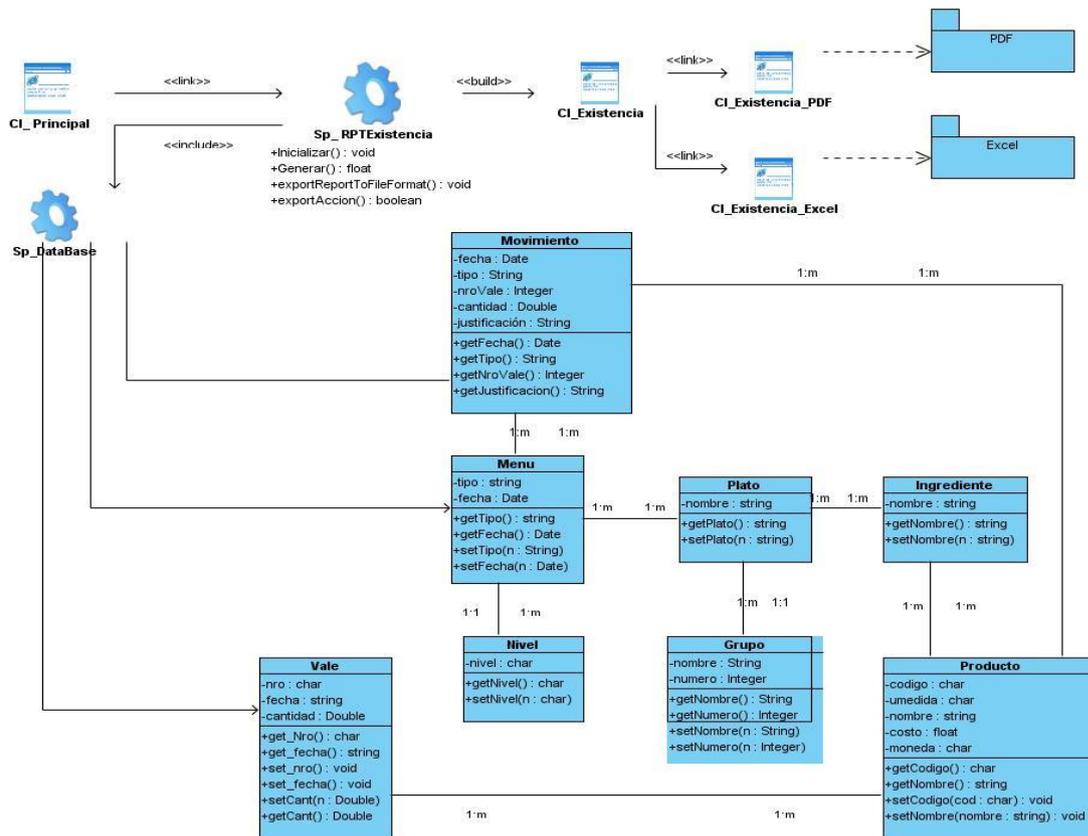
R70

### Postcondiciones

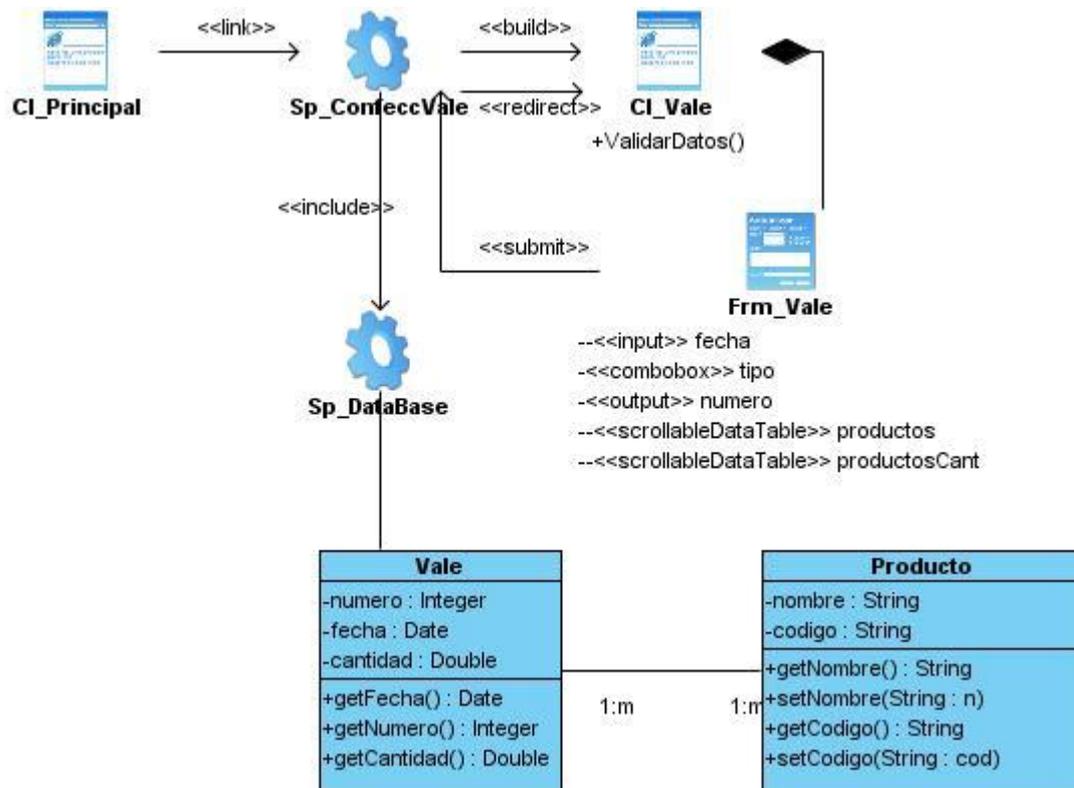
Queda autenticado el usuario en el sistema.

