

UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ MARTÍ PÉREZ”
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA INFORMÁTICA



Título

“Aplicación de escritorio para la gestión de las ventas de la masa porcina en la Empresa Provincial Porcina en Sancti Spíritus”

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERÍA
EN INFORMÁTICA

Autor:

Reynel Rodríguez Morgado

Tutor:

Lic. Alain Pereira Toledo

Consultante:

Ing. Yanetsy Jiménez Hidalgo

Sancti Spíritus, Cuba
2013

Pensamiento

“Cuando algo te gusta seguro es ilegal, inmoral o malo para tu salud.”

La ley de Murphy.

Dedicatoria

... a toda mi familia, amigos y conocidos.

Agradecimientos

- ❖ Al Autor y Consumador de mi vida.
- ❖ A mi madre por el gran apoyo que brindó y brinda constantemente en mi vida, a mi padre por poder compartir conmigo en todo momento.
- ❖ A mi hermano Reydel por soportarme en todo momento.
- ❖ A mi abuelo materno Maurilio y paterno Pedro.
- ❖ A Héctor y su familia por poderme soportar por todo este tiempo y por su apoyo incondicional.
- ❖ A mis hermanos Eduardo, Yander, Angel, Sonita, Sunny y demás por su gran apoyo.
- ❖ A mis compañeros de aula Claudia, Ivonne, Yangzet y en especial mi hermano Yonelki por su apoyo y soporte en todas la ocasiones de dudas y dificultades en la tesis.
- ❖ A la una de las profesoras de la Cátedra de Matemática-Física....
- ❖ A mis tutores Alain y Yanetsy por este tiempo de “corre corre”, por sus asesoramientos y consejos en la parte profesional. Debo reconocerlo a pesar del poco tiempo aprendí de ellos lo mejor.
- ❖ Otro de los profesores que me apoyo incondicionalmente en todo tiempo fue el profesor Arley de Programación, gracias por su ayuda y conocimientos.
- ❖ A todos mis profesores durante la carrera por su ayuda, soporte y colaboración en cuanto al conocimiento que llevara a la práctica en el mundo profesional.
- ❖ A todos los que no he mencionado y en algún momento pensaron en mí.

...A todos muchas gracias.

Resumen

En la Empresa Provincial Porcina en Sancti Spíritus, que radica en Carretera del Jíbaro Km #1, en el municipio de Sancti Spíritus; se almacena los datos de todas las ventas de masa porcina efectuadas en la provincia. Todo este proceso se realiza de forma manual por parte de un especialista; limitando en gran medida los análisis diarios de esta importante información. Por estas razones se hizo necesario implementar una aplicación de escritorio para la gestión de los datos de las ventas que se realizan en la empresa. Con este sistema se desea, mediante una aplicación visual en java modelada a través de la metodología RUP, agilizar la gestión de los datos almacenados en planillas y documentos en la empresa. De esta manera se hace un mejor control y manejo de las estadísticas de cada tipo de venta que se realiza, y se facilita el análisis y observación de los datos estadísticos de las ventas. Además, el resultado podrá ser aplicado en cualquier otra empresa de nuestro país dedicada a la venta de masa porcina.

Abstract

In the Swinish Provincial Company in Sancti Spíritus that resides in Highway of the Jíbaro Km #1, in the municipality of Sancti Spíritus; the data of all the sales of swinish mass made in the county is stored. This whole process is carried out in a manual way on the part of a specialist; limiting in great measure the daily analyses of this important information. For these reasons it became necessary to implement a desk application for the administration of the data of the sales that you/they are carried out in the company. With this system it is wanted, by means of a visual application in modeling java through the methodology RUP, to speed up the administration of the data stored in schedules and documents in the company. This way it is made a better control and handling of the statistics of each sale type that he/she is carried out, and it is facilitated the analysis and observation of the statistical data of the sales. Also, the result will be able to be applied in any other company of our country dedicated for sale of swinish mass.

INTRODUCCIÓN	1
Capítulo I: Fundamentación teórica de una aplicación de escritorio para facilitar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.	4
1.1 Gestión	4
1.2 Gestión de la información.....	4
1.3 Gestión de datos en las ventas de la Empresa Provincial Porcina en Sancti Spíritus	5
1.3.1 Comercialización de los cerdos.	6
1.3.2 Deficiencias en el proceso de gestión de los datos de las ventas.....	7
1.4 Metodologías utilizadas para el diseño	8
1.4.1 Metodología RUP	8
1.4.2 Lenguaje Unificado de Modelado (UML).....	9
1.5 Lenguaje de programación Java	9
1.6 Lenguaje de consulta estructurado SQL.....	10
1.7 Lenguaje de consulta estructurado PL/pgSQL	10
1.8 Arquitectura cliente-servidor.....	11
1.8.1 Arquitectura cliente-servidor en 2 capas	11
1.9 Patrón de diseño Modelo-Vista-Controlador.	14
1.10 Sistemas Gestores de Bases de Datos.	14
1.10.1 MySQL	15
1.10.2 PostgreSQL.....	15
1.11 NetBeans 6.8.....	17
1.12 Rational Rose	18
1.13 Conclusiones parciales	19
Capítulo II Descripción de la aplicación de escritorio para facilitar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.	20
2.1 Descripción del modelo de negocio	20
2.1.1 Descripción de los procesos de negocio	20
2.2 Modelo de casos de uso del negocio.....	20
2.2.1 Actores del negocio.....	21
2.2.2 Trabajadores del negocio.....	21
2.2.3 Diagrama de casos de uso del negocio	22

2.2.4 Descripción de los casos de uso del negocio	23
2.2.5 Diagrama de Actividad	24
2.3 Diagrama de clases del modelo de objetos	24
2.4 Requerimientos funcionales	24
2.5 Requisitos no funcionales	27
2.6 Descripción de la aplicación de escritorio propuesta	30
2.6.1 Actores del sistema	30
2.6.2 Modelo de casos de uso del sistema por paquetes	31
2.6.3 Diagramas de casos de uso del sistema	32
2.6.4 Descripción de los casos de uso del sistema	34
2.7 Conclusiones parciales	36
Capítulo III Construcción de la aplicación de escritorio para facilitar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.....	37
3.1 Diagrama de clases del diseño	37
3.1.1 Paquete Seguridad.....	38
3.1.2 Paquete Administración	40
3.1.3 Paquete Gestión.....	43
3.1.4 Paquete Reportes	46
3.2 Diseño de la Base de Datos.....	53
3.2.1 Diagrama de clases persistentes	53
3.2.2 Diagrama físico de datos.....	54
3.3 Modelo de Implementación	54
3.3.1 Diagrama de despliegue	54
3.3.2 Diagrama de componentes	55
3.4 Diseño de la interfaz del sistema.	57
3.5 Tratamiento de errores.	57
3.6 Concepción general de la ayuda.....	57
3.7 Concepción del sistema de seguridad y protección.....	57
3.8 Conclusiones parciales.	58
CONCLUSIONES.....	59
RECOMENDACIONES	60
REFERENCIAS BIBLIOGRÁFICAS.....	61

ANEXO 63

TABLAS

Tabla 2.1. Actores del negocio	21
Tabla 2.2. Trabajadores del negocio	21
Tabla 2.3 Descripción del CU "Solicitar información de las ventas de cerdos realizadas"	23
Tabla 2.4. Actores del sistema.	30

Figura #1.1: La arquitectura cliente-servidor con dos capas	12
Figura #1.2: Conexión con un DBMS en la arquitectura cliente-servidor con dos capas.....	13
Figura #2.1.Diagrama de casos de uso del negocio.....	22
Figura # 2.2. Diagrama de caso de uso por paquetes.	31
Figura # 2.3. Diagrama de casos de uso. Paquete Seguridad.	32
Figura # 2.4. Diagrama de casos de uso. PaqueteAdministración.	33
Figura # 2.5. Diagrama de casos de uso. Paquete Gestión.	33
Figura # 2.6. Diagrama de casos de uso. Paquete Reportes.	34
Figura # 3.1: Diagrama de clases del diseño del CU “Autenticar usuario”	38
Figura # 3.2: Diagrama de clases del diseño del CU “Cambiar contraseña”	39
Figura # 3.3: Diagrama de clases del diseño del CU “Gestionar vendedor”	40
Figura # 3.4: Diagrama de clases del diseño del CU “Gestionar comprador”	41
Figura # 3.5: Diagrama de clases del diseño del CU “Gestionar usuario”.....	42
Figura # 3.6: Diagrama de clases del diseño del CU “Gestionar venta por matadero”	43
Figura # 3.7: Diagrama de clases del diseño del CU “Gestionar venta por nevera”	44
Figura # 3.8: Diagrama de clases del diseño del CU “Gestionar venta por municipio”	45
Figura # 3.9: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en el Municipio”	46
Figura # 3.10: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en la Nevera”	47
Figura # 3.11: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en el Matadero”	48
Figura # 3.12: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas al MINAL- DIVISA en un mes y año determinado”.....	49
Figura # 3.13: Diagrama de clases del diseño del CU “Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento”.....	50
Figura # 3.14: Diagrama de clases del diseño del CU “Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio”.....	51
Figura # 3.15: Diagrama de clases del diseño del CU “Mostrar comparación del mayor importe de ventas realizadas”	52

Figura # 3.16. Diagrama de clases persistentes.....	53
Figura # 3.17. Diagrama de despliegue.....	55
Figura # 3.18. Diagrama de componentes.....	56

INTRODUCCIÓN

Actualmente en el mundo no se puede hablar de desarrollo sin mencionar las Tecnologías de la Información y las Comunicaciones (TIC), estas se han ganado un lugar importante en la vida cotidiana del hombre y también en el mundo empresarial, ofreciendo numerosas propuestas que garantizan el éxito en la informatización de la dirección y organización de la información que se maneja y procesa.

Muchas son las definiciones que se han referido a las TIC, una de ellas sostiene que son "aquellas herramientas y métodos empleados para recabar, retener, manipular o distribuir información" (Bologna & Walsh, 1997). De modo que generalmente las TIC se asocian con las computadoras y las tecnologías afines. Para muchas empresas, la información y la tecnología que la soporta, representan sus activos más valiosos. Es por ello que las mismas han incrementado sus expectativas relacionadas con la entrega de servicios de las Tecnologías de la Información (TI). En consecuencia es necesario que dichos servicios presenten incrementos en calidad, funcionalidad y facilidades de uso, así como una mejora continua y disminución de los tiempos de entrega.

En Cuba se dan pasos para el ordenamiento de un trabajo continuo destinado a impulsar el uso y desarrollo de las TIC en cada uno de los centros laborales y docentes del país, atendiendo no solo a las grandes ventajas que posee el uso de estas, sino también con el fin de incorporarse en el mercado a nivel mundial. Una de las organizaciones interesadas en la utilización de las TIC, existente hoy en el país, es la Empresa Provincial Porcina en Sancti Spíritus.

Actualmente en el Departamento de Planificación y Estadísticas de la Empresa Provincial Porcina en Sancti Spíritus el proceso de gestión de los datos de las ventas de la masa porcina se torna engorrosa, pues no existe una herramienta informática que viabilice dicho proceso y los especialistas deben dedicar mayor esfuerzo debido a que el trabajo se realiza con la Hoja de Cálculo Excel. Este es el único medio con que se cuenta para controlar los datos, de manera que no hay un sistema que satisfaga las necesidades de agilidad y seguridad.

Todo lo anterior trae consigo la ineficiencia en el proceso, pues el grado de interrelación es pobre; y en las diferentes facturas existe repetición innecesaria de los datos, pérdida de información, y dificultad en el acceso y manejo. Además existe falta de seguridad, pues

al no existir ninguna medida para el acceso a los datos puede ocurrir que el personal no autorizado acceda a ellos para modificarlos o dañarlos.

Por tanto se enuncia el siguiente **problema de investigación** el cual está centrado en: ¿cómo perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina en Sancti Spíritus?

Luego, el **objetivo general** es desarrollar una aplicación de escritorio para gestionar los datos de las ventas que se realizan en la Empresa Provincial Porcina en Sancti Spíritus.

Para cumplir el objetivo general se realizan las siguientes **preguntas de investigación**:

- 1) ¿Qué fundamentos teóricos y metodológicos sustentan la elaboración de una aplicación de escritorio que perfeccione la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus?
- 2) ¿Cómo diseñar una aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus?
- 3) ¿Cómo implementar una aplicación de escritorio que perfeccione la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus?

Para dar cumplimiento a las preguntas de investigación se plantean las siguientes **tareas de investigación**:

- 1) Determinar de los fundamentos teóricos y metodológicos que sustentan la elaboración de una aplicación de escritorio que perfeccione la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.
- 2) Diseñar una aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.
- 3) Implementar una aplicación de escritorio que perfeccione la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.

El informe posee la siguiente estructura:

Capítulo I Fundamentación teórica de una aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.

En este capítulo se exponen los fundamentos teóricos del presente trabajo. Además se revisan las tecnologías, lenguajes y herramientas existentes; se describen sus características y el por qué se seleccionaron.

Capítulo II Descripción de la aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.

Describe todos los elementos que componen el proceso estudiado, utilizando la metodología RUP. Se determinan los actores y trabajadores del negocio, las operaciones principales que producen resultados en forma de casos de uso y su descripción literal, los diagramas de casos de uso y el de modelo de objetos, se determinan los requerimientos funcionales y no funcionales; además, se definen los actores y casos de uso del sistema y se describe cada uno de ellos.

Capítulo III Construcción de la aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.

Se describe el diagrama de clases de diseño, el diagrama de clases persistentes, el modelo de datos y la descripción de cada una de las tablas; se enuncian también los principios de diseño según los estándares usados en la interfaz de la aplicación. Se concibe la ayuda y se detalla el tratamiento de excepciones y el manejo de la seguridad; además, se describe la implementación, y se muestra el diagrama de despliegue y el diagrama de componentes.

Capítulo I: Fundamentación teórica de una aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus

El presente capítulo contiene la fundamentación teórica sobre el tema a desarrollar. Describe los conceptos asociados al problema, brinda un análisis de los sistemas automatizados existentes vinculados al campo de acción. Además, se realiza un estudio sobre las tecnologías, lenguajes y herramientas existentes determinando cuáles van a ser las utilizadas en el desarrollo del sistema.

1.1 Gestión

Según el Diccionario de uso del español de América y España define gestión como: “conjunto de operaciones que se realizan para dirigir y administrar un negocio o una empresa: una buena gestión hace que las empresas funcionen” (“gestión”, 2003).

Por otra parte la Real Academia Española (2001) lo define como: acción y efecto de gestionar o de administrar.

En la actual investigación se señala el concepto propuesto por Marecos como el más apropiado para el tema:

“(…) gestionar es coordinar todos los recursos disponibles para conseguir determinados objetivos, implica amplias y fuertes interacciones fundamentalmente entre el entorno, las estructuras, el proceso y los productos que se desean obtener” (Marecos, 2001).

1.2 Gestión de la información

Existen varias conceptualizaciones del término gestión de la información por varios autores entre ellos esta Ajá Quiroga el cual plantea “que la gestión de la información, comprende las actividades relacionadas con la obtención de información adecuada, a un precio adecuado, en tiempo y lugar adecuado, para tomar la decisión adecuada” (Ajá Quiroga, 2006).

Por otra parte Capote Marrero la define como “el proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, así como para mejorar los procesos, productos y servicios de la organización” (Capote Marrero, 2006).

Sobre la importancia de los Sistemas de gestión de Información, se comparte la idea de (Castillo, 2008, p. 25) cuando expresa que con los adelantos tecnológicos actuales en el mundo de la informática y las comunicaciones, es prácticamente imposible que una organización no haga uso de los mismos para el desarrollo de sus actividades cotidianas, pues constituyen un elemento fundamental para alcanzar un alto nivel de competitividad y posibilidades de desarrollo. Cuando las organizaciones son grandes y complejas, poseen grandes cantidades de usuarios y niveles, la gestión de la información requiere el tratamiento, almacenamiento y difusión de grandes volúmenes de datos. En estos casos los métodos manuales para gestionar adecuadamente la información y responder con rapidez y agilidad a las diversas solicitudes, así como optimizar la comunicación con los demás sectores de la institución, no son viables.

En la presente investigación se enfatiza como gestión de la información aquella que comprende las actividades relacionadas con la obtención de la información adecuada, a un precio adecuado, en el tiempo y lugar adecuado, para tomar la decisión adecuada (Woodman, 1985).

1.3 Gestión de datos en las ventas de la Empresa Provincial Porcina en Sancti Spíritus

La Empresa Porcina Sancti Spíritus, creada por Resolución 313 de fecha 15 de diciembre de 1976 y adscripta por Resolución 704 de fecha 31 de diciembre de 1997 al Grupo de Producción Porcina (GRUPOR) ambas del Ministerio de la Agricultura, está ubicada en Carretera del Jíbaro Km #1 en la ciudad cabecera provincial.

El objeto social de la empresa fue aprobado por la *Resolución 232* de fecha 24 de enero de 2005 del Ministerio de Economía y Planificación, adaptado a nuestra U.E.B. es el siguiente:

- ❖ Comercializar de forma mayorista carne de cerdo en pie y en bandas, cortes especiales, cerdos para asar, mamones, subproductos y procesados, en moneda nacional.
- ❖ Comprar y comercializar de forma mayorista y minorista cerdo en pie y carne de cerdo en banda mediante vínculos contractuales con los productores estatales y privados tenedores legales de tierras, pudiendo además comprar animales de las especies porcinas, a personas naturales criadores de la misma y que no sean intermediarios, cumpliendo las indicaciones establecidas por el Ministerio de la Agricultura, en moneda nacional.

La **misión** de esta empresa es producir y acopiar carne de cerdo para los destinos planificados y contribuir al ahorro de importaciones y la alimentación del pueblo.

La **visión** está centrada a ser una empresa perfeccionada, eficiente en su Gestión Económica, sostenible en sus producciones, con trabajadores y dirigentes motivados y una calidad elevada.

Esta fábrica se encuentra en búsqueda de alternativas, que le permita un mejor manejo de la información relacionada con el proceso productivo. Esto reduciría los riesgos en la administración de la empresa, como son la toma de decisiones apresuradas, tardías o inconsistentes de los directivos. Además, garantizaría un mejor control y análisis del comportamiento de la fábrica.

En nuestra U.E.B. los cerdos transitan por diferentes lugares, desde la formación de los mismos hasta el destino final, que es la venta en pie a las Empresas Cárnicas ó la venta en bandas a los organismos.

1.3.1 Comercialización de los cerdos.

La venta de los cerdos se realiza en horarios tempranos del día y a partir de las 5.00 PM por lotes de cerdos estabulados o recepcionados, donde participan el pesador, un representante de la Empresa Cárnica y un representante de nuestra U.E.B. Los mismos son trasladados desde los centros de recepción hasta los mataderos de la Empresa Cárnica en carros de nuestra empresa y, en algunos casos, en carros alquilados por las UTPNE para el traslado hacia los centros de recepción.

Cada carro de cerdos vendido va acompañado de un conduce emitido por un representante de nuestra U.E.B. para justificar el traslado de los cerdos.

La facturación de las ventas de cerdos en pie a la Empresa Cárnica se elabora por parte de un representante de nuestra U.E.B. con la participación de un representante de dicha empresa, la cual se deja constancia en el área económica de nuestra U.E.B. con copia a la empresa receptora, las cuales se costean según los rangos de peso de acuerdo al precio facturado.