## Anexo D: Diagramas de clases web

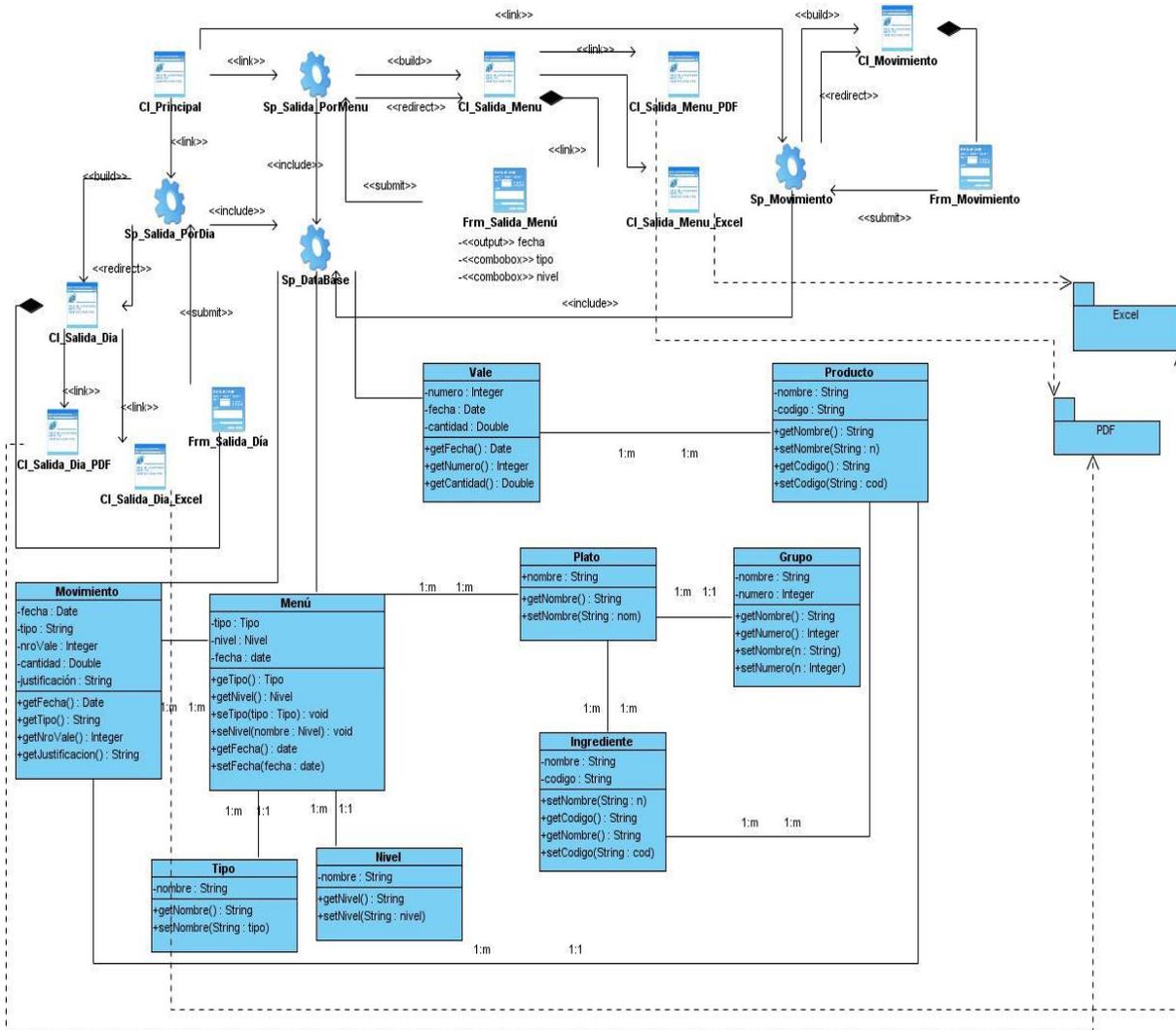
### Anexo D1 Mostrar existencia



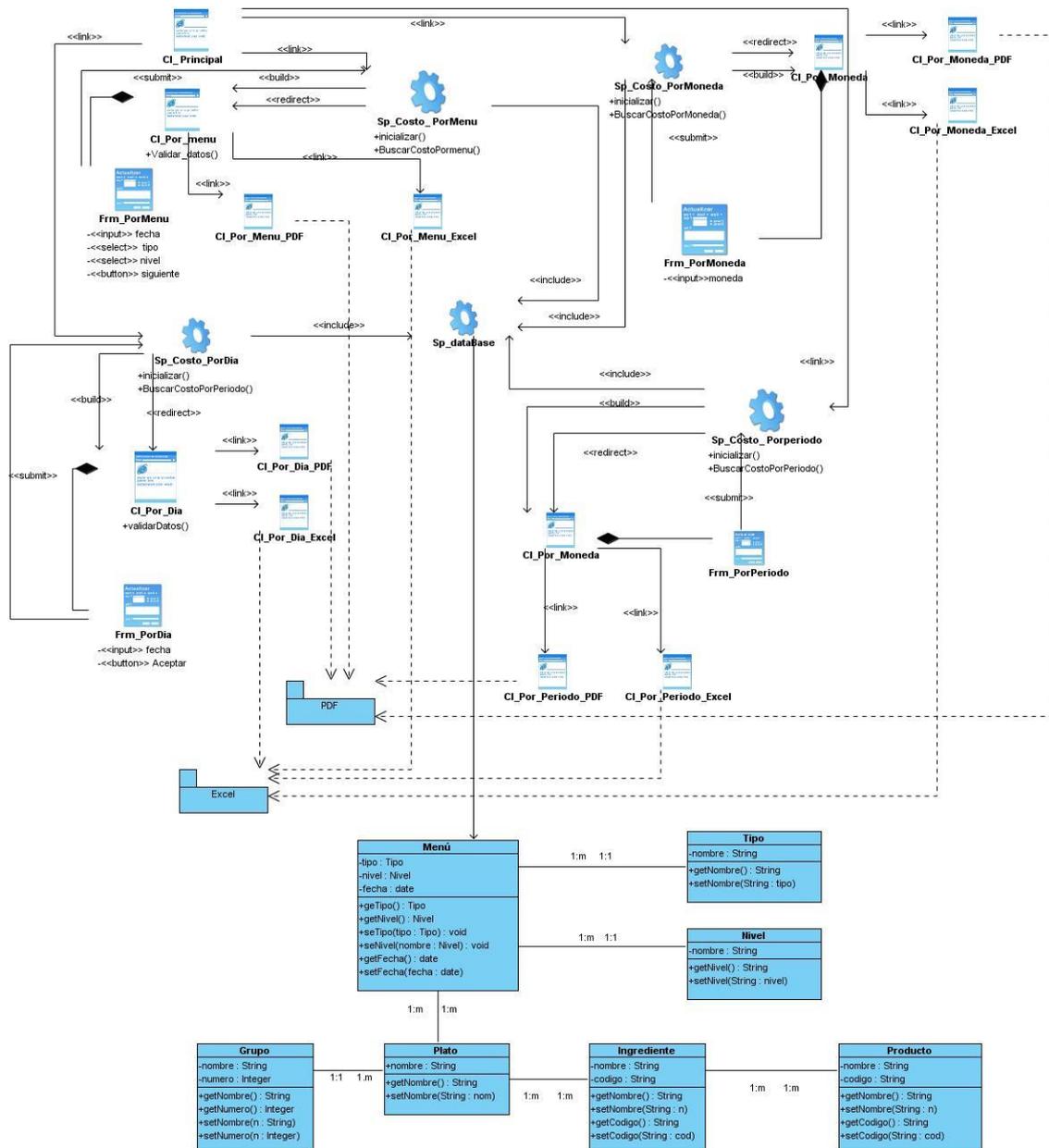
Anexo D2 Entrada de productos



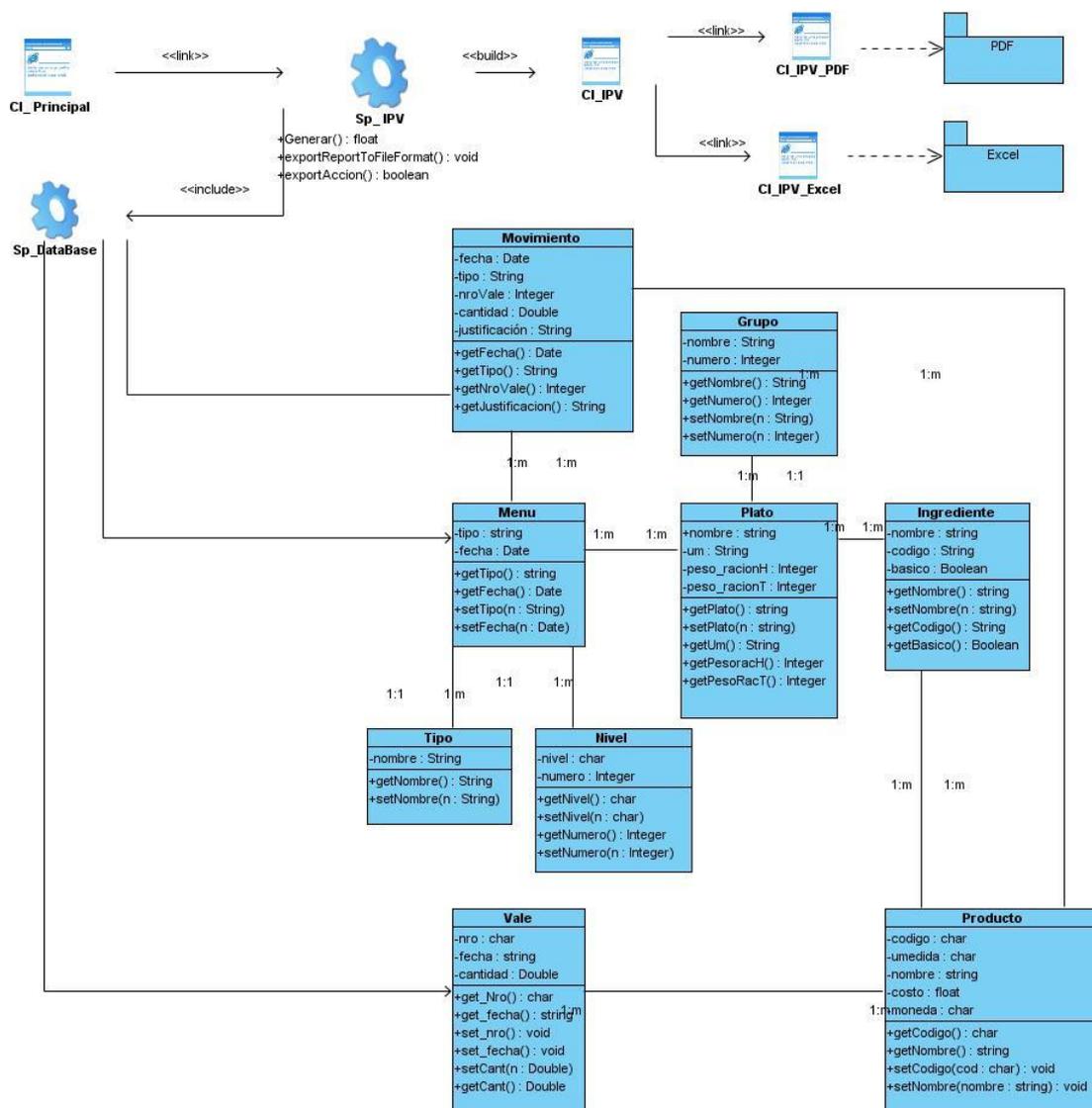
Anexo D3 Salida de productos



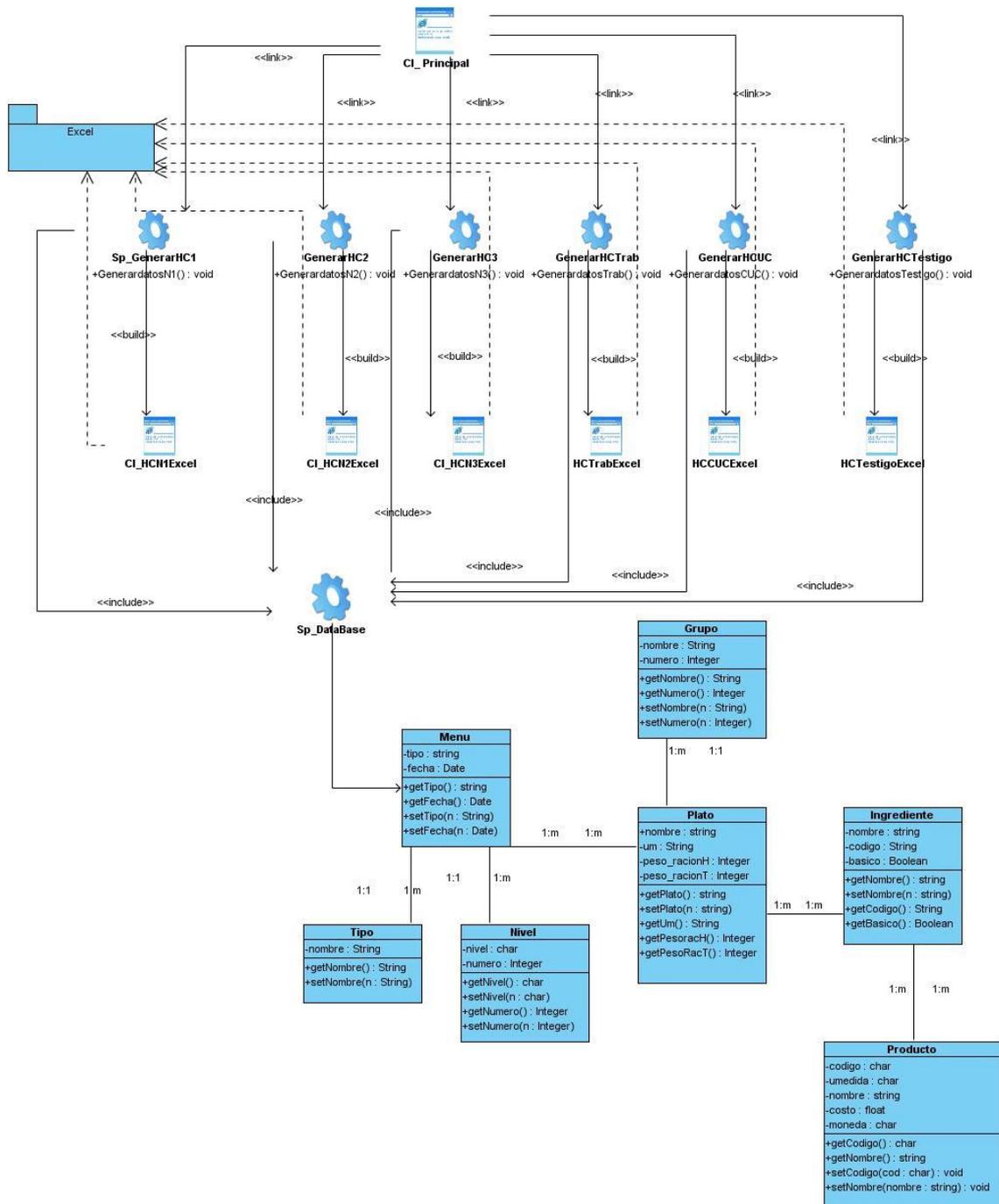
Anexo D4 Determinar Costos



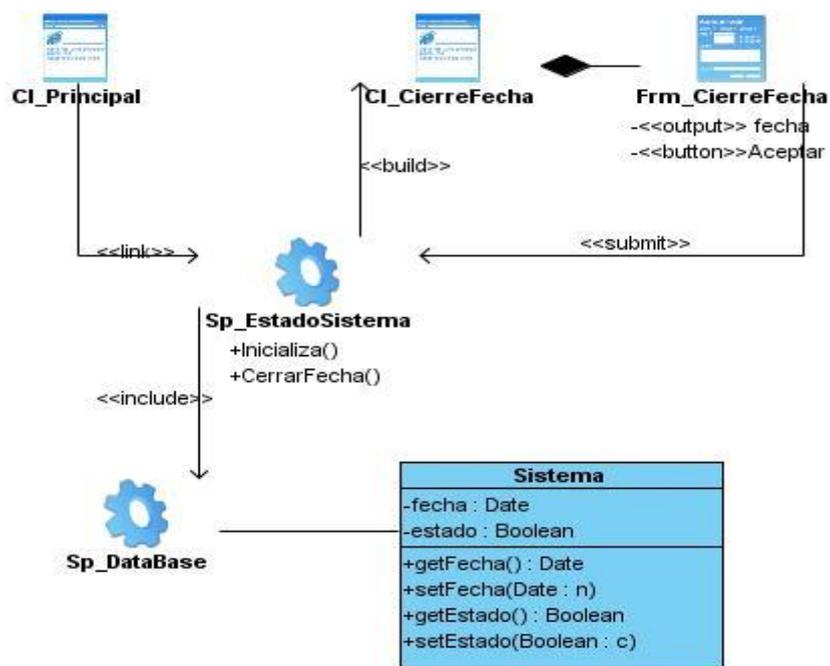
Anexo D5 Mostrar IPV