Los cerdos que llegan al centro de recepción lesionados son pasados mediante transferencia a la losa de sacrificio del matadero de nuestra U.E.B., para posteriormente ser vendidos en bandas a organismos como Comercio, Gastronomía y Acopios. En cada municipio existe una losa sanitaria destinada a este efecto y para la venta de carne de producción de traspatio. Las partes lesionadas de los cerdos que no son aprovechables constituyen decomisos en la matanza y no forman parte del rendimiento en banda de los cerdos.

1.3.2 Deficiencias en el proceso de gestión de los datos de las ventas

La Empresa Provincial Porcina de la provincia Sancti Spíritus no cuenta actualmente con un software que permita el almacenamiento de estos datos referentes a las ventas de la masa porcina.

Con el fin de tener una herramienta que permita registrar y calcular, entre otros factores, los índices de ventas realizadas en el centro la Empresa Provincial Porcina en la provincia llevaron a cabo el diseño y realización de una Hoja de Cálculo en Excel. Esta hoja de cálculo se diseñó con la metodología que utilizan muchas fábricas convencionales en el país. Dicha hoja de cálculo puede ser fácilmente adaptada a las condiciones de las fábricas cubanas actuales. Sin embargo, no es del todo eficiente ya que no brinda la posibilidad a los directivos de interactuar de manera directa con la información, no posee el nivel de seguridad requerido para acceder a sus datos y la búsqueda de la información es complicada, lo que provoca que la utilidad de esta herramienta dependa, en gran medida, de la habilidad, la destreza y la experiencia del técnico que la manipule.

En la Empresa Provincial Porcina, la hoja de cálculo en Excel, a pesar de sus limitaciones, reúne datos de interés para los directivos. Es por ello que esta herramienta fue tomada como guía para el desarrollo del nuevo software. Teniendo en cuenta de que Cuba es un

país subdesarrollado y bloqueado económicamente, la creación del sistema propuesto estuvo muy apegada a los preceptos del Software Libre.

Aprovechando todas las ventajas que este ofrece sobre todo con respecto al pago de las licencias y a la libertad de los usuarios de poder ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Esto permitirá la factibilidad y la continuidad de un proyecto de este tipo, así como su extensión a otras empresas relacionadas con la rama de la producción de carne porcina.

Con el sistema propuesto se pretende perfeccionar la gestión de la información relacionada al proceso productivo de la Empresa Provincial Porcina, Sancti Spíritus. El sistema en general permitirá al Especialista tener acceso de manera instantánea a todos aquellos datos relacionados al proceso productivo de la empresa. La gestión de la información será más cómoda, confiable y segura. Se podrá realizar un análisis más específico del estado de las producciones de la masa porcina en la empresa.

1.4 Metodologías utilizadas para el diseño

Con el fin de lograr un mejor desempeño y calidad para desarrollar un producto informático, se hace necesario la consulta de las nuevas tendencias y del uso de una metodología que indique los pasos correctos a seguir. Además, el desarrollo del software propuesto, exige de antemano una búsqueda, estudio y análisis de diversas tecnologías, lenguajes, gestores de base de datos y herramientas de desarrollo que hagan posible su adecuada realización y que garantice el cumplimiento de todas sus funcionalidades.

1.4.1 Metodología RUP

El Proceso Unificado de Rational, (Rational Unified Process, de ahí las siglas RUP), es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos (Rumbaugh, Booch, & Jacobson, 2006).

Es un proceso basado en componentes y utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. No obstante, los verdaderos aspectos definitorios de RUP se resumen en tres frases clave: está dirigido por casos de uso, centrado en la arquitectura y es iterativo e incremental (Rumbaugh, Booch, & Jacobson, 2006).

1.4.2 Lenguaje Unificado de Modelado (UML)

El Lenguaje Unificado de Modelado (Unified Modeling Language, UML) es un lenguaje estándar para escribir planos de software, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. Puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software (Rumbaugh, Booch, & Jacobson, 2006).

1.5 Lenguaje de programación Java

Java es un lenguaje de desarrollo de propósito general, y como tal es válido para realizar todo tipo de aplicaciones profesionales. Incluye una combinación de características que lo hacen único y está siendo adoptado por multitud de fabricantes como herramienta básica para el desarrollo de aplicaciones comerciales de gran repercusión (Schildt, 2005).

Una de las características más importantes es que los programas “ejecutables”, creados por el compilador de Java, son independientes de la arquitectura. Se ejecutan indistintamente en una gran variedad de equipos con diferentes microprocesadores y sistemas operativos. Pero además Java se caracteriza por (Schildt, 2005):

- ❖ Ser un lenguaje intrínsecamente orientado a objetos.
- ❖ Funcionar perfectamente en red.
- ❖ Aprovechar características de la mayoría de los lenguajes modernos evitando sus inconvenientes. En particular los del C++.
- ❖ Tener una gran funcionalidad gracias a sus librerías (clases).
- ❖ No tener punteros manejables por el programador, aunque los maneja interna y transparentemente.
- ❖ El manejo de la memoria no es un problema, la gestiona el propio lenguaje y no el programador.
- ❖ Generar aplicaciones con pocos errores posibles.

Muchos fabricantes de software están migrando sus aplicaciones a este nuevo lenguaje. Posiblemente debido a que valoran más las ventajas y, fundamentalmente la posibilidad de desarrollar una sola aplicación para múltiples plataformas así como la mayor facilidad que ofrece Java para el mantenimiento posterior de las aplicaciones (Schildt, 2005).

Java es uno de los lenguajes de programación más utilizados en el desarrollo de aplicaciones de código abierto y posee una variada bibliografía para aquellos programadores interesados en el aprendizaje de este lenguaje. Por todas estas razones fue tomado en consideración el lenguaje de programación Java para la realización de este proyecto. Existen varias herramientas de desarrollo que utilizan el lenguaje Java para la realización de diferentes aplicaciones.

1.6 Lenguaje de consulta estructurado SQL

SQL (Structured Query Language) o Lenguaje de Consulta Estructurado es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Posibilita lanzar consultas con el fin de recuperar información de interés de una base de datos de una forma sencilla.

SQL permite la concesión y denegación de permisos, la implementación de restricciones de integridad y controles de transacción, y la alteración de esquemas. Debido a que es un lenguaje declarativo, especifica qué es lo que se quiere y no cómo conseguirlo, por lo que una sentencia no establece explícitamente un orden de ejecución.

1.7 Lenguaje de consulta estructurado PL/pgSQL

PostgreSQL posee varios lenguajes para programar las funciones. Uno de estos lenguajes es el PL/pgSQL, muy parecido al PL/SQL de Oracle (The PostgreSQL Global Development Group, 2003).

Según The PostgreSQL Global Development Group, 2003 los objetivos de diseño para PL/pgSQL fueron crear un lenguaje procedural cargable para:

- ❖ Crear funciones y triggers,
- ❖ Añadir estructuras de control al lenguaje SQL,
- ❖ Realizar computaciones complejas,
- ❖ Heredar todos los tipos definidos por el usuario, funciones y operadores,
- ❖ Ser definido para ser validado por el servidor,
- ❖ Ser sencillo de utilizar.

1.8 Arquitectura cliente-servidor

El término cliente-servidor se refiere principalmente a una arquitectura o división lógica de responsabilidades. El cliente es la aplicación (conocida también como interfaz o parte frontal) y el servidor es el DBMS (conocido también como servidor de fondo o parte dorsal). Como claramente el sistema puede ser dividido en dos partes, es posible ejecutar a las dos partes en máquinas diferentes. Debido a que esta posibilidad es tan atractiva, el término cliente-servidor ha venido aplicándose casi exclusivamente para el caso donde el cliente y el servidor están en máquinas diferentes (Date, 2001).

Existen varias posibilidades de topología para esta arquitectura (Connolly & Begg, 2005; Date, 2001):

- ❖ Varios clientes y un servidor
- ❖ Un cliente puede acceder a varios servidores:

 - ❖ El cliente accede a un servidor a la vez
 - ❖ El cliente accede a varios servidores simultáneamente; es decir, una solicitud puede involucrar datos de varios servidores y el cliente lo ve como si fuera una sola unidad (sistema de base de datos distribuida).

1.8.1 Arquitectura cliente-servidor en 2 capas

Según (Connolly & Begg, 2005) las aplicaciones con requerimientos de datos están formadas básicamente por cuatro componentes: la base de datos, la lógica de transacciones, la lógica del negocio, y la gestión de datos y la interfaz de usuario. La arquitectura cliente-servidor con dos capas (conocidas también como niveles, o por los términos del inglés *level*, *tier layer*), proporciona una separación muy básica para estos componentes. El cliente es el responsable de la presentación de los datos al usuario, mientras que el servidor es quien los provee al cliente (ver Figura 1.1).

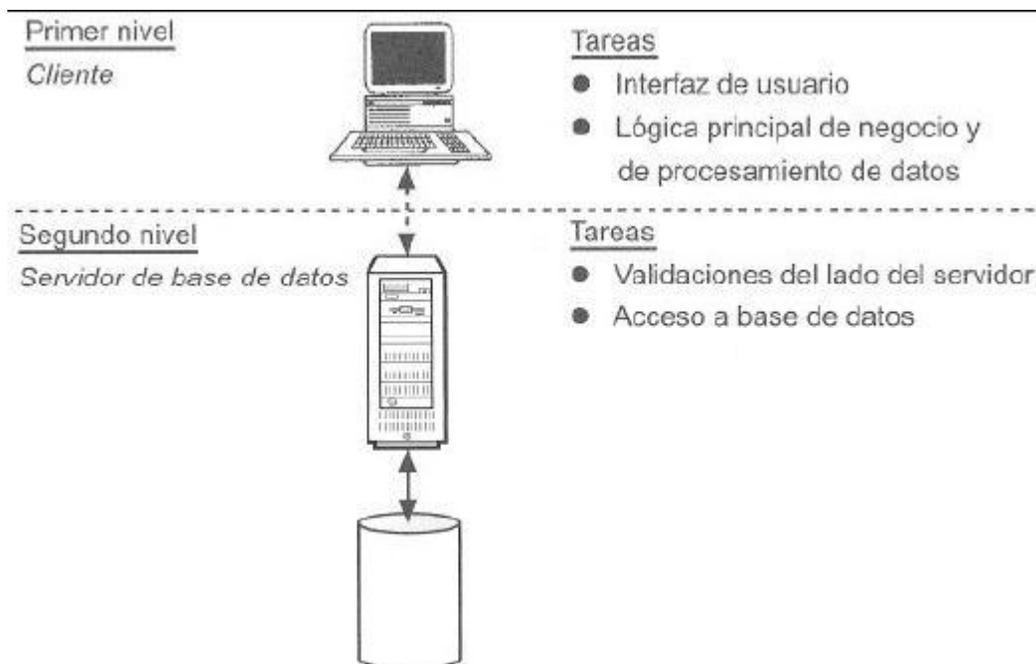


Figura #1.1: La arquitectura cliente-servidor con dos capas

En una interacción típica entre el cliente y el servidor, el cliente acepta la solicitud del usuario, comprueba la sintaxis y genera solicitudes de base de datos en SQL o en otro lenguaje que resulta apropiado para la lógica de la aplicación. Luego transmite el mensaje al servidor, espera la respuesta y formatea los resultados para el usuario final. El servidor acepta y procesa las solicitudes de base de datos y finalmente transmite los resultados al cliente.

A los clientes de la arquitectura con dos capas, también se les conoce como clientes pesados (del inglés *fatclient*) (Microsoft SQL Server, 2004; Reese, 2000) o complejos (Connolly & Begg, 2005), pues contienen la interfaz de usuario y la lógica empresarial (reglas del negocio) de la aplicación (Connolly & Begg, 2005; Microsoft SQL Server, 2004; Silberschatz, Korth, & Sudarshan, 2002). Los clientes pesados, por lo general, se conectan a través de una API (Application Program Interface) con el DBMS (ver Figura 1.2) que se encuentra en la máquina servidor (Elmasri & Navathe, 2004). Las API más ampliamente difundidas son ODBC para sistemas Windows, y JDBC de la plataforma Java (Fisher, Ellis, & Bruce, 2003).

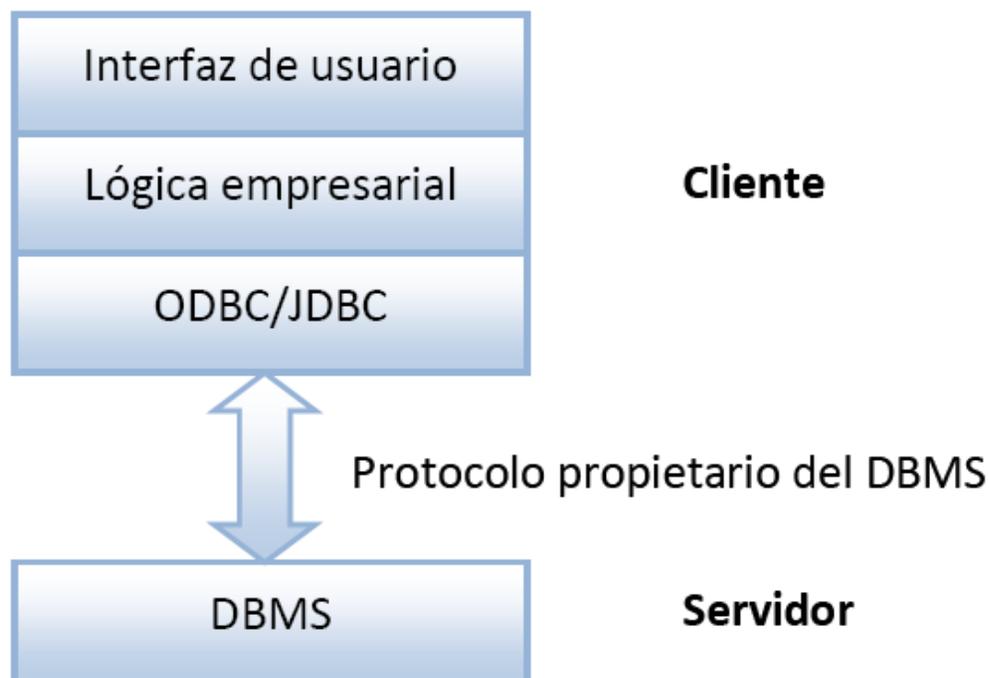


Figura #1.2: Conexión con un DBMS en la arquitectura cliente-servidor con dos capas.

Para (Connolly & Begg, 2005) las dos capas presentan las siguientes ventajas:

- ❖ Permite un acceso más universal a las bases de datos existentes.
- ❖ Mejores prestaciones: si los clientes y el servidor residen en computadoras distintas, las diferentes CPU podrán procesar distintas aplicaciones en paralelo. También resulta más fácil optimizar la máquina servidora si su única tarea consiste en realizar el procesamiento relacionado con la base de datos.
- ❖ Pueden reducirse los costos de hardware: sólo el servidor requiere el suficiente espacio de almacenamiento y la suficiente capacidad de proceso como para almacenar y gestionar la base de datos.
- ❖ Se reducen los costes de comunicaciones: las aplicaciones llevan a cabo parte de las operaciones en el cliente y sólo envían a través de la red las solicitudes de acceso a la base de datos, con lo que se envían menos datos a través de la red.
- ❖ Mayor coherencia: el servidor puede gestionar las comprobaciones de identidad, por lo que sólo es necesario definir y validar las restricciones en un único lugar, en lugar de hacer que cada programa de aplicación lleve a cabo sus propias comprobaciones.

1.9 Patrón de diseño Modelo-Vista-Controlador.

En el diseño de aplicaciones con sofisticadas interfaces se utiliza el patrón de diseño Modelo-Vista-Controlador (MVC). Pues la lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica del negocio. Por lo que si se realiza un diseño que mezcle los componentes de interfaz y de negocio, entonces la consecuencia será que, cuando se necesite cambiar la interfaz, se tendrán que modificar los componentes de negocio, siendo esto trabajoso y engorroso, además de provocar posibles errores en el sistema. Por lo que se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio (Lago, 2007).

Elementos que componen al patrón:

- ❖ **Modelo:** datos que serán mostrados por la vista.
- ❖ **Vista:** muestra la información del modelo al usuario.
- ❖ **Controlador:** gestiona las entradas del usuario y determina qué debe hacer el modelo o la vista en respuesta a las acciones del usuario.

1.10 Sistemas Gestores de Bases de Datos.

Una base de datos (BD) es un conjunto de datos interrelacionados entre sí, almacenados con carácter más o menos permanente en la computadora. O sea, que una BD puede considerarse una colección de datos variables en el tiempo (Mato García, 2006).

El software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez, se denomina sistema de gestión de bases de datos (SGBD) (Mato García, 2006).

El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado (Mato García, 2006).

Existen muchos SGBD entre los que se pueden mencionar fundamentalmente tales como el PostgreSQL, MySQL entre otros.

1.10.1 MySQL

MySQL surgió alrededor de la década del 90, creada por la empresa sueca MySQL AB. MySQL es un gestor de base de datos sencillo de usar y increíblemente rápido. También es uno de los motores de base de datos más usados en Internet, la principal razón de esto es que es gratis para aplicaciones no comerciales (Riveros, 2008).

Es una base de datos muy rápida, segura y fácil de usar, por eso es una de las bases de datos más usadas en Internet.

1.10.2 PostgreSQL

PostgreSQL es un SGBD relacional orientado a objetos y *libre*, publicado bajo la licencia BSD (Riveros, 2008). PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre (Riveros, 2008).

PostgreSQL posee una serie de características que fueron la que dieron al traste para su utilización como SGBD en este proyecto (Riveros, 2008).

Alta concurrencia: Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*. Esta estrategia es superior al uso de bloqueos por tabla o por *filas* común en otras bases, eliminando la necesidad del uso de bloqueos explícitos (Riveros, 2008).

Amplia variedad de tipos nativos: PostgreSQL provee nativamente soporte para:

- ❖ Números de precisión arbitraria.
- ❖ Texto de largo ilimitado.
- ❖ Figuras geométricas (con una variedad de funciones asociadas)
- ❖ Direcciones IP (IPv4 e IPv6).
- ❖ Bloques de direcciones estilo CIDR.
- ❖ Direcciones MAC.
- ❖ Arrays.

Adicionalmente los usuarios pueden crear sus propios tipos de datos.

Funciones

Bloques de código que se ejecutan en el servidor. Pueden ser escritos en varios lenguajes, con la potencia que cada uno de ellos da, desde las operaciones básicas de programación, tales como bifurcaciones y bucles, hasta las complejidades de la programación orientada a objetos o la programación funcional.

Los disparadores (*triggers* en inglés) son funciones enlazadas a operaciones sobre los datos. Algunos de los lenguajes que se pueden usar son los siguientes:

- ❖ Un lenguaje propio llamado PL/PgSQL (similar al PL/SQL de Oracle).C.
- ❖ C++.
- ❖ Java PL/Java web.
- ❖ PL/Perl.plPHP.
- ❖ PL/Python.
- ❖ PL/Ruby.
- ❖ PL/sh.
- ❖ PL/Tcl.
- ❖ PL/Scheme.