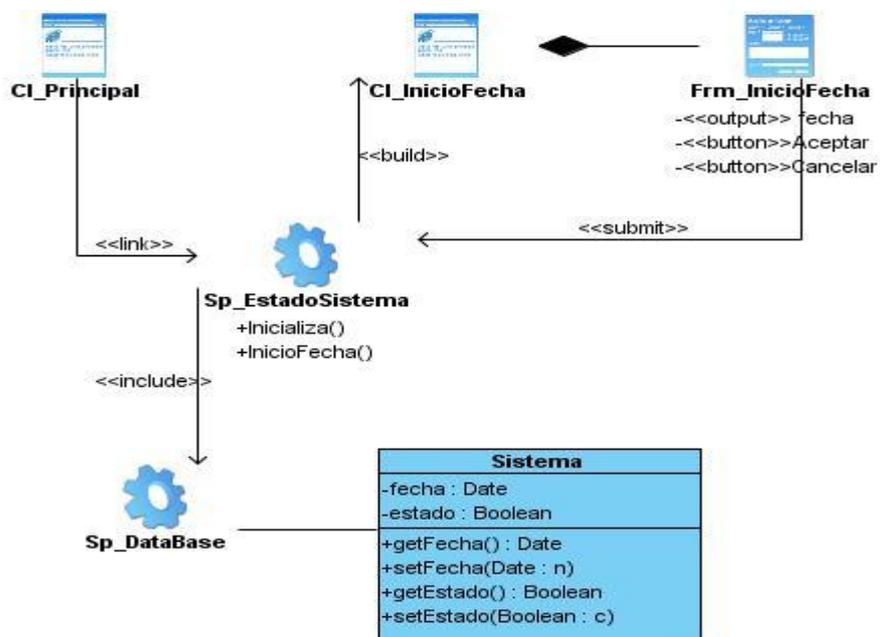
Anexo D6 Generar HC



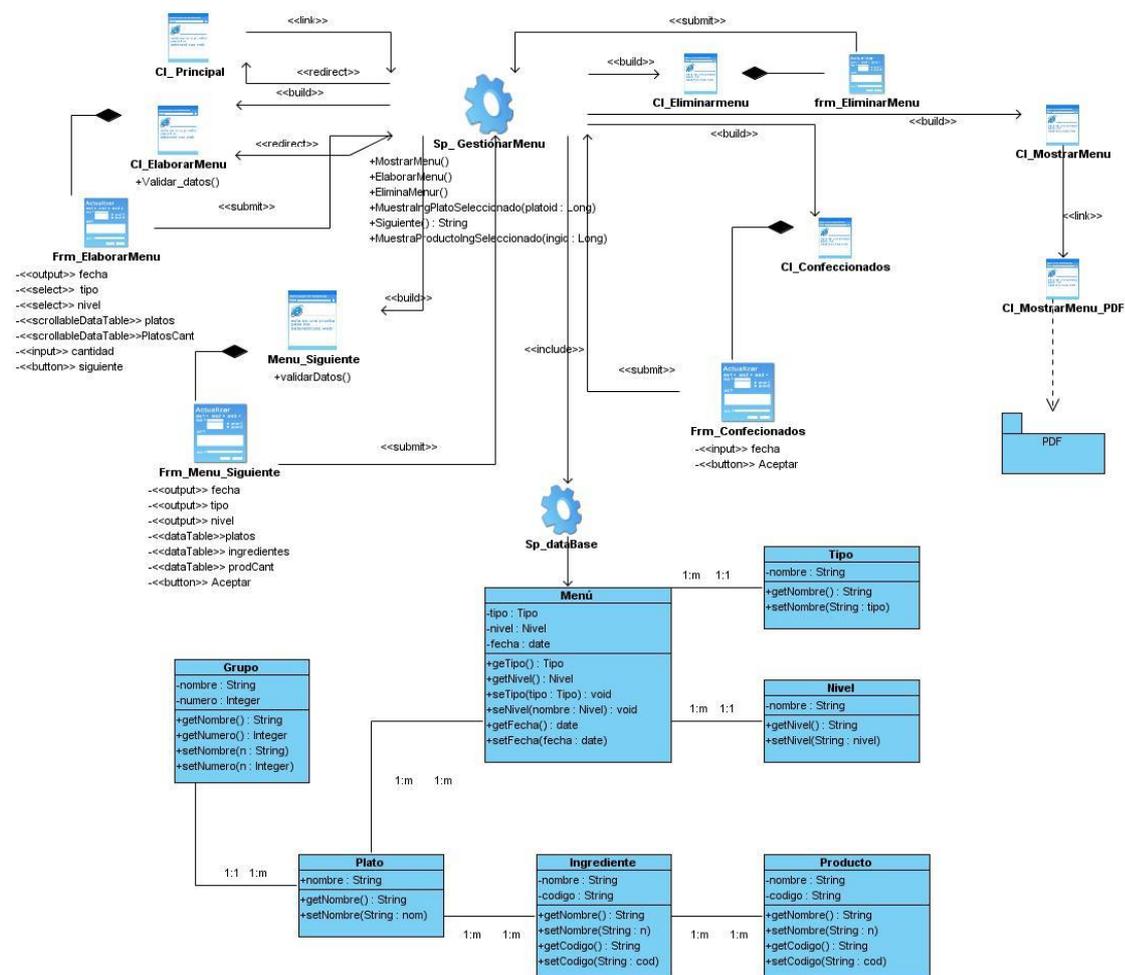
Anexo D7 Cierre de Fecha



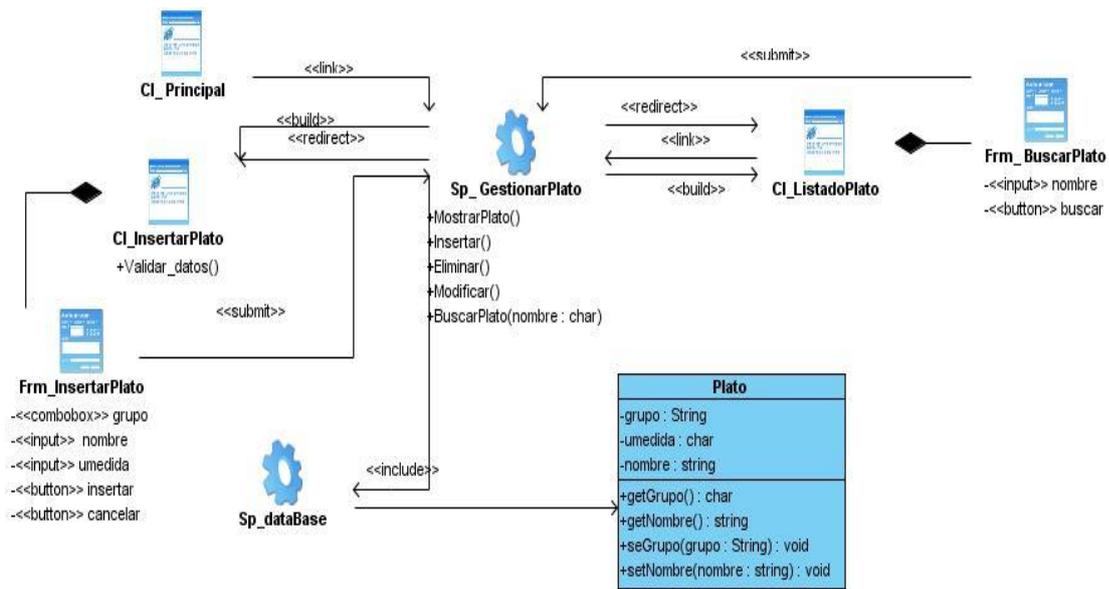
Anexo D8 Inicio de Fecha



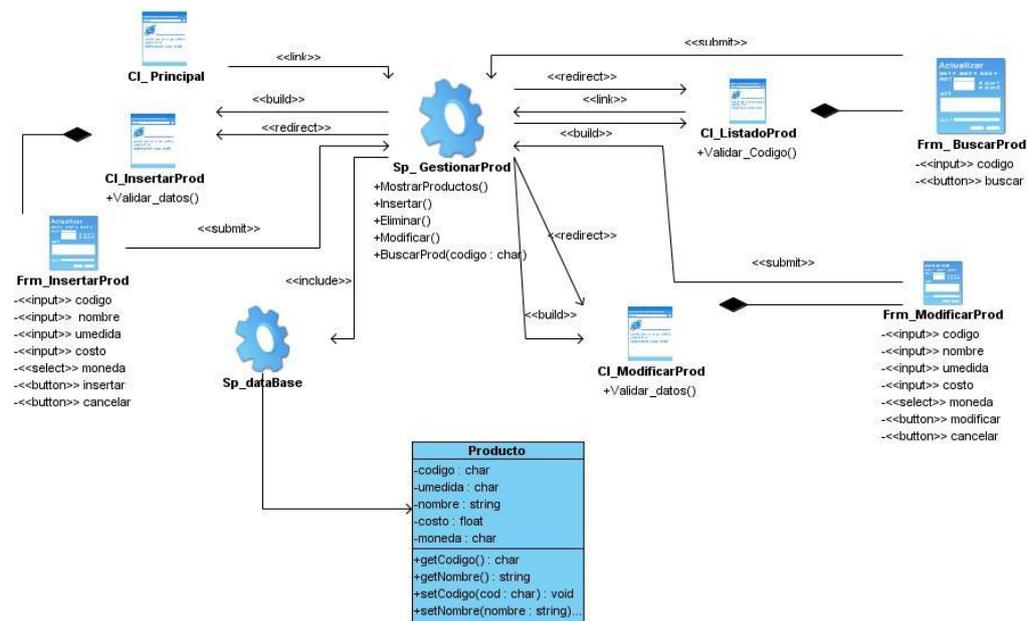
## Anexo D9 Gestionar Menú



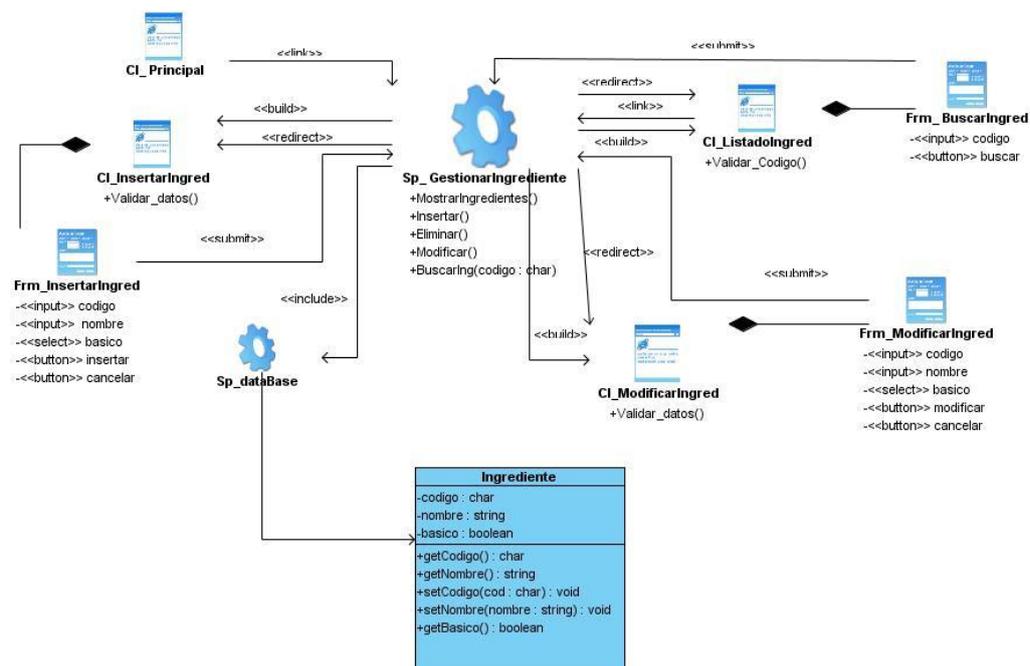
### Anexo D10 Gestionar Plato



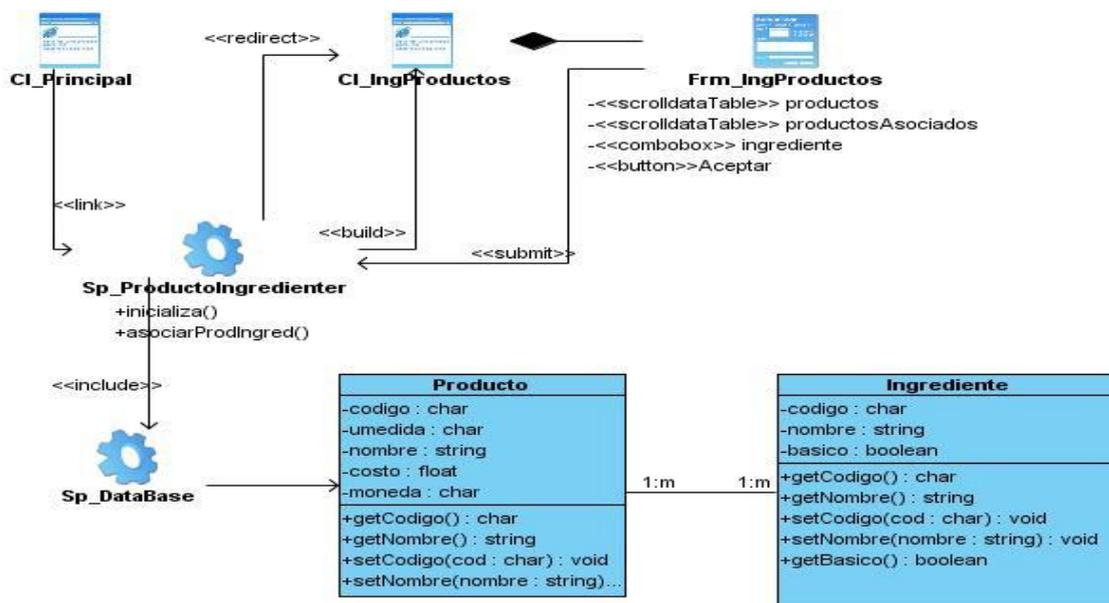