Lenguaje para aplicaciones estadísticas R por medio de PL/R. PostgreSQL soporta funciones que retornan "filas", donde la salida puede tratarse como un conjunto de valores que pueden ser tratados igual a una fila retornada por una consulta (query en inglés).

Este lenguaje brinda una serie de ventajas entre ellas:

- ❖ Mayor rendimiento: SQL es el lenguaje que PostgreSQL (y la mayoría del resto de bases de datos relacionales) usa como lenguaje de consultas. Es portable y fácil de aprender. Pero cada estamento SQL debe ser ejecutado individualmente por el servidor de bases de datos. Esto significa que su aplicación cliente debe enviar cada consulta al servidor de bases de datos, esperar a que se procese, recibir el resultado, realizar alguna computación, y luego enviar otras consultas al servidor. Todo esto incurre en una comunicación entre procesos y también puede sobrecargar la red si su cliente se encuentra en una máquina distinta al servidor de bases de datos (The PostgreSQL Global Development Group, 2003).

- ❖ Soporte SQL:PL/pgSQL añade a la potencia de un lenguaje procedural la flexibilidad y sencillez del SQL. Con PL/pgSQL puede usar todos los tipos de datos, columnas, operadores y funciones de SQL (The PostgreSQL Global Development Group, 2003).
- ❖ Portabilidad: debido a que las funciones PL/pgSQL corren dentro de PostgreSQL, estas funciones funcionarán en cualquier plataforma donde PostgreSQL corra. Así podrá reutilizar el código y reducir costes de desarrollo (The PostgreSQL Global Development Group, 2003). Por las características mostradas anteriormente se seleccionó PostgreSQL como SGBD en este proyecto.

Se escoge la herramienta PostgreSQL debido a que resulta suficientemente adecuada para el desarrollo de una aplicación de escritorio para facilitar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus. Además presenta características deseables como que es un software libre, buen rendimiento, portabilidad y alta concurrencia.

1.11 NetBeans 6.8

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos (Netbeans, 2009).

La Plataforma NetBeans es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones (Netbeans, 2009). La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación.

Entre las características de la plataforma están:

- ❖ Administración de las interfaces de usuario (ej. menús y barras de herramientas)
- ❖ Administración de las configuraciones del usuario
- ❖ Administración del almacenamiento (guardando y cargando cualquier tipo de dato)

- ❖ Administración de ventanas
- ❖ Framework basado en asistentes (diálogos paso a paso).

El IDE NetBeans es un producto libre y gratuito sin restricciones de uso y de amplias comodidades para el programador (Netbeans, 2009).

Para la realización de este proyecto se utilizó el *IDE 6.8 de NetBeans*, una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación.

1.12 Rational Rose

Para facilitar la documentación de la presente investigación, se emplea la herramienta de ingeniería de software asistida por computadora, CASE (Computer Assisted Software Engineering), Rational Rose Enterprise. Potente aplicación que integra todos los elementos de la metodología RUP y la notación UML, para cubrir el análisis, diseño, implementación y prueba del ciclo de vida básico de un sistema informático.

Permitiendo realizar la generación de diagramas y códigos, la automatización de sistemas, y la ingeniería inversa.

La metodología RUP es la que mejor se ajusta a las necesidades que existen actualmente en el desarrollo de software, pues propone un modelo iterativo e incremental, muy acorde con la naturaleza cambiante de los requisitos en muchos proyectos. Después del análisis realizado se decidió, utilizar la metodología RUP para llevar a cabo paso a paso el proceso de desarrollo del software propuesto. Esto responde fundamentalmente a que esta metodología se ha convertido en un estándar internacional para guiar el proceso de desarrollo de software, al igual que en nuestro país.

1.13 Conclusiones parciales

En este capítulo se analizaron las características de las distintas metodologías y herramientas de desarrollo, así como las ventajas de las mismas y se determinó seleccionar:

- ❖ RUP (Proceso Unificado de Desarrollo) como la metodología más apropiada para el desarrollo del proyecto.
- ❖ UML como el lenguaje de modelación utilizado.
- ❖ La arquitectura 2 Capas.
- ❖ Para la implementación el lenguaje Java, utilizando como herramienta de programación NetBeans IDE 6.8.
- ❖ PostgreSQL como sistema gestor de bases de datos.
- ❖ Rational Rose como herramienta de modelado UML.

La selección de estas tecnologías se basó principalmente en la tecnología código libre y la característica de integrarse para funcionar en cualquier plataforma.

Capítulo II Descripción de la aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus

En este capítulo se describe de modo general el funcionamiento del sistema diseñado. Para lograr una mejor comprensión del problema a resolver y el común entendimiento entre clientes y desarrolladores se realizará la modelación del negocio. Para ello se exponen las reglas del negocio a tener en cuenta para el diseño, una caracterización de los actores y trabajadores, así como sus casos de uso, conformándose el diagrama de casos de uso, la descripción de estos en formato expandido, y se muestran los diagramas de actividad y del modelo de objetos. Además se definen los requerimientos funcionales y no funcionales.

2.1 Descripción del modelo de negocio

El modelado del negocio es una técnica para comprender los procesos del negocio de la organización (Burbeck, 2007).

2.1.1 Descripción de los procesos de negocio

Los procesos de negocio son el conjunto total de actividades necesarias para producir un resultado de valor percibido y medible para un cliente individual de un negocio (Rumbaugh, Booch, & Jacobson, 2006).

Partiendo del criterio anterior se identificó como proceso del negocio: El proceso de la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus.

2.2 Modelo de casos de uso del negocio

El modelo de Casos de Uso del Negocio (CUN) describe los procesos de un centro 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.

Este modelo es definido a través de tres elementos: el diagrama de casos de uso del negocio, la descripción de los casos de uso del negocio y el diagrama de actividades.

2.2.1 Actores del negocio

Según (Rumbaugh, Booch, & Jacobson, 2006) un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa.

Actor del Negocio	Justificación
Especialista	Interesado en recibir información estadística sobre las ventas de los cerdos en la empresa.

Tabla 2.1. Actores del negocio

2.2.2 Trabajadores del negocio

“Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol” (Rumbaugh, Booch, & Jacobson, 2006).

Trabajador	Justificación
Técnico	Persona que es la encargada de recibir información estadística de cada venta que se realice e introducirla en cada planilla de constancia.

Tabla 2.2. Trabajadores del negocio

2.2.3 Diagrama de casos de uso del negocio

El modelo de casos de uso del negocio es un modelo que describe los procesos de negocio 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 del negocio presenta un sistema (en este caso, el negocio) desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios (Rumbaugh, Booch, & Jacobson, 2006).

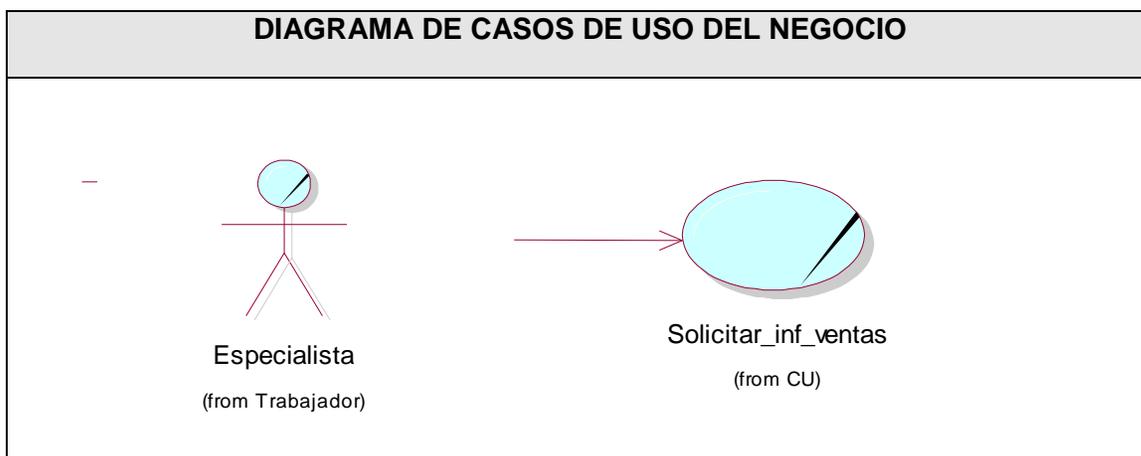


Figura #2.1. Diagrama de casos de uso del negocio.

2.2.4 Descripción de los casos de uso del negocio

CU 1	Solicitar información de las ventas de cerdos realizadas.	
Actores	Especialista	
Propósito	Conocer de las ventas de cerdos realizadas.	
Resumen		
El caso de uso inicia cuando el especialista llega y solicita la información sobre las ventas de cerdos al técnico y culmina cuando el técnico le da el resultado de la búsqueda.		
Acción del actor		
Respuesta del proceso del Negocio		
1- El especialista se dirige al técnico para solicitar información de las ventas de cerdos que se han hecho en un mes y año especificado.		2- El técnico realiza una búsqueda en las facturas realizadas en ese mes y año. En caso de no encontrarla ir a <u>Subsección 1</u> 3-En caso de encontrar la información el técnico confecciona una planilla con los resultados de la búsqueda.
4-El especialista recibe la información solicitada.		
Prioridad		Alta.
Mejoras		El Especialista podrá adquirir la información en forma automatizada de manera que las respuestas a sus solicitudes serán ágiles.
Otras secciones		
Subsección 1		5-El técnico no encuentra la información y se lo comunica al especialista.
6-El especialista recibe la información.		

Tabla 2.3 Descripción del CU "Solicitar información de las ventas de cerdos realizadas".

2.2.5 Diagrama de Actividad

El diagrama de actividad es un grafo que contiene los estados en que puede hallarse la actividad a analizar. Cada estado de la actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En resumen describe un proceso que explora el orden de las actividades que logran los objetivos del negocio (Rumbaugh, Booch, & Jacobson, 2006).

En el Anexo II.17 se muestra dicho diagrama.

2.3 Diagrama de clases del modelo de objetos

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 (Rumbaugh, Booch, & Jacobson, 2006).

El diagrama de clases del modelo de objeto, es un artefacto que se construye para describir el modelo de objetos del negocio. Una entidad del negocio representa algo, que los trabajadores toman, inspeccionan, manipulan, producen o utilizan en un caso de uso del negocio.

En el Anexo II.16 se muestra dicho diagrama.

2.4 Requerimientos funcionales

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo (Rumbaugh, Booch, & Jacobson, 2006). Todo esto basándose en las necesidades de los usuarios y clientes.

Los requerimientos funcionales del sistema propuesto son los siguientes:

R1- Autenticar usuario.

R2- Cambiar contraseña.

R3- Gestionar usuario.

3.1- Añadir usuario.

3.2- Modificar usuario.

3.3- Eliminar usuario.

3.4- Buscar usuario según su nombre.

R4- Gestionar vendedor.

4.1- Añadir vendedor.

4.2- Modificar vendedor.

4.3- Eliminar vendedor.

4.4- Buscar vendedor según su código.

R5- Gestionar comprador.

5.1- Añadir comprador.

5.2- Modificar comprador.

5.3- Eliminar comprador.

5.4- Buscar comprador según su código.

R6- Gestionar venta por matadero.

6.1- Añadir venta por matadero.

6.2- Eliminar venta por matadero.

6.3- Modificar venta por matadero.

6.4- Buscar venta por matadero por número de factura.

R7- Gestionar venta por municipio.

7.1- Añadir venta por municipio.

7.2- Eliminar venta por municipio.

7.3- Modificar venta por municipio.

7.4- Buscar venta por municipio por número de factura.

R8- Gestionar venta por nevera.

8.1- Añadir venta por nevera.

8.2- Eliminar venta por nevera.

8.3- Modificar venta por nevera.

8.4- Buscar venta por nevera por número de factura.

R9- Mostrar las ventas realizadas en el Municipio.

9.1- Mostrar las ventas realizadas por cada municipio a Comercio y Gastronomía en un mes y año determinado.

9.2- Mostrar las ventas realizadas por cada municipio a Mercado Agropecuario en un mes y año determinado.

R10- Mostrar las ventas realizadas en la Nevera.

10.1- Mostrar las ventas realizadas en la nevera hacia MINAL en un mes y año determinado.

- 10.2- Mostrar las ventas realizadas en la nevera hacia MINAL-DIVISA en un mes y año determinado.
- 10.3- Mostrar las ventas realizadas en la nevera hacia Comedor en un mes y año determinado.
- 10.4- Mostrar las ventas realizadas en la nevera hacia Venta a trabajadores en un mes y año determinado.
- 10.5- Mostrar las ventas realizadas en la nevera hacia Albergue y Casa de Tránsito en un mes y año determinado.
- 10.6- Mostrar las ventas realizadas en la nevera hacia Comercio y Gastronomía en un mes y año determinado.
- 10.7- Mostrar las ventas realizadas en la nevera hacia Mercado Agropecuario en un mes y año determinado.
- 10.8- Mostrar las ventas realizadas en la nevera hacia Aseguramiento MINAGRI en un mes y año determinado.
- 10.9- Mostrar las ventas realizadas en la nevera hacia Otros destinos en un mes y año determinado.

R11- Mostrar las ventas realizadas en el Matadero.

- 11.1- Mostrar las ventas realizadas en las unidades especializadas hacia MINAL desde el Matadero en un mes y año determinado.
- 11.2- Mostrar las ventas realizadas en las unidades especializadas hacia MINAL-DIVISA desde el Matadero en un mes y año determinado.
- 11.3- Mostrar las ventas realizadas en las unidades especializadas hacia Comedor desde el Matadero en un mes y año determinado.
- 11.4- Mostrar las ventas realizadas en las unidades especializadas hacia Venta a trabajadores desde el Matadero en un mes y año determinado.
- 11.5- Mostrar las ventas realizadas en las unidades especializadas hacia Albergue y Casa de Tránsito desde el Matadero en un mes y año determinado.
- 11.6- Mostrar las ventas realizadas en las unidades especializadas hacia Comercio y Gastronomía desde el Matadero en un mes y año determinado.
- 11.7- Mostrar las ventas realizadas en las unidades especializadas hacia Mercado Agropecuario desde el Matadero en un mes y año determinado.
- 11.8- Mostrar las ventas realizadas en las unidades especializadas hacia Aseguramiento MINAGRI desde el Matadero en un mes y año determinado.

11.9- Mostrar las ventas realizadas en las unidades especializadas hacia Otros destinos desde el Matadero en un mes y año determinado.

R12- Mostrar las ventas realizadas al MINAL-DIVISA en un mes y año determinado.

R13- Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento.

R14- Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio.

14.1- Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio por los Vendedores Especializados y sus Compradores hasta el momento.

14.2- Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio por los Vendedores Especializados hasta el momento.

R15- Mostrar comparación del mayor importe de ventas realizadas.

15.1- Mostrar comparación del mayor importe de ventas según los Vendedores Especializados y sus Compradores.

15.2- Mostrar comparación del mayor importe de ventas según los Vendedores Especializados.

2.5 Requisitos no funcionales

Los requerimientos no funcionales especifican cualidades, propiedades del sistema; como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, etc (Rumbaugh, Booch, & Jacobson, 2006).

Los requerimientos no funcionales del sistema propuesto son los siguientes:

Requerimientos de Apariencia o Interfaz Externa

- ❖ La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.
- ❖ La interfaz del módulo de administración del sistema debe ser personalizada de acuerdo al tipo de usuario que acceda a ella.

Requerimientos de Usabilidad

- ❖ El sistema es de libre acceso, pero a la aplicación diseñada para el mantenimiento del mismo solo tendrán acceso los usuarios definidos por la Administración del Departamento de Estadística y Planificación en la Empresa Provincial Porcina en Sancti Spíritus.
- ❖ Serán el informático, técnico y administrador de la aplicación pertenecientes a la Empresa Provincial Porcina en Sancti Spíritus quienes utilicen más este software.

Requerimientos de Rendimiento

La eficiencia del sistema estará determinada por el aprovechamiento de los recursos que se disponen en la arquitectura 3 capas, y la velocidad de las consultas en la Base de Datos.

Requerimientos de Soporte

- ❖ Se requiere un servidor de bases de datos con soporte para grandes volúmenes de información, velocidad de procesamiento y tiempo de respuesta rápido en accesos concurrentes. Se documentará la aplicación para garantizar su soporte.
- ❖ El administrador tendrá la responsabilidad de mantener actualizada la aplicación.
- ❖ Las pruebas del sistema se realizarán en la Empresa Provincial Porcina en Sancti Spíritus. Dichas pruebas permitirán evaluar en la práctica la funcionalidad y las ventajas de este nuevo producto.
- ❖ El sistema debe propiciar su mejoramiento y la incorporación de otras opciones.

Requerimientos de Portabilidad

La plataforma seleccionada para desarrollar la aplicación fue Windows, pero puede ser ejecutada desde otras plataformas que soporten el lenguaje Java y SQL.

Requerimientos de Seguridad

- ❖ Debe garantizar la conectividad e integridad de los datos almacenados a través de la red. Esto está garantizado por el Sistema Operativo.
- ❖ Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos.
- ❖ El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad.
- ❖ La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo con la política de seguridad del sistema.

Requerimientos de Ayuda y Documentación en Línea

- ❖ El sistema contará con una ayuda que explicará de manera clara y sencilla al usuario todas las funcionalidades del sistema.
- ❖ La ayuda quedará conformada por un menú general que le ofrecerá al usuario una forma fácil para navegar por toda la ayuda, mediante el menú el usuario podrá ir de un lugar a otro sin perderse.

Requerimientos de Software

- ❖ Se debe disponer de un sistema operativo compatible. El sistema debe poderse ejecutar en entornos Windows y/o Linux (Multiplataforma).
- ❖ Del lado del servidor se utilizará PostgreSQL 8,4 o superior como gestor de base de datos para garantizar la integridad y consistencia de la información.
- ❖ Del lado del cliente se utilizará Java (TM) SE Development Kit 6 Update 3 como máquina virtual, Java v.6 o superior como lenguaje de programación, PostgreSQL 8,4 o superior como gestor de base de datos para garantizar la integridad y consistencia de la información.

Requerimientos de Hardware

Para el desarrollo y puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

Se requiere de una máquina que funcione como servidor, la cual debe tener como mínimo:

- ❖ Un Procesador Pentium III con 1 GHz de frecuencia ó superior.
- ❖ 512 Mb de Memoria RAM.
- ❖ 40 GB de Disco Duro.

Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo:

- ❖ Un Procesador Pentium III.
- ❖ 128 Mb de Memoria RAM.
- ❖ Deben estar conectadas en red con el servidor a través de una tarjeta de red de 100 Mbps.

2.6 Descripción de la aplicación de escritorio propuesta

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 (Pressman, 2007).

Actor del Sistema	Justificación
Especialista	Persona que se encarga de brindar información estadística sobre los datos de las ventas realizadas en la institución.
Técnico	Persona que es la encargada de introducir al sistema la información estadística de cada venta.
Administrador	Es una persona que se encarga de crear los usuarios del sistema, así como de todo el proceso de soporte y mantenimiento del mismo.
Usuario	Actor genérico que hace uso de funcionalidades que son comunes a varios actores y por ende se crea para que aquellos hereden de él.

Tabla 2.4. Actores del sistema.

2.6.2 Modelo de casos de uso del sistema por paquetes

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 (Pressman, 2007).