### Anexo D11 Gestionar Producto



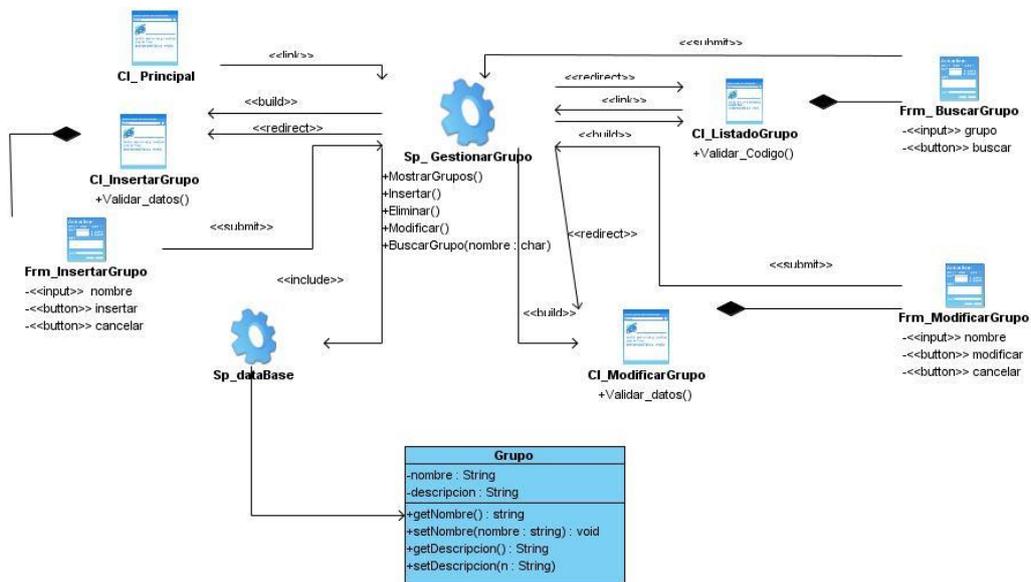
## Anexo D12 Gestionar Ingrediente



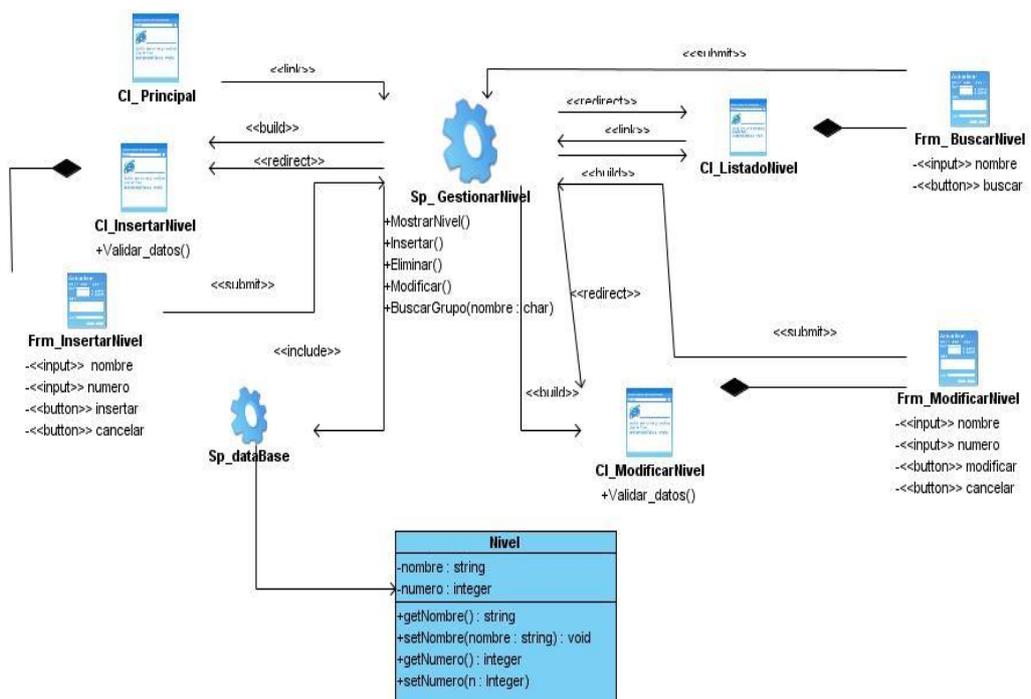
## Anexo D13 Asociar productos a un ingrediente