Subdividir los casos de uso en paquetes resulta de mucha ayuda en la modelación de cualquier sistema informático. Los paquetes son un mecanismo de organización de elementos que subdividen el modelo en otros más pequeños que colaboran entre sí (Pressman, 2007).

La siguiente figura muestra las relaciones existentes entre los paquetes:

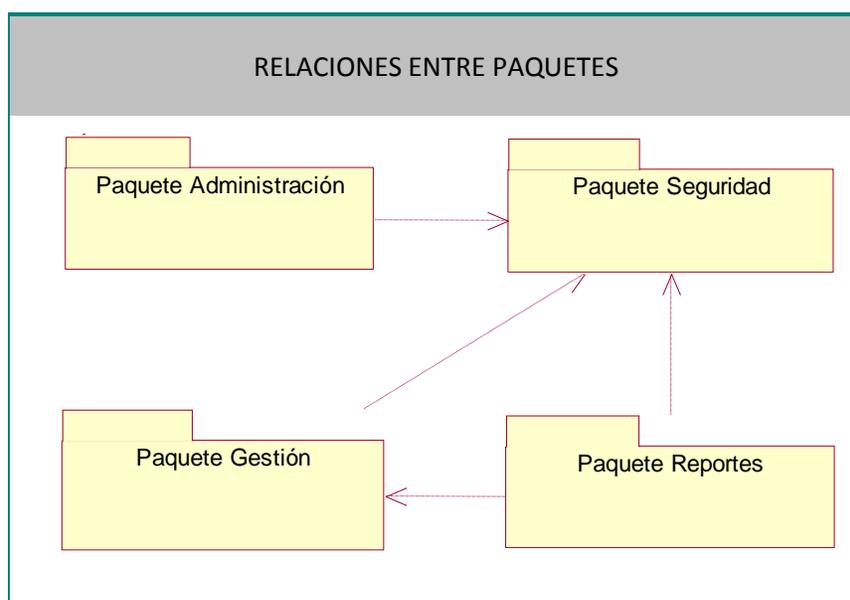


Figura # 2.2. Diagrama de caso de uso por paquetes.

Paquete Administración: En este paquete se agrupan las funcionalidades relacionadas con la gestión de los datos de los usuarios, compradores y unidades de procedencia.

Paquete Seguridad: En este paquete se agrupan las funcionalidades relacionadas con autenticar usuario y cambiar contraseña.

Paquete Gestión: En este paquete se agrupan las funcionalidades relacionadas con la actualización (altas, bajas y modificaciones) de los datos.

Paquete Reportes: En este paquete se agrupan las funcionalidades relacionadas con, la visualización de los datos, filtrados según diferentes criterios de búsqueda; la obtención de reportes.

2.6.3 Diagramas de Casos de Uso del Sistema

El modelo de casos de uso permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. Describe lo que hace el sistema para cada tipo de usuario y proporciona la entrada fundamental para el análisis, el diseño y las pruebas (Jacobson, Booch, & Rumbaugh, 2006).

“De acuerdo con el RUP los casos de uso se derivan del análisis de los requerimientos enriqueciendo requerimientos funcionales a los que se encuentran asociados” (Cáceres, 2005, p. 91).

El diagrama de casos de uso muestra la relación que existe entre estos y los actores del sistema. A continuación se muestran los casos de uso del sistema identificados por paquetes.

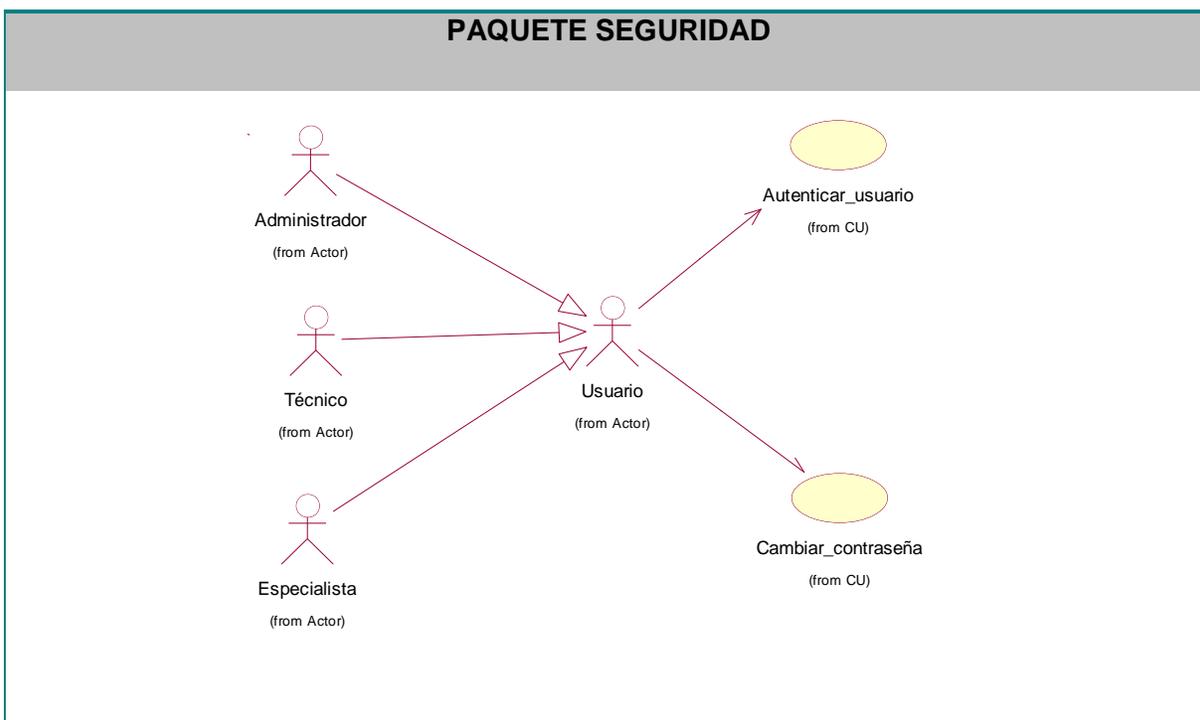


Figura # 2.3. Diagrama de casos de uso. Paquete Seguridad.

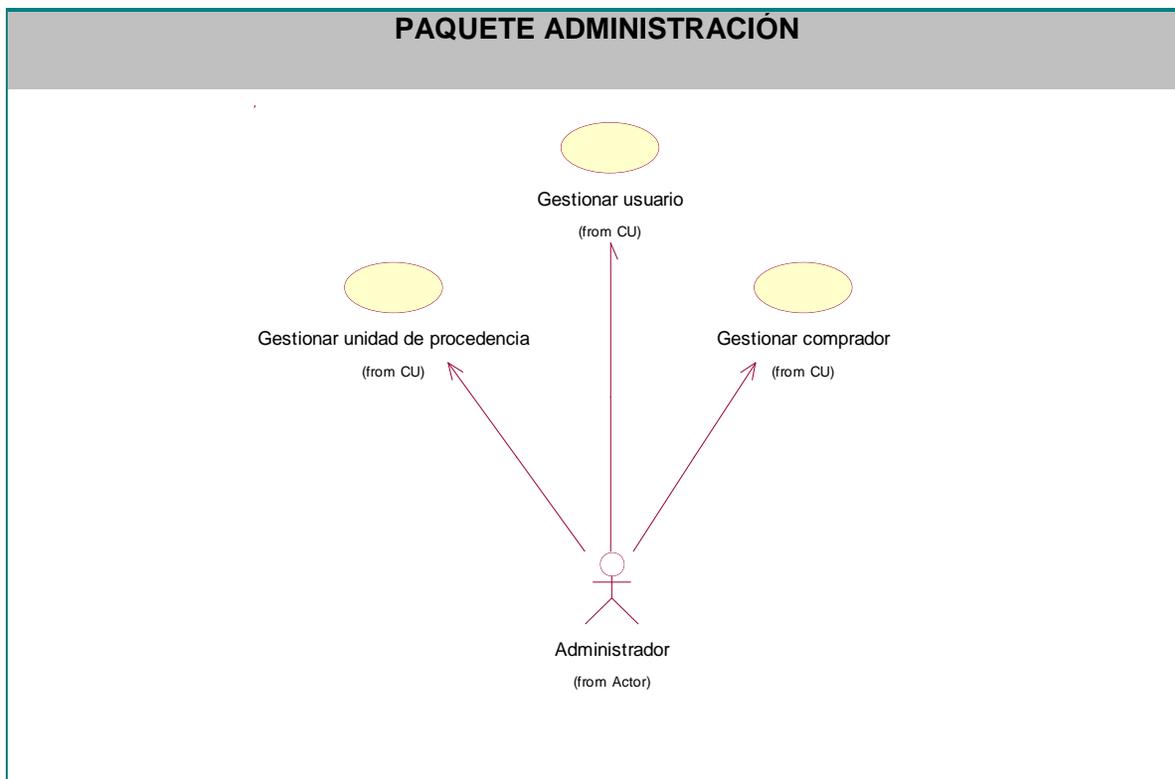


Figura # 2.4. Diagrama de casos de uso. Paquete Administración.

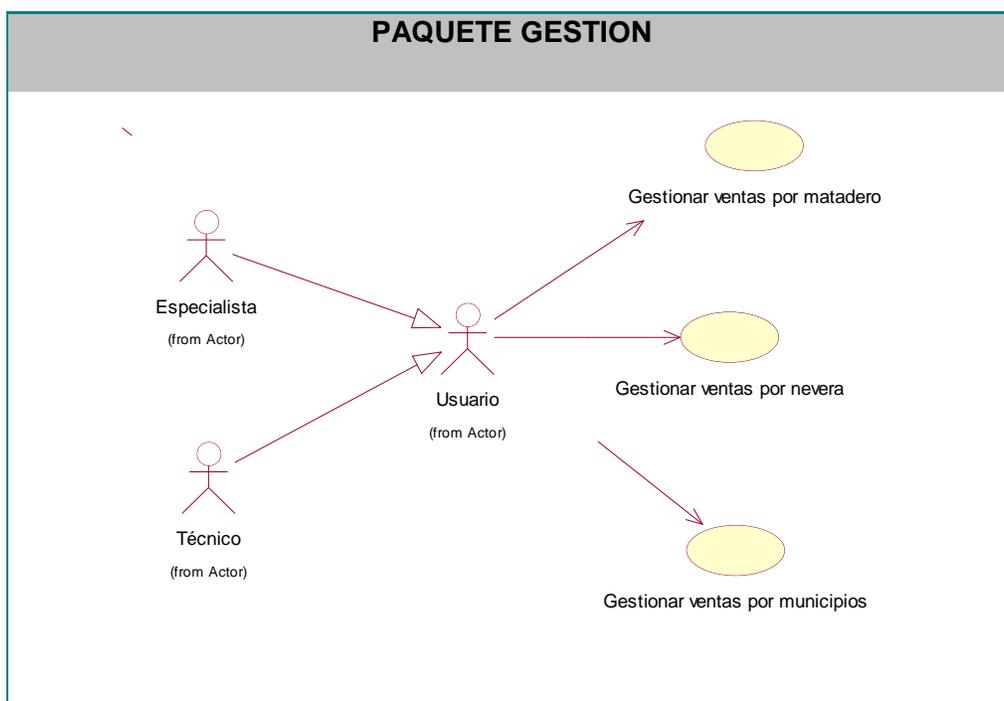


Figura # 2.5. Diagrama de casos de uso. Paquete Gestión.

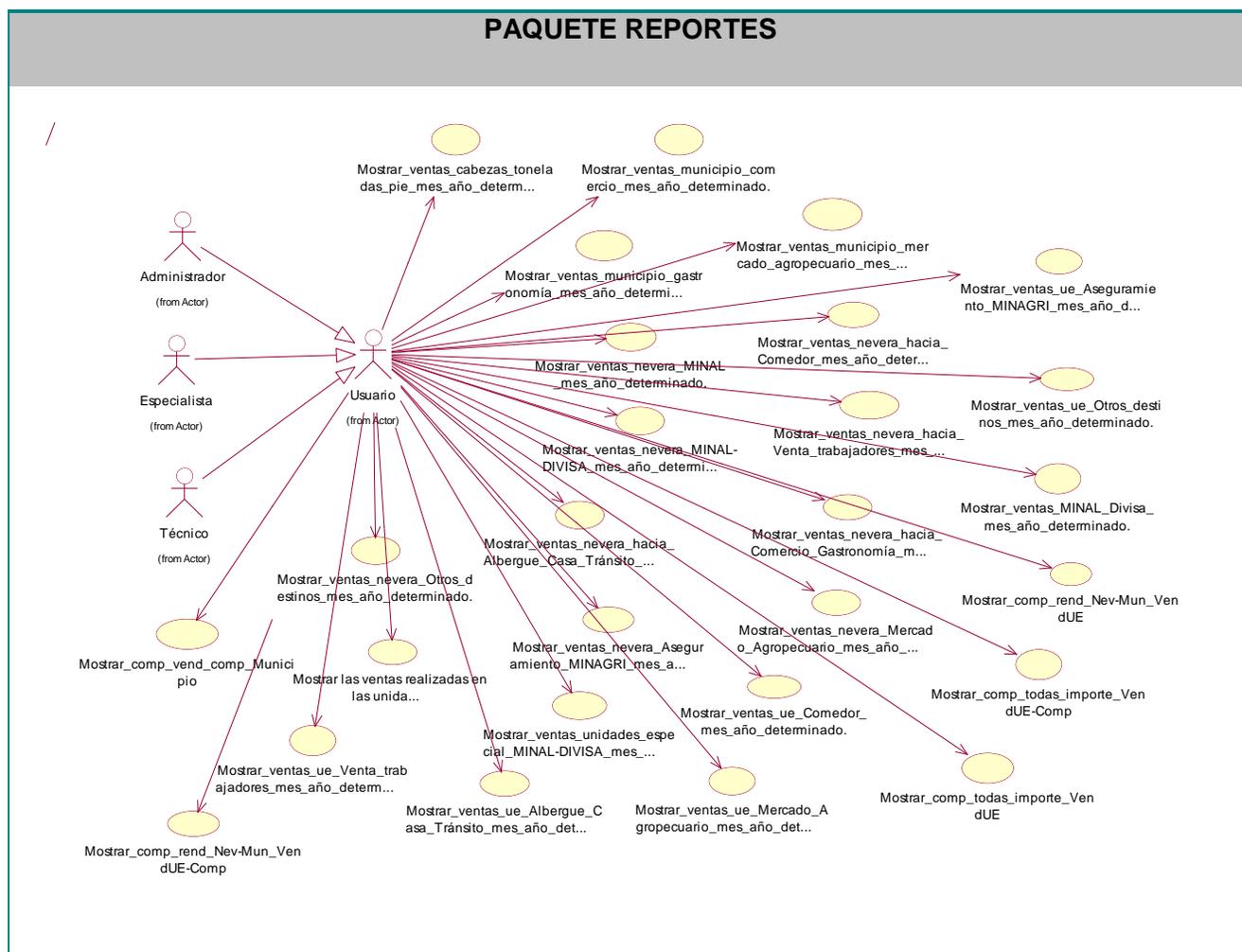


Figura # 2.6. Diagrama de casos de uso. Paquete Reportes.

2.6.4 Descripción de los casos de uso del sistema

En este epígrafe se realiza una descripción textual detallada de todos los casos de uso del sistema. Por cada caso de uso se exponen aspectos tales como: los actores que intervienen en ellos, su propósito general, un resumen de cómo se realizan, las referencias a los requisitos que cubren, entre otros.

Las descripciones de los casos de uso aparecen en los anexos siguientes:

Paquete de Seguridad:

1. Autenticar usuario. Ver Anexo II.1
2. Cambiar contraseña. Ver Anexo II.2

Paquete de Administración:

3. Gestionar vendedor. Ver Anexo II.3
4. Gestionar comprador. Ver Anexo II.4
5. Gestionar usuario. Ver Anexo II.5

Paquete de Gestión:

6. Gestionar ventas por matadero. Ver Anexo II.6
7. Gestionar ventas por nevera. Ver Anexo II.7
8. Gestionar ventas por municipios. Ver Anexo II.8

Paquete de Reportes:

9. Mostrar las ventas realizadas en el Municipio.
10. Mostrar las ventas realizadas en la Nevera.
11. Mostrar las ventas realizadas en el Matadero.
12. Mostrar las ventas realizadas al MINAL-DIVISA en un mes y año determinado.
13. Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento.
14. Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio.
15. Mostrar comparación del mayor importe de ventas realizadas hasta el momento.

2.7 Conclusiones parciales

A través del modelo del sistema se identificaron actores del sistema (Técnico, Especialista, Administrador y Usuario del sistema), además se obtuvieron 15 casos de usos del sistema que fueron descritos y documentados a través del correspondiente diagrama. Se especificaron también los requerimientos funcionales y no funcionales. Todo contribuye a una mejor distribución de cómo funcionará el software a implementarse, así como la documentación de cada respuesta a las necesidades del cliente.

Capítulo III Construcción de la aplicación de escritorio para perfeccionar la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spiritus

En este capítulo se plasman los resultados de la etapa de diseño del sistema, utilizando UML para su modelado. Se elaboran los diagramas de clases de diseño y se elabora el diagrama de despliegue. Se describen también los principios del buen diseño para la interfaz, la seguridad y ayuda.

3.1 Diagrama de clases del diseño

Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema (Rumbaugh, Booch, & Jacobson, 2006).

Un diagrama de clases es una colección de elementos declaratorios del modelo, como clases, tipos y sus relaciones; conectados unos a otros y a sus contenidos en forma de grafo (Rumbaugh, Booch, & Jacobson, 2006).

3.1.1 Paquete Seguridad

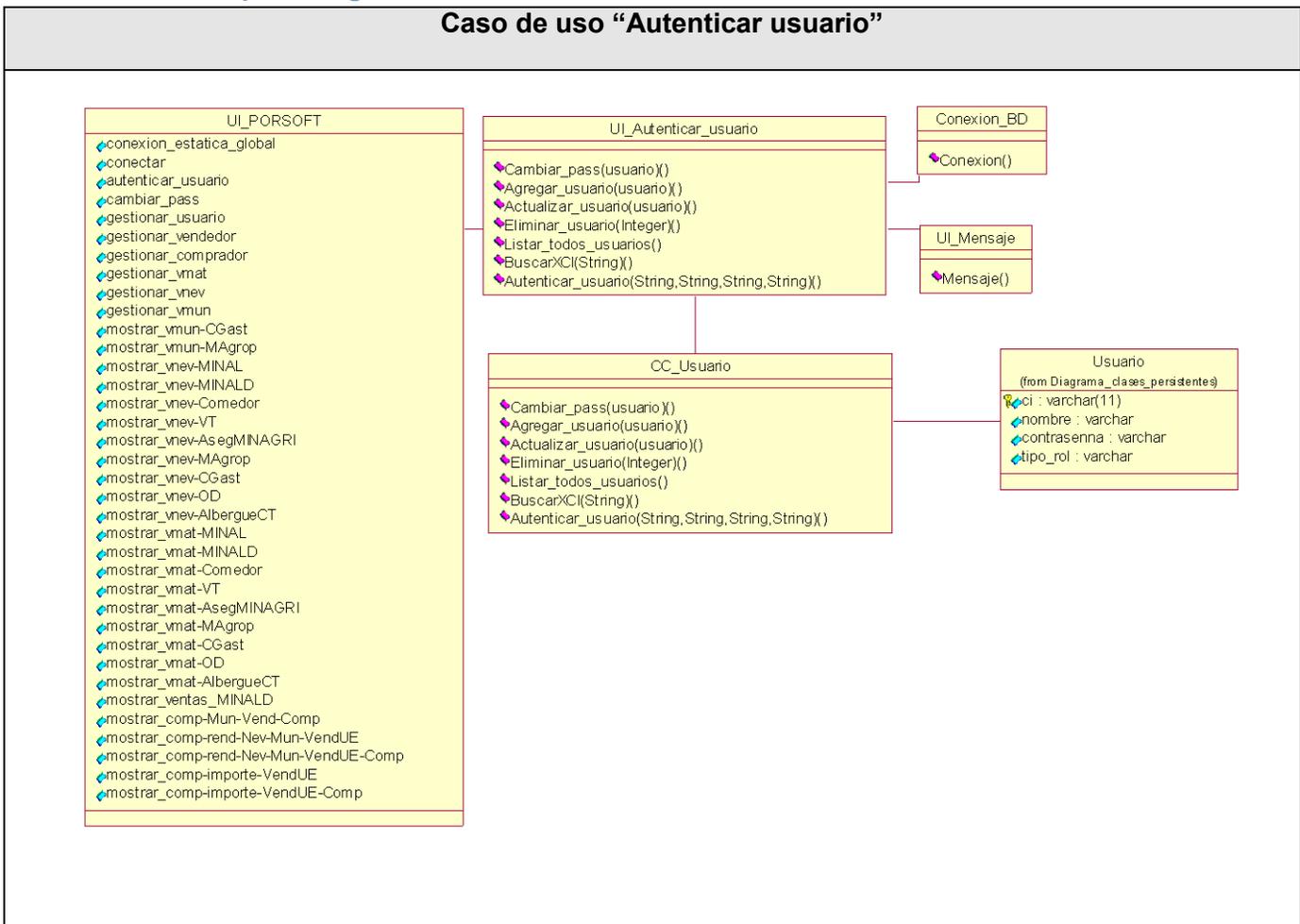


Figura # 3.1: Diagrama de clases del diseño del CU "Autenticar usuario".

Caso de uso "Cambiar contraseña"

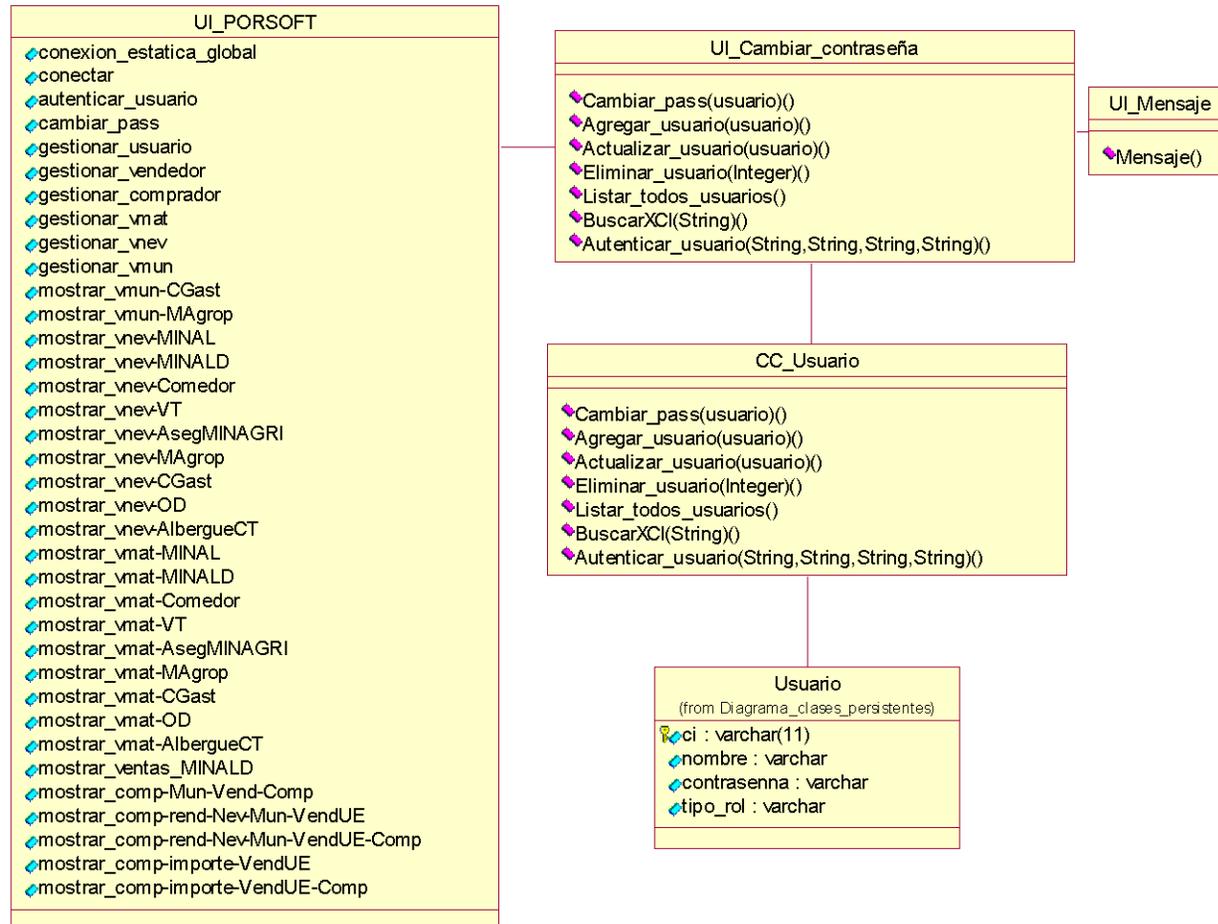


Figura # 3.2: Diagrama de clases del diseño del CU "Cambiar contraseña".

3.1.2 Paquete Administración

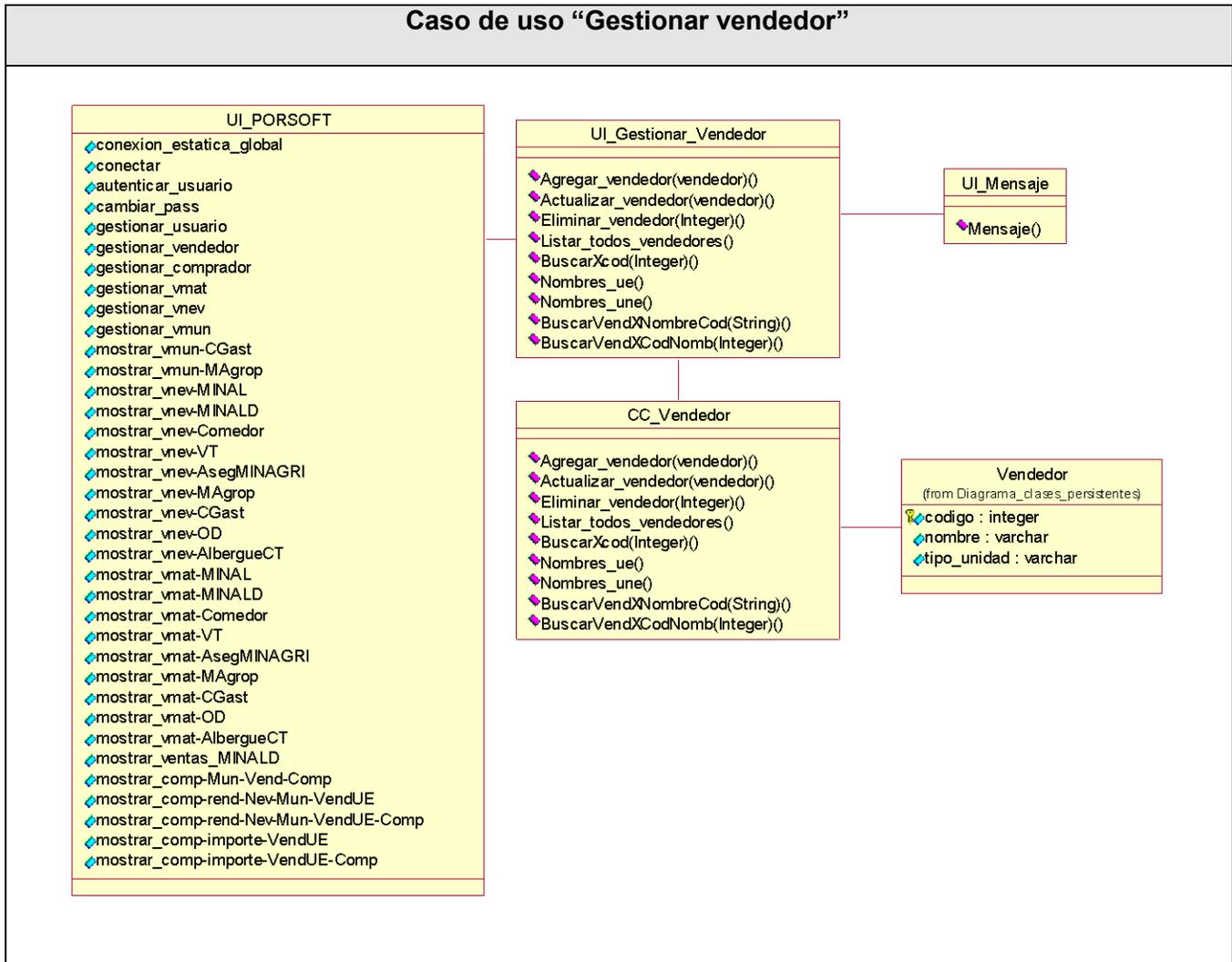


Figura # 3.3: Diagrama de clases del diseño del CU “Gestionar vendedor”.

Caso de uso "Gestionar comprador"

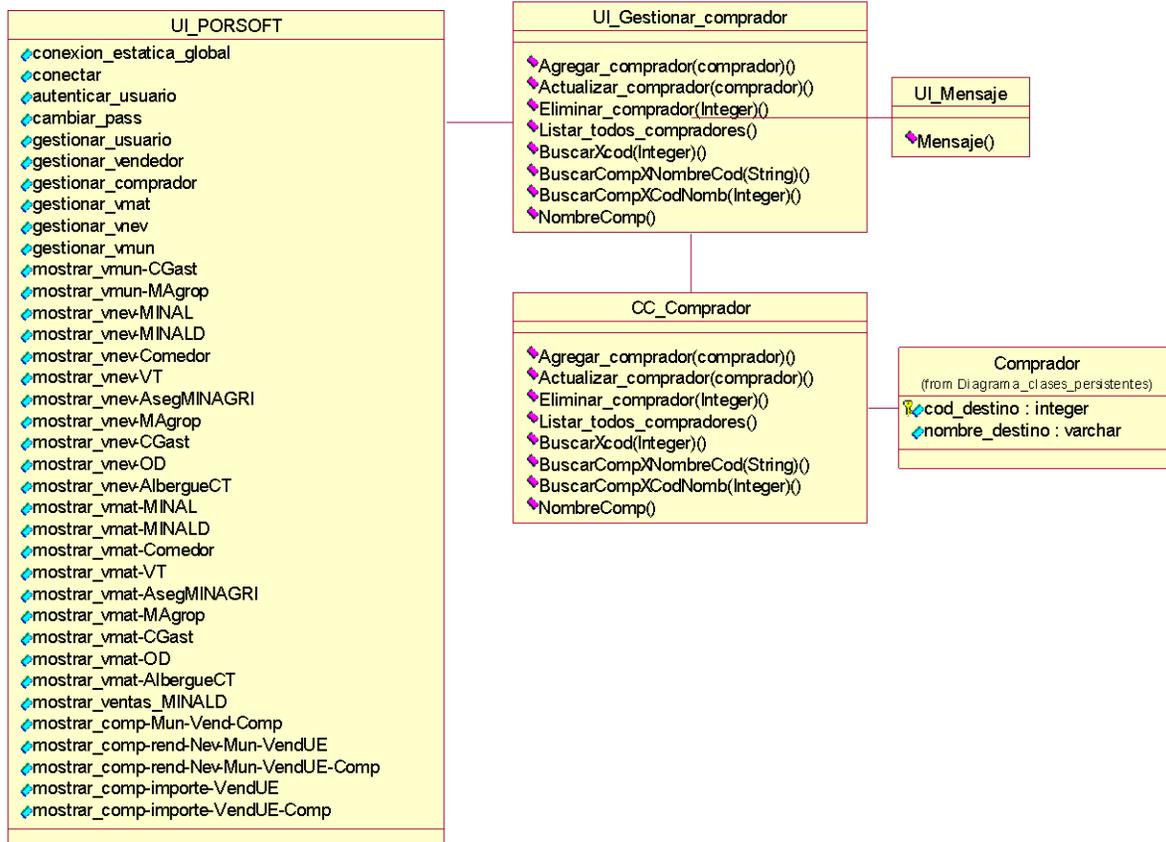


Figura # 3.4: Diagrama de clases del diseño del CU "Gestionar comprador".

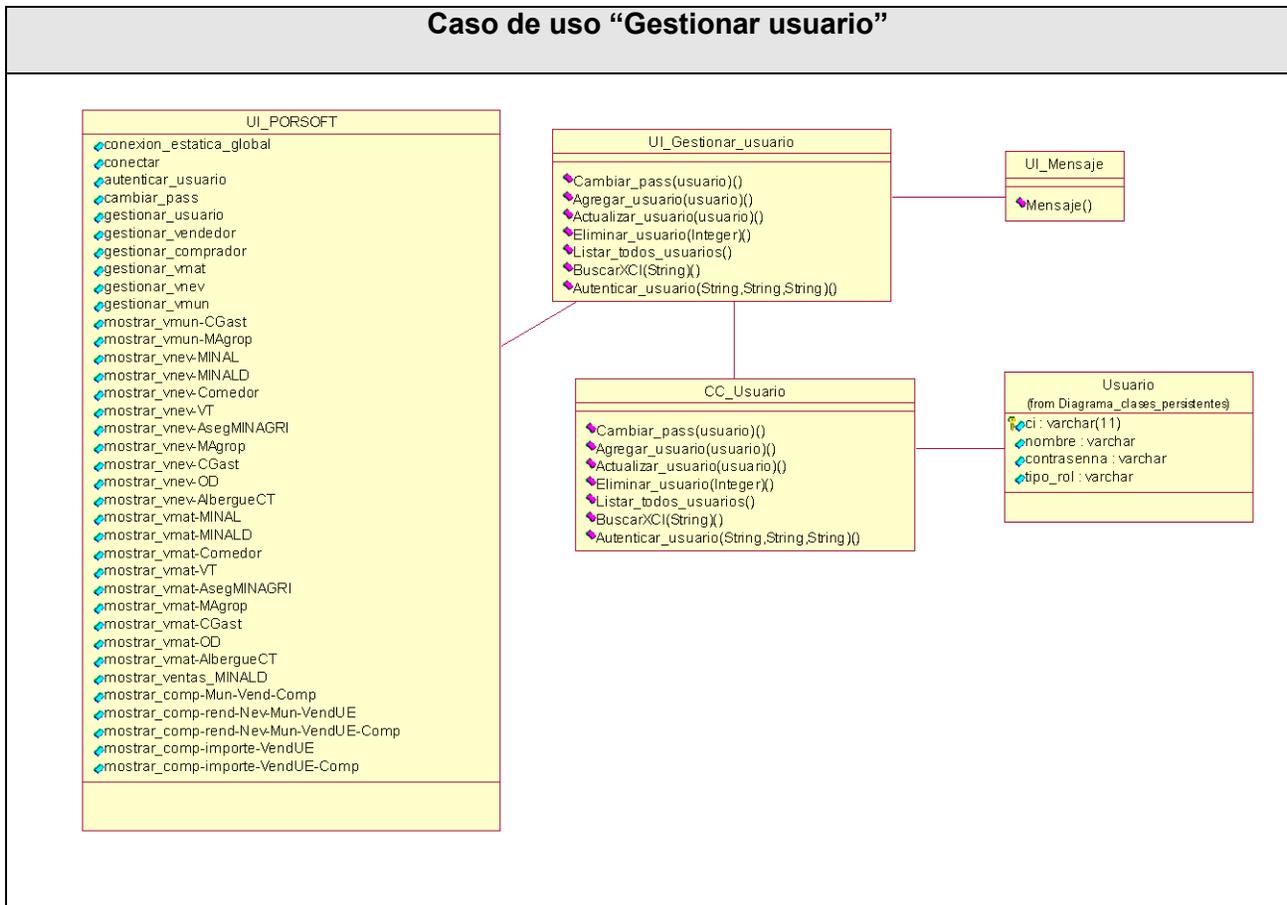


Figura # 3.5: Diagrama de clases del diseño del CU “Gestionar usuario”.

3.1.3 Paquete Gestión

Caso de uso “Gestionar venta por Matadero”

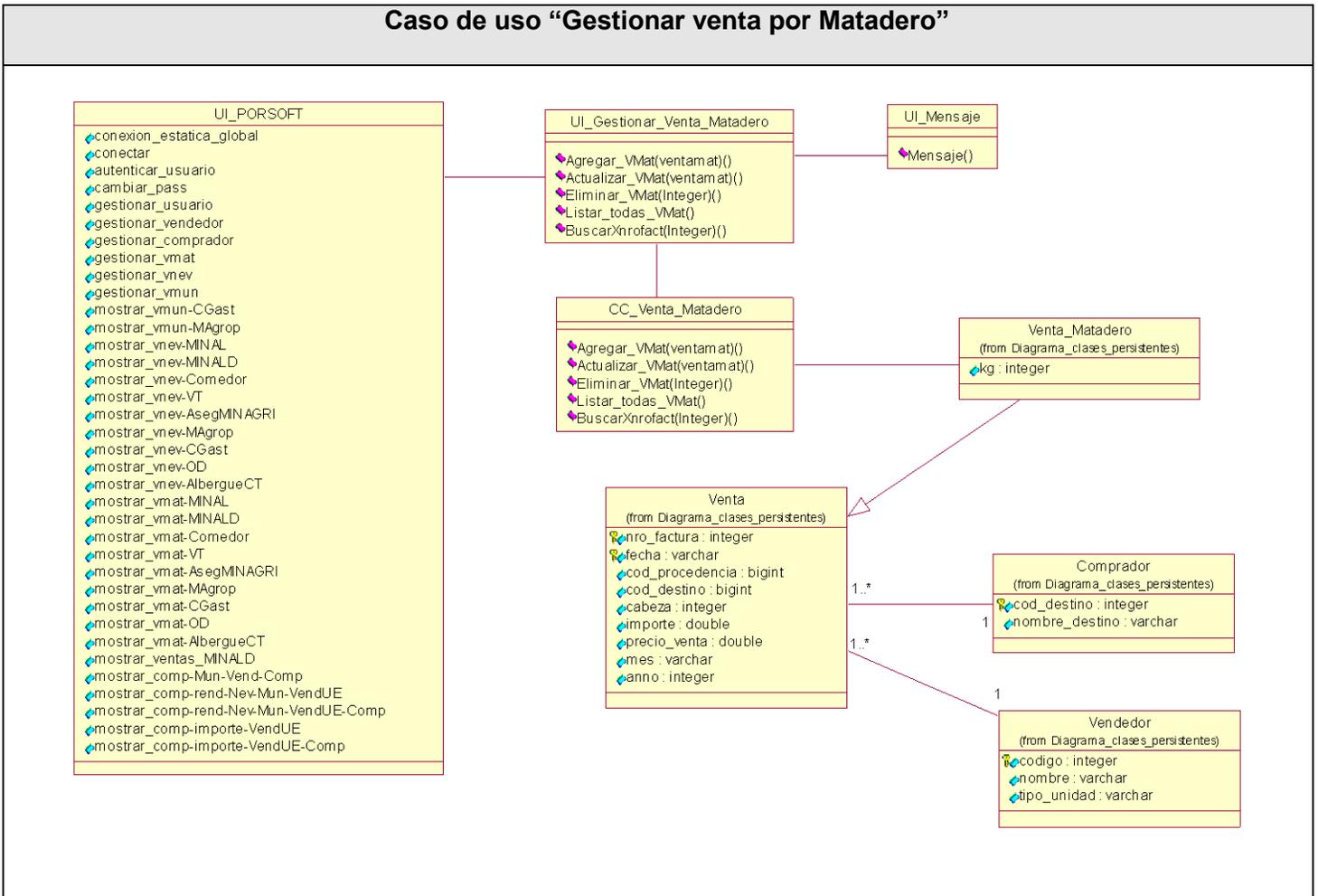


Figura # 3.6: Diagrama de clases del diseño del CU “Gestionar venta por matadero”.