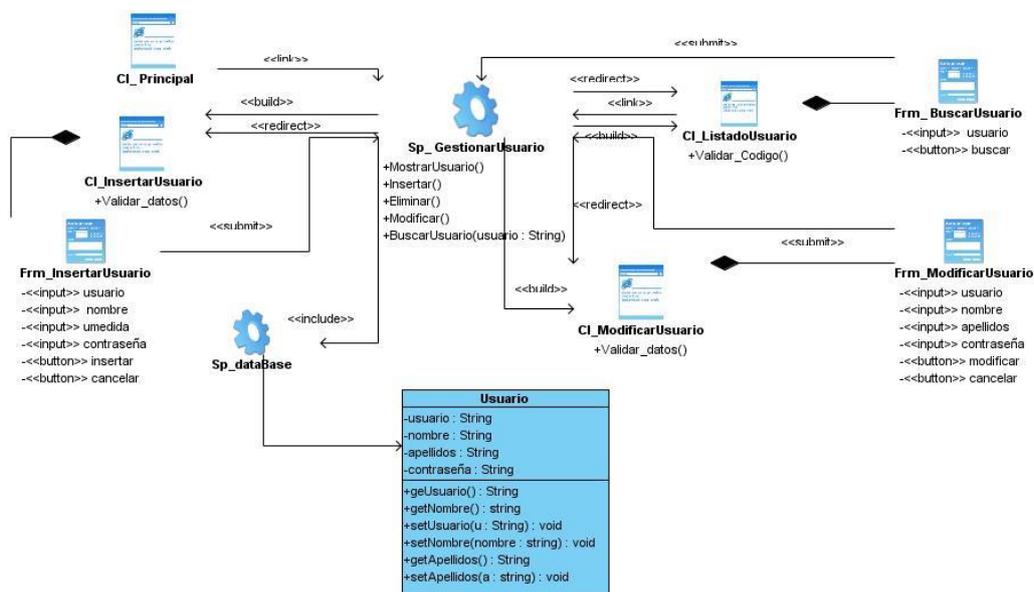
### Anexo D14 Gestionar Grupo



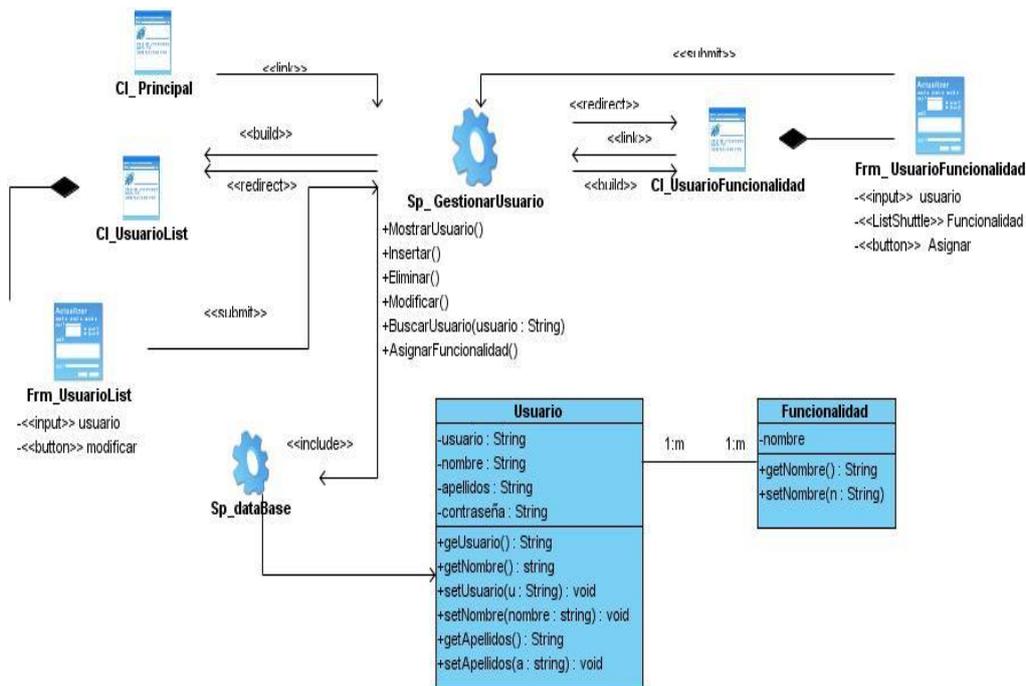
### Anexo D15 Gestionar Nivel



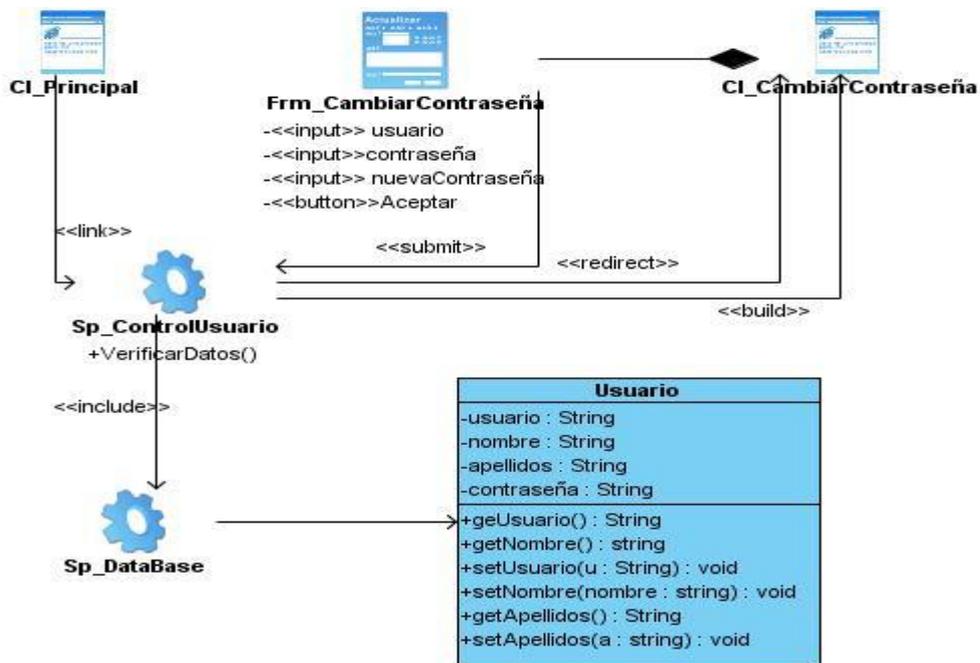
### Anexo D16 Gestionar Usuario



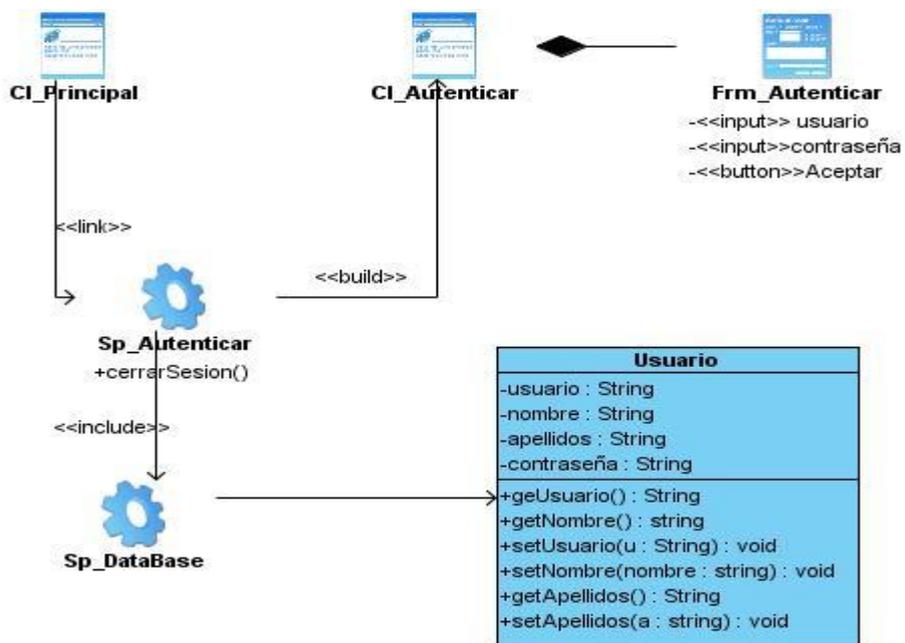
### Anexo D17 Asignar funcionalidades a un usuario



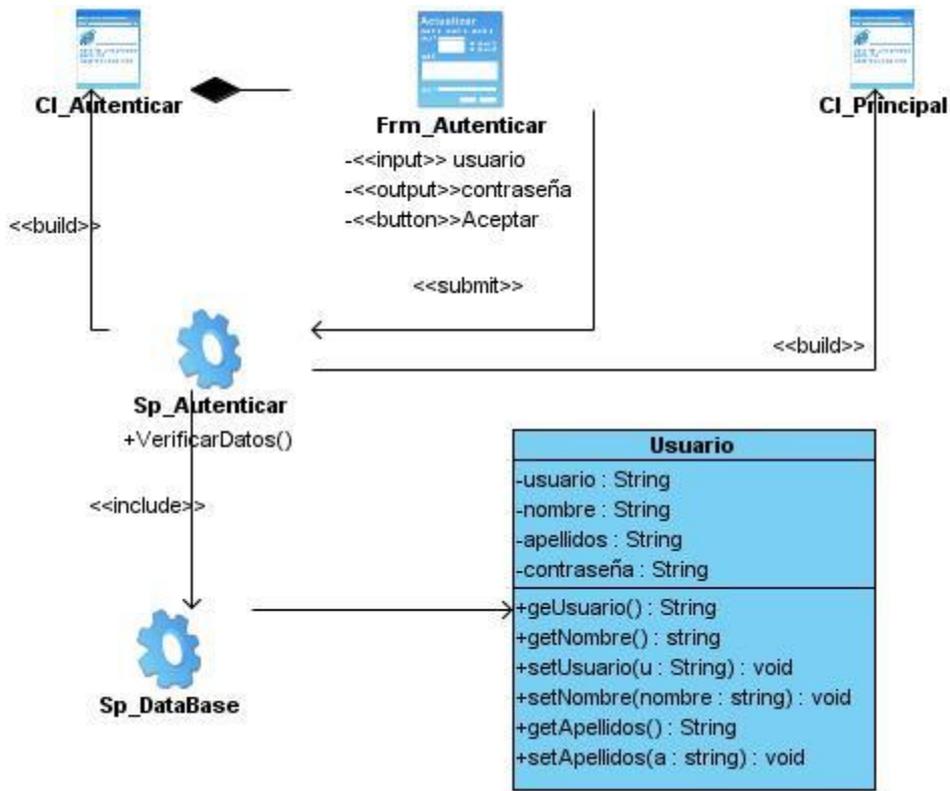