Caso de uso "Gestionar venta por Nevera"

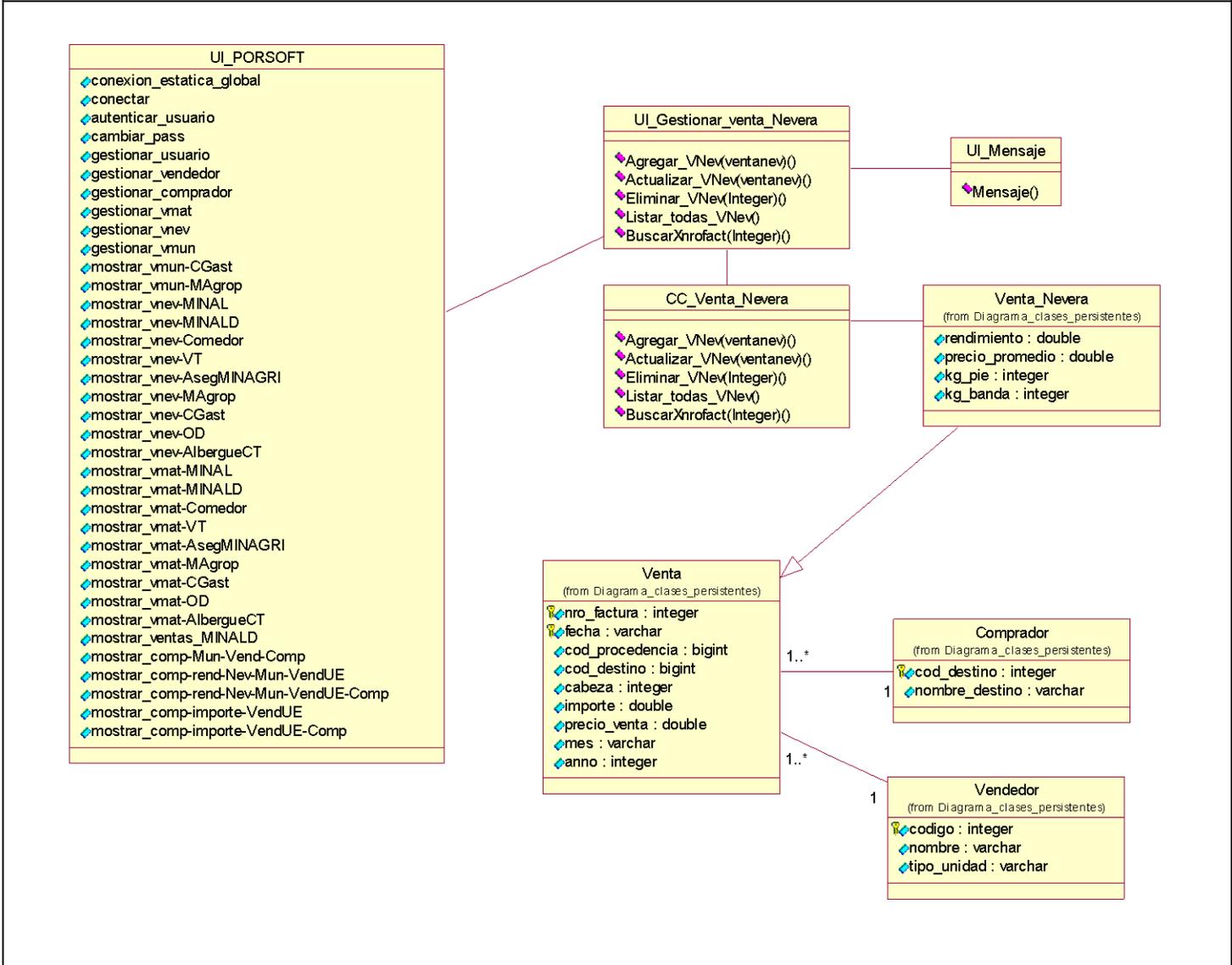


Figura # 3.7: Diagrama de clases del diseño del CU "Gestionar venta por nevera".

Caso de uso "Gestionar venta por Municipio"

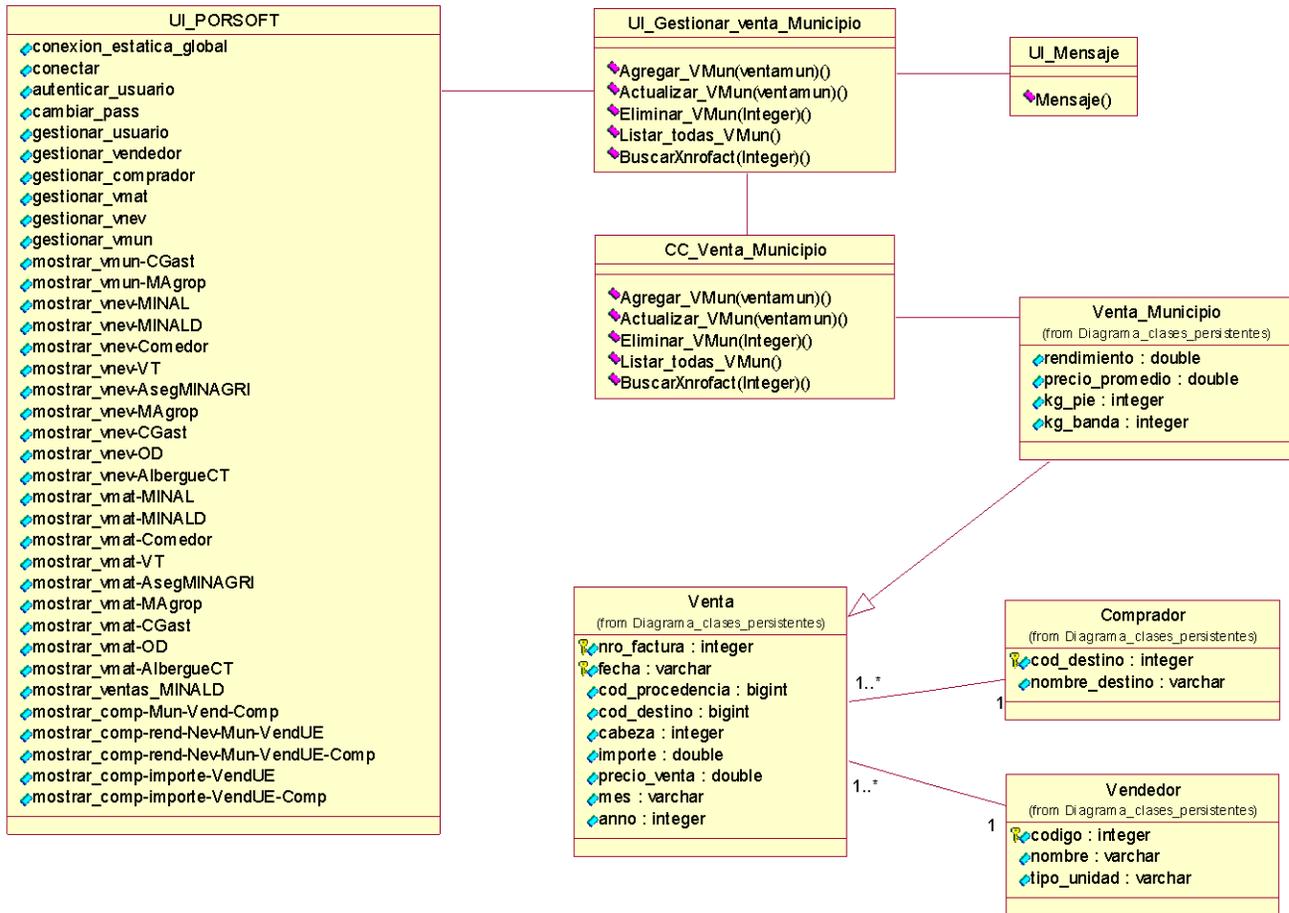


Figura # 3.8: Diagrama de clases del diseño del CU "Gestionar venta por municipio".

3.1.4 Paquete Reportes

Caso de uso “Mostrar las ventas realizadas en el Municipio”

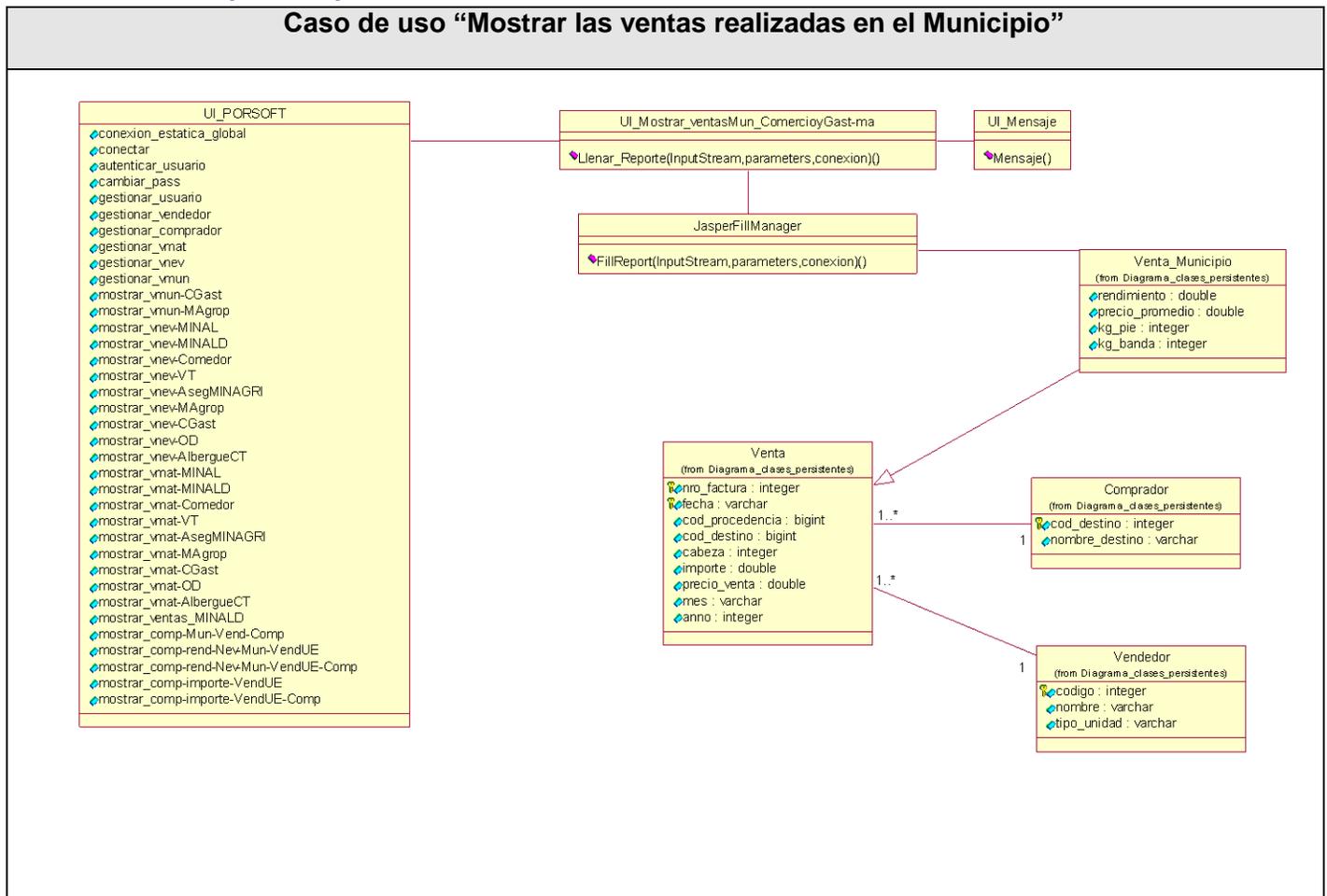


Figura # 3.9: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en el Municipio”.

Caso de uso “Mostrar las ventas realizadas en la Nevera”

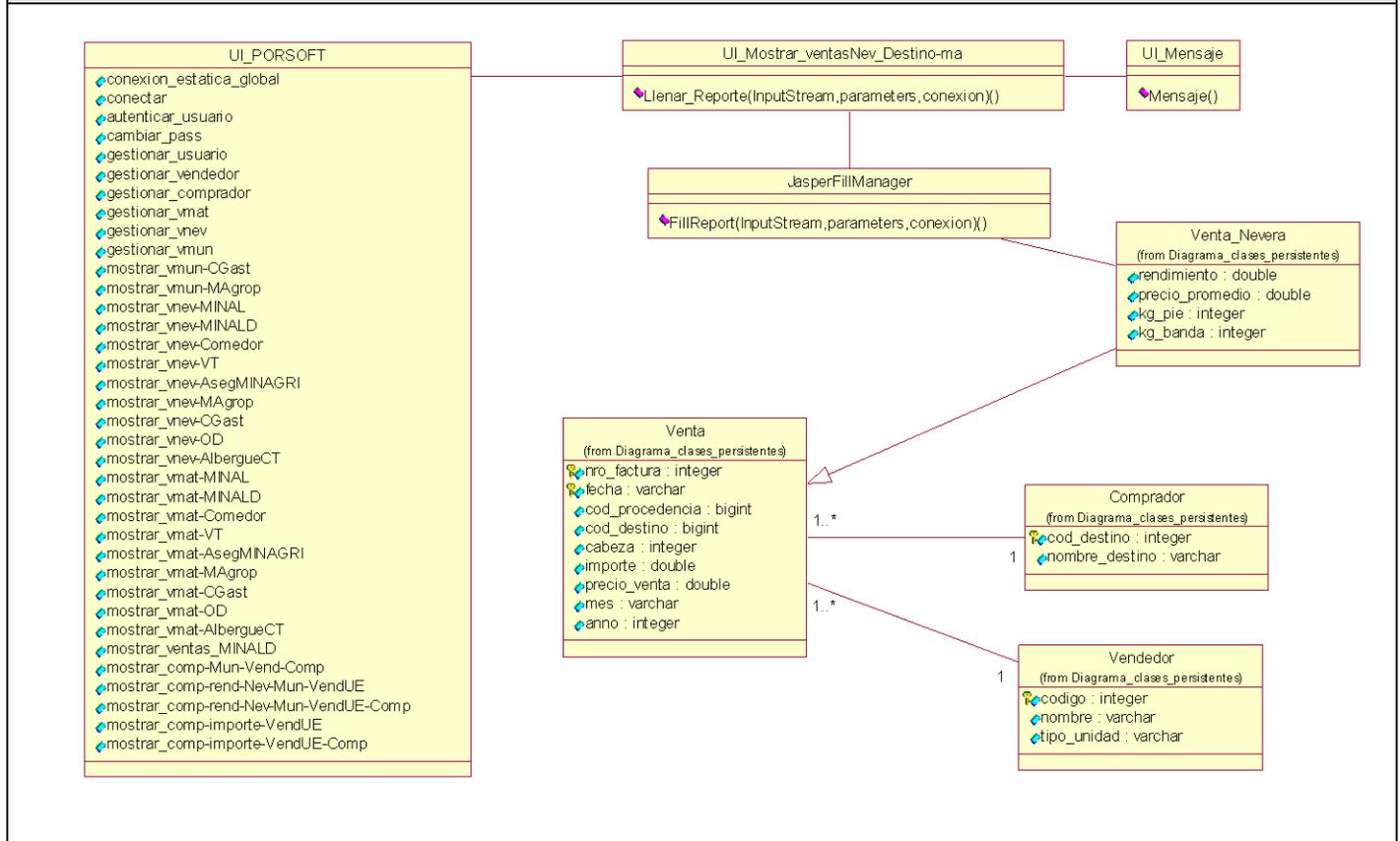


Figura # 3.10: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en la Nevera”.

Caso de uso “Mostrar las ventas realizadas en el Matadero”

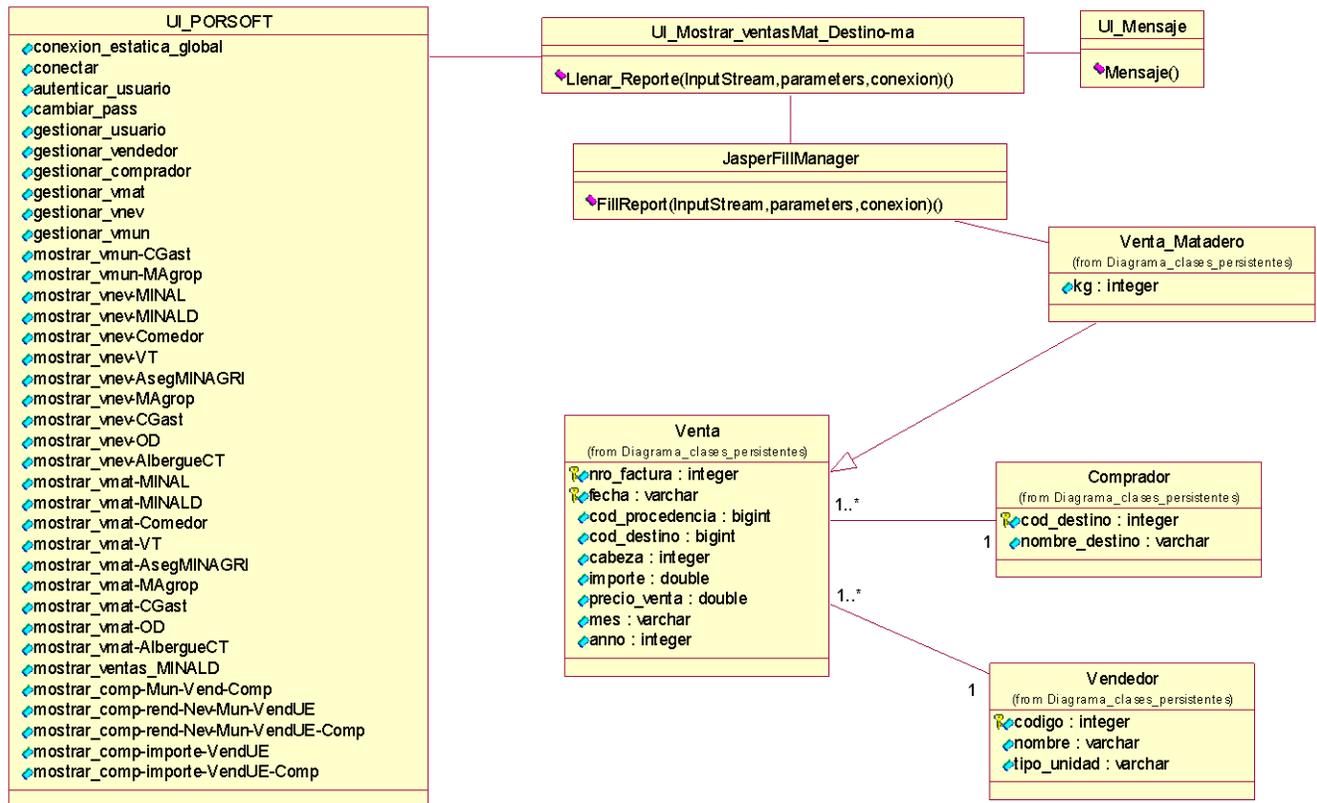


Figura # 3.11: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas en el Matadero”.

Caso de uso “Mostrar las ventas realizadas al MINAL-DIVISA en un mes y año determinado”

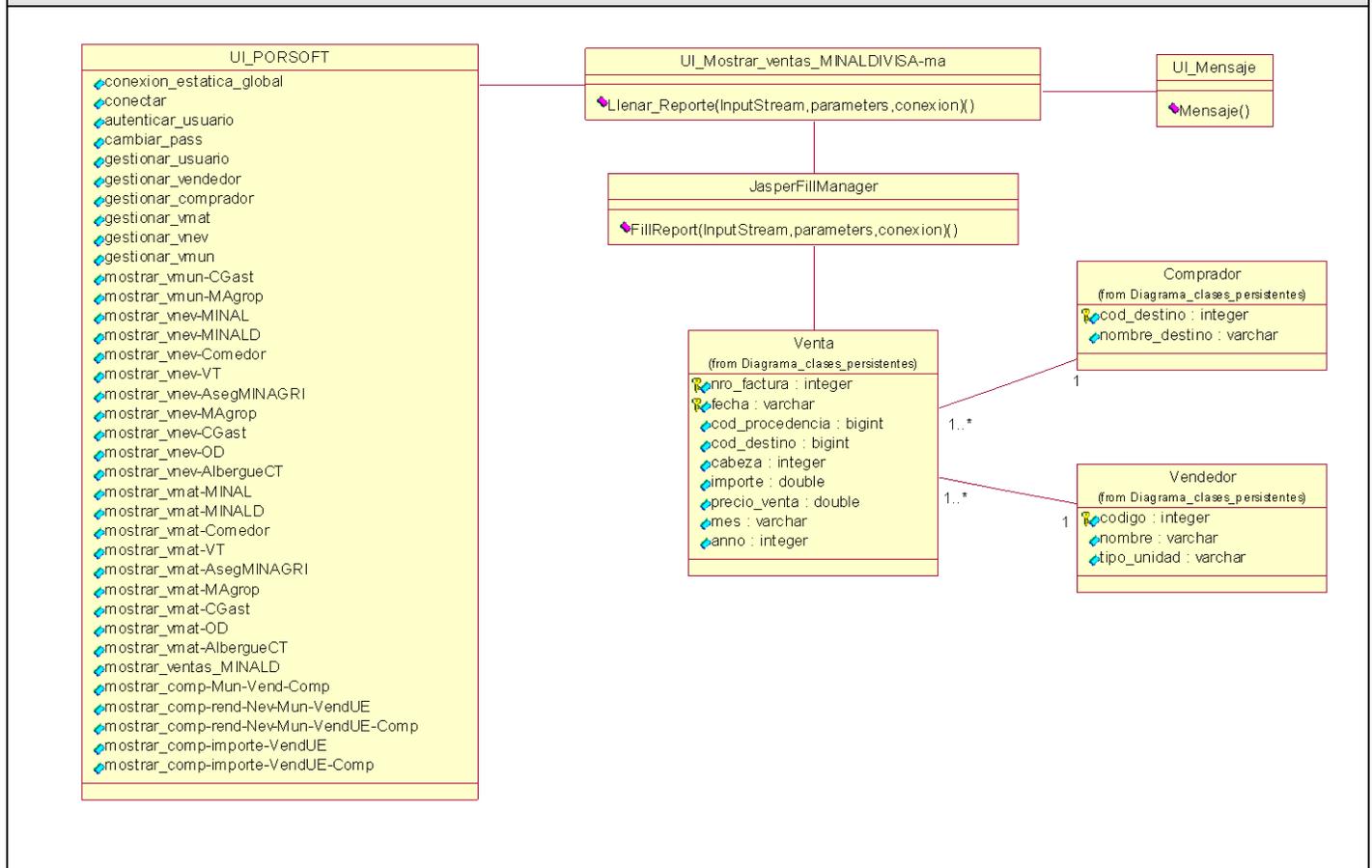


Figura # 3.12: Diagrama de clases del diseño del CU “Mostrar las ventas realizadas al MINAL-DIVISA en un mes y año determinado”.

Caso de uso “Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento”

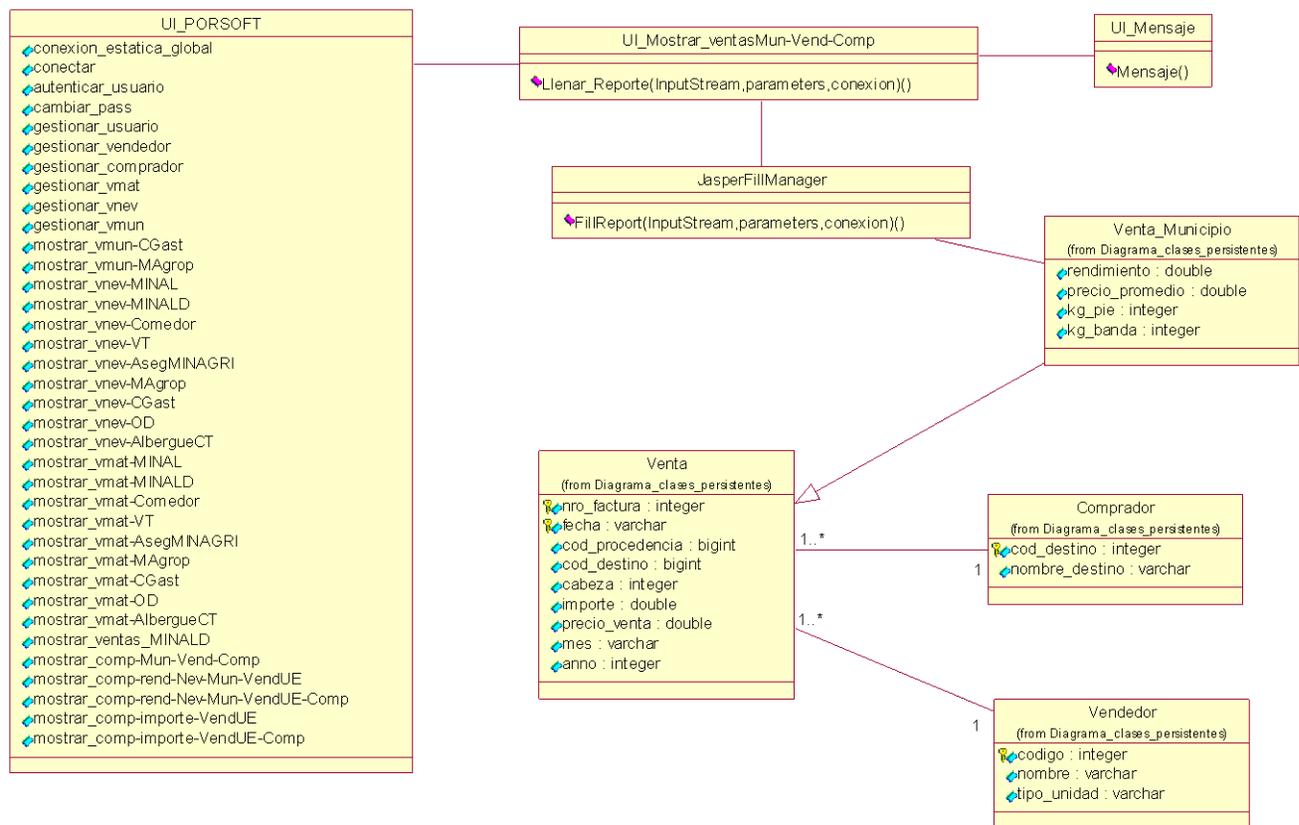


Figura # 3.13: Diagrama de clases del diseño del CU “Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento”.

Caso de uso “Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio”

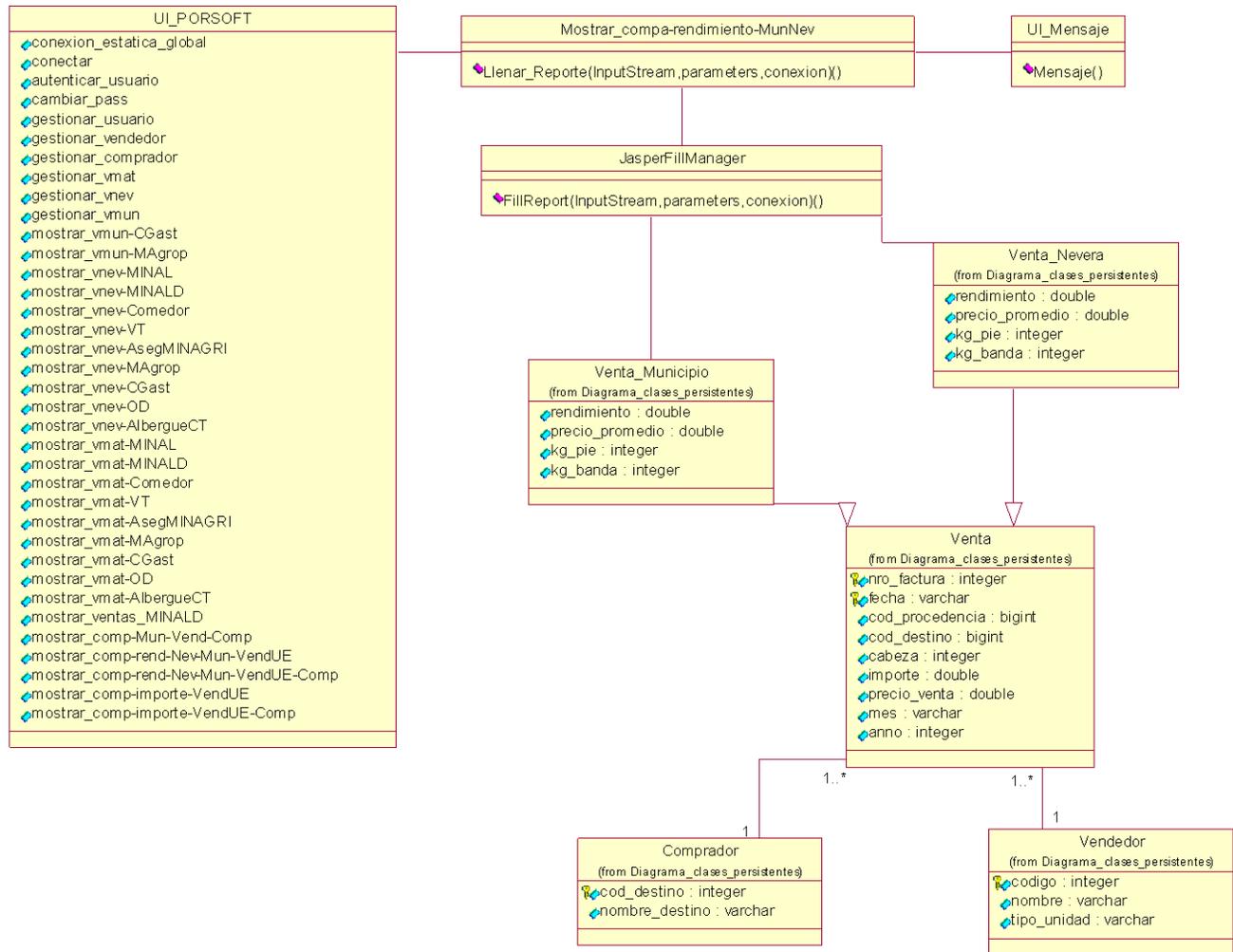


Figura # 3.14: Diagrama de clases del diseño del CU “Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio”.

Caso de uso “Mostrar comparación del mayor importe de ventas realizadas”

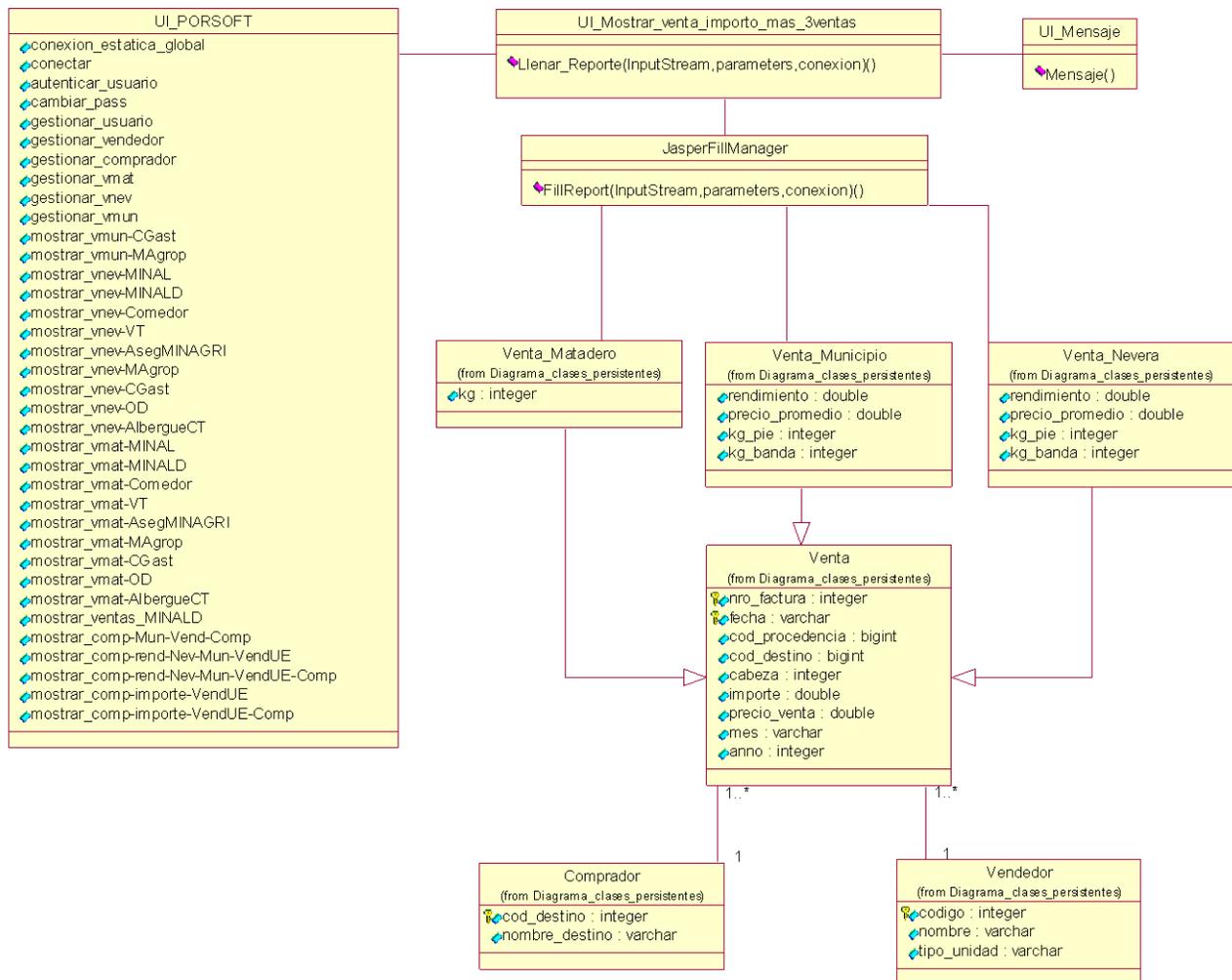


Figura # 3.15: Diagrama de clases del diseño del CU “Mostrar comparación del mayor importe de ventas realizadas”.

3.2 Diseño de la Base de Datos

Una base de datos (del inglés Database) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. El diseño de la base de datos es una de las tareas más importantes en la construcción de un sistema, ya que una base de datos bien diseñada contendrá información correcta, almacenará los datos más eficientemente y será más fácil de gestionar y de mantener (Wesley, 2005).

3.2.1 Diagrama de clases persistentes

El diagrama del modelo lógico de datos o diagrama de clases persistentes, muestra las clases capaces de mantener su valor en el espacio y en el tiempo (Rumbaugh, Booch, & Jacobson, 2006).

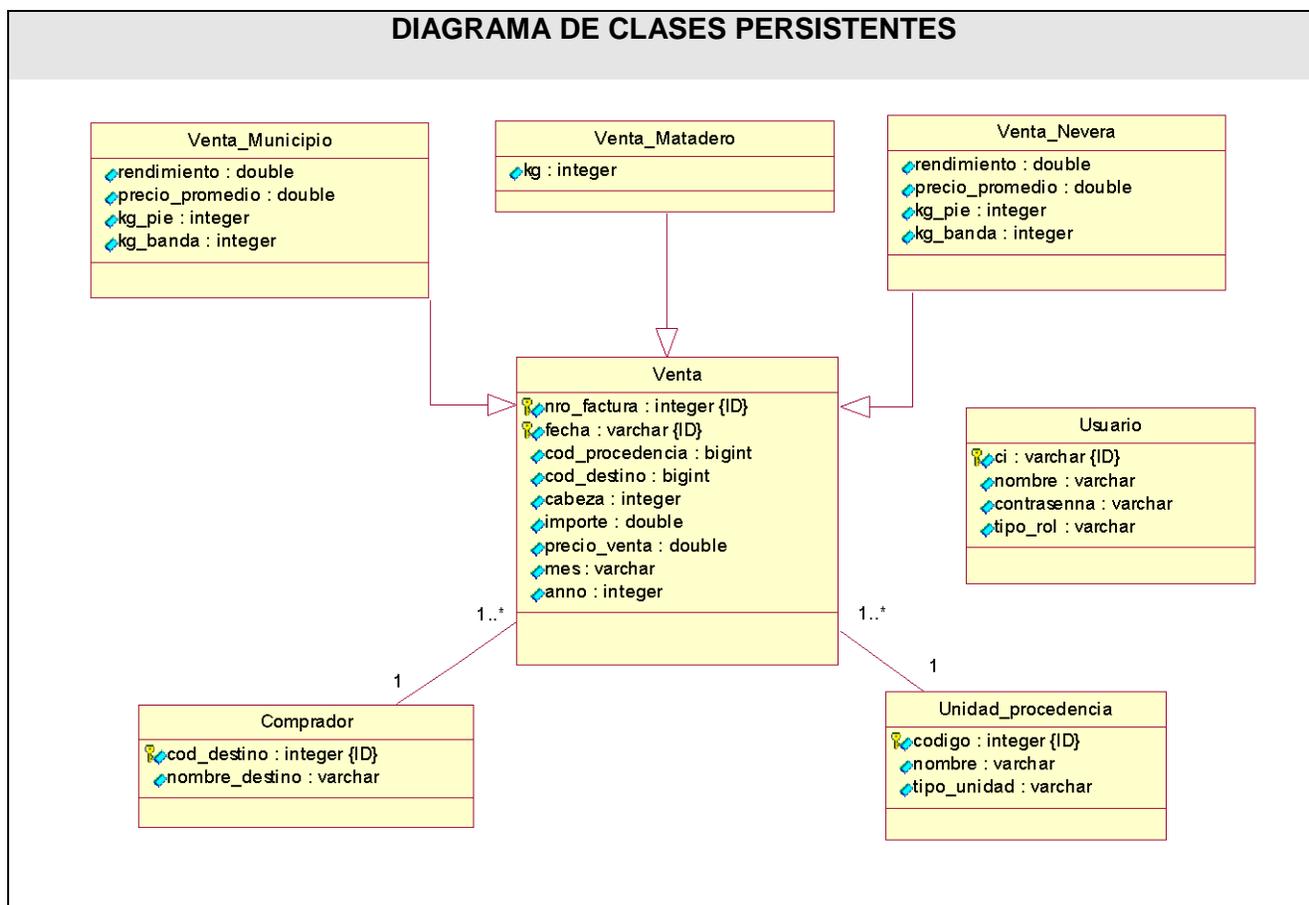


Figura # 3.16. Diagrama de clases persistentes.