Anexo D18 Cambiar contraseña



Anexo D 19 Cerrar sesión

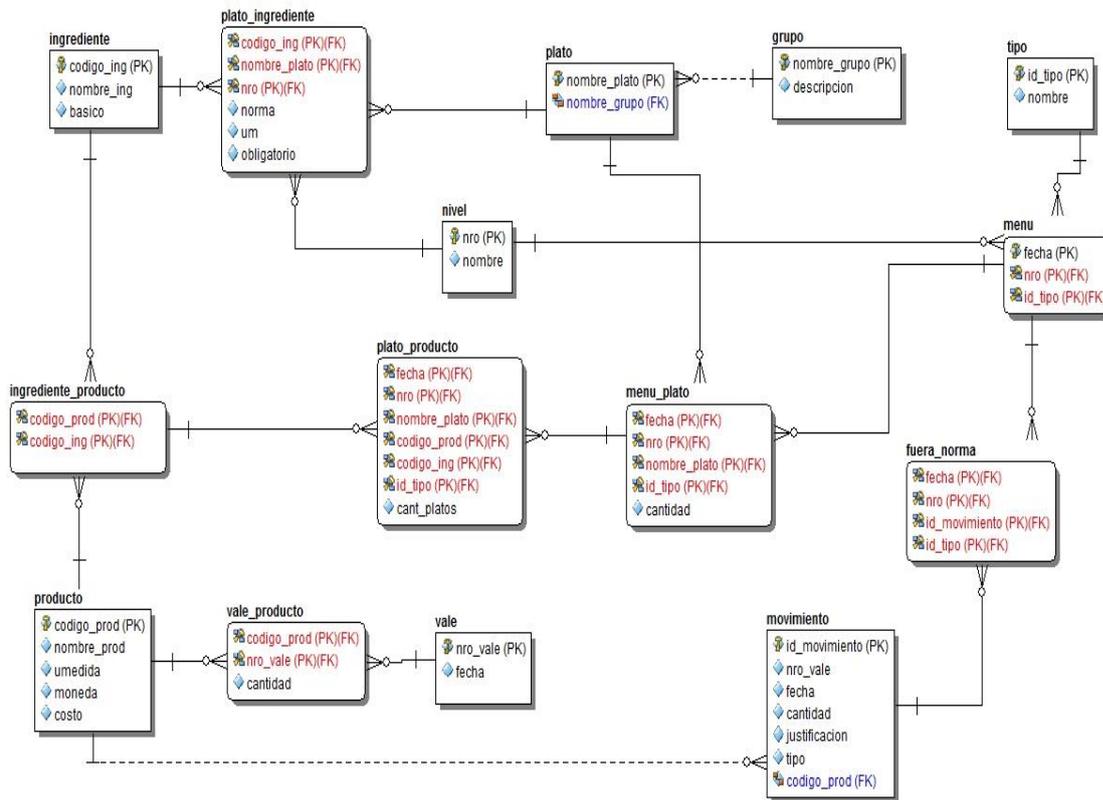


Anexo D 20 Autenticarse



**Anexo E: Diseño de la base de datos.**

**Anexo E 1 Diagrama del modelo lógico de datos.**



Anexo E 2 Diagrama del modelo físico de datos.