3.2.2 Diagrama físico de datos.

El modelo físico de datos, representa la estructura o descripción física de las tablas de la base de datos, obtenido a partir del modelo lógico de datos (Rumbaugh, Booch, & Jacobson, 2006).

En el Anexo III.1 se muestra dicho modelo.

3.3 Modelo de Implementación

El modelo de implementación denota la implementación del sistema en términos de componentes y subsistemas de implementación. Describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración, y modularización disponibles en el entorno de la implementación y en el lenguaje o lenguajes de programación utilizados, y como dependen los componentes unos de otros (Cáceres, 2005).

3.3.1 Diagrama de despliegue

El diagrama de despliegue muestra la forma en que los componentes se desplegarán en el sistema, en este se detalla las especificaciones de la red, del servidor y los requisitos de hardware. Según Jacobson, Booch, & Rumbaugh un diagrama de despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos (Rumbaugh, Booch, & Jacobson, 2006).

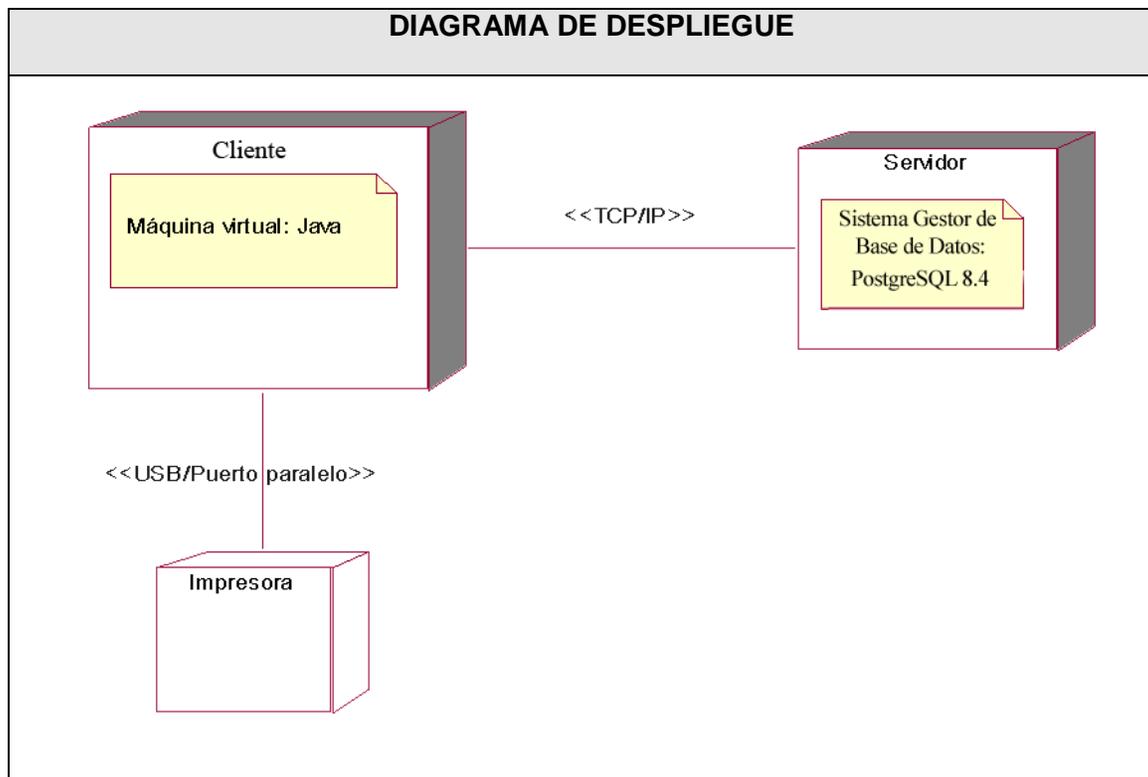


Figura # 3.17. Diagrama de despliegue.

3.3.2 Diagrama de componentes

Los diagramas de componentes se utilizan para modelar la vista de implementación estática de un sistema. Muestran tanto los componentes de software (código fuente, binario y ejecutable) como las relaciones lógicas entre ellos en un sistema. Y como todos los diagramas, también pueden contener paquetes o subsistemas utilizados para agrupar elementos del modelo.

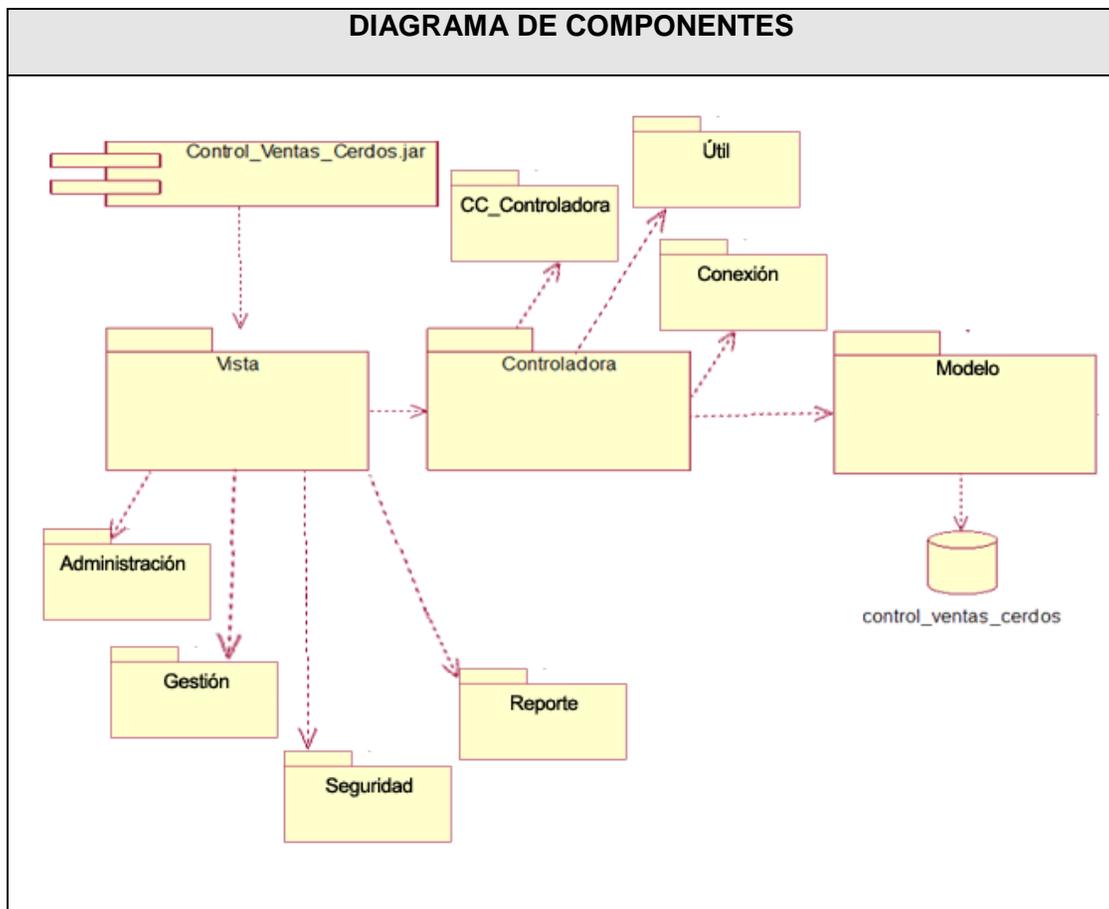


Figura # 3.18. Diagrama de componentes.

Descripción de Diagrama de Componentes:

Control_Ventas_Cerdos.jar: Este componente representa al ejecutable de la aplicación.

Vista: Dentro de este sistema se encuentran todas las clases que muestran una interfaz al usuario. Existen subsistemas tales como administración, reporte, seguridad y gestión. Representa la capa de presentación de la aplicación.

Controladora: Este sistema representa la capa de la lógica del negocio. Dentro de él se encuentran todas las clases del negocio, principalmente clases controladoras que actúan de intermediaria entre la capa de presentación al usuario y la capa de datos. Existen subsistemas tales como cc_controladora y conexion.

Modelo: Este subsistema representa la lógica del negocio, dentro de él se encuentran las clases entidades con las entidades del negocio y sus atributos.

control_ventas_cerdos: Representa el Sistema Gestor de Base de Datos donde se almacena toda la información. En este sistema se utilizó PostgreSQL para manipular toda esta información.

3.4 Diseño de la interfaz del sistema.

Uno de los aspectos más relevantes en el uso de un sistema que lo puede llevar a un éxito o a un fracaso, es la consistencia de la interfaz de usuario. El producto debe ser legible y con colores agradables que no llamen mucho la atención, debido a que su uso requiere de concentración. El vocabulario a utilizar será el más adecuado para el usuario sin emplear palabras técnicas de informática. El software brinda un menú superior permanente que brindará las diferentes funciones que puede realizar el usuario en dependencia del rol que juegue este dentro del sistema. Además aparecen una serie de iconos que corresponden a las funcionalidades más comunes del sistema.

3.5 Tratamiento de errores.

El sistema propuesto presenta un nivel de validación constante de la información, con el propósito de minimizar las posibilidades de introducir información errónea por parte del usuario. En caso de errores se le comunica el error cometido en cuadros de alerta. Los mensajes de error que emite el sistema se muestran en un lenguaje de fácil comprensión para los usuarios.

3.6 Concepción general de la ayuda.

El sistema cuenta con una ayuda capaz de brindarle al usuario una detallada explicación de cómo sería la manipulación y funcionamiento del mismo. La ayuda está formada por un menú interno que contiene todas las funcionalidades del sistema. El usuario podrá navegar por cada uno de ellos y consultar su funcionalidad, descrita de un modo fácil de entender.

3.7 Concepción del sistema de seguridad y protección.

El sistema mantiene un fuerte mecanismo de seguridad y protección, basado en login y contraseña para el acceso al mismo. Las personas encargadas de actualizar cualquier información presente en la base de datos del sistema, tendrán un login y una contraseña

que sólo ellos podrán usar, evitando que esta acción pueda ser realizada por cualquier usuario que acceda a él. Además cada usuario del sistema tendrá solamente permiso de acceso a las funcionalidades correspondientes a su rol.

3.8 Conclusiones parciales.

En el presente capítulo se mostraron los resultados de la etapa de diseño e implementación del sistema. Por medio de los diagramas de clases de diseño se identificaron cada una de las clases que participan en la ejecución de cada caso de uso. Además a partir del diseño de la base de datos se obtuvo la base de datos a implementar en el sistema gestor de base de datos: PostgreSQL. Se desplegó el sistema en dos unidades de procesamiento, una destinada al cliente y la otra al servidor, lo cual fue documentado a través del diagrama de despliegue. Por último se estructuró cada uno de los sistemas y subsistemas que intervienen en la implementación por medio del diagrama de componentes. Se describieron una serie de estándares a seguir que mantienen la integridad de la aplicación, tales como el diseño de la interfaz, la concepción del tratamiento de errores y de la ayuda, y la seguridad de la aplicación.

CONCLUSIONES

1. Se determinaron las herramientas y lenguajes apropiados que, mediante la metodología RUP utilizando el lenguaje de modelado UML, permiten el desarrollo una aplicación para la gestión de los datos de las ventas de masa porcina en la Empresa Provincial Porcina en Sancti Spíritus. Para ello se determinó una interfaz de usuario para escritorio debido a que es muy utilizada en entornos de oficina. Java fue elegido como lenguaje de programación por su amplio uso y efectividad en la creación de aplicaciones de escritorio; además de PostgreSQL como sistema gestor de base de datos, considerado el mejor dentro de todas las alternativas en el software libre.
2. Se diseñó una propuesta para aplicación de escritorio que permite gestionar los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus. El punto de partida fue la descripción de los procesos de negocio y sistema, la modelación de los casos de uso correspondientes, así como los requisitos funcionales y no funcionales definidos por el usuario.
3. Se implementó una aplicación de escritorio capaz de gestionar los datos de las ventas de masa porcina en la Empresa Provincial Porcina de Sancti Spíritus. Para ello se siguieron los principios del diseño, los estándares de codificación y se ajustó a los requerimientos funcionales y no funcionales del sistema. El sistema se desplegó a través de dos nodos, en los cuales se distribuyen los componentes de la aplicación, la cual está basada en una arquitectura cliente-servidor con dos capas que proporciona escalabilidad y posibilidades de mantenimiento del código. Dicha aplicación queda sustentada bajo los preceptos de software libre, el patrón de diseño MVC y la programación orientada a objeto, lo que brinda flexibilidad y posibilidades para futuras modificaciones y mejoras.

RECOMENDACIONES

1. Aplicar las herramientas, lenguajes y metodología utilizada para la gestión de ventas de masa porcina, en el desarrollo de una nueva funcionalidad para la gestión de las compras en la Empresa Provincial Porcina de Sancti Spíritus.
2. Construir una interfaz web para la divulgación de la información estadística de la Empresa Provincial Porcina de Sancti Spíritus para directivos y económicos.

REFERENCIAS BIBLIOGRÁFICAS

Ajá Quiroga, L. (2006). Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.

Burbeck, S. (2007). *Application programming in Smalltalk-80: How to use Model-View-Controller (MVC)*. Technology Series.

Cáceres, L. (2005). *Sistema de Promoción y Gestión Comercial para la Oficina de Transferencia Tecnológica de la Universidad de Cienfuegos*. Ciudad de La Habana: Trabajo de Diploma.

Capote Marrero, B. (2006). Recuperado el Febrero de 2006, de La gestión de información como herramienta fundamental en el desarrollo de los centros toxicológicos: www.biblioteca.utp.edu.co/perfildsi.html.

Castillo, I. M. (2008). Implementación de un Servicio Web para la Administración de los departamentos docentes. La Habana.

Concepto de gestión. (2003, Septiembre). *Diccionario de uso del español de América y España*. Barcelona: SPES EDITORIAL. Recuperado de www.vox.es

Connolly, T. M., & Begg, C. E. (2005). *Sistemas de Bases de Datos: Un Enfoque Práctico Para Diseño, Implementación y Gestión* (4ta ed.). Madrid: Pearson Education.

Date, C. J. (2001). *Introducción a los sistemas de bases de datos* (7ma ed.). México: Pearson Education.

Elmasri, R., & Navathe, S. B. (2004). *Fundamentals of Database Systems* (4ta ed.). USA: Pearson Education

Fisher, M., Ellis, J., & Bruce, J. (2003). *JDBC™ API Tutorial and Reference* (3ra ed.). Addison Wesley.

Lago, R. (Abril de 2007). *Patrones de diseño software*.

Mato García, R. M. (2006). *Sistema de Base de Datos*. La Habana: Félix Varela.

Microsoft SQL Server. (2004). *Libros en pantalla de SQL Server* (8va ed.). Microsoft Corporation.

Netbeans. (2009). Recuperado el 10 de Noviembre de 2010, de <http://www.netbeans.org>

Ponjuán Dante, G. (1998). *Gestión de información en las organizaciones: principios, conceptos y aplicaciones*. Santiago de Chile: Universidad de Chile

Pressman, R. S. (2007). *Ingeniería del Software Un enfoque práctico*. La Habana: Félix Varela.

Real Academia Española. (2001). Recuperado el 3 de Febrero de 2010, de www.raes.es

Riveros, F. (2008). *Gestor de Base de Datos: MySQL, PostgreSQL, SQLite*. Recuperado el 16 de Febrero de 2010, de [http://www.eaprende.com/base de datos SQL Server con PHP y ADODB.html](http://www.eaprende.com/base_de_datos_SQL_Server_con_PHP_y_ADODB.html)

Rumbaugh, J., Booch, G., & Jacobson, I. (2006). *El proceso unificado de desarrollo de software*. La Habana: Félix Varela.

Schildt, H. (2005). *Java™:A Beginner's Guide*. New York: McGraw-Hill.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *Fundamentos de bases de datos* (4ta ed.). Madrid: McGraw-Hill.

ThePostgreSQL Global DevelopmentGroup. (2003). PL/pgSQL - SQL Procedural Language. En T. G. DevelopmentGroup, *PostgreSQL 7.4Devel Documentation*. ThePostgreSQL Global Development Group.

Wesley, A. (2005). *Introducción a los Sistemas de Bases de Datos*.

Woodman, L. (1985). Information management in large organizations. En W. L, *Information management from strategies to action* (págs. 95-114). London: ASLIB.

ANEXO

ANEXO II.1 Descripción del CU "Autenticar usuario".

CU #1	Autenticar usuario	
Actores	Usuario del sistema.	
Propósito	Autenticarse el tipo de usuario para entrar al sistema	
Resumen		
El caso de uso inicia cuando el usuario introduce su nombre de usuario y su contraseña y culmina cuando el sistema lo verifica permitiéndole la entrada al mismo según los permisos que tenga asignados.		
Referencias	RF-1	
Prototipo	Anexo III.2	

ANEXO II.2 Descripción del CU "Cambiar contraseña".

CU #2	Cambiar Contraseña	
Actores	Usuario del sistema.	
Propósito	Cambiar la contraseña de un usuario del sistema.	
Resumen		
El Caso de uso se inicia cuando el usuario accede al sistema para cambiar la contraseña correspondiente a ese usuario. Para cambiar la contraseña el usuario deberá ingresar en		

una ventana de la aplicación su nombre de usuario, la contraseña actual, luego la contraseña nueva y rectificarla. El caso de uso finaliza cuando se ejecuta la operación antes mencionada.

Referencias	RF-2
Prototipo	Anexo III.3

ANEXO II.3 Descripción del CU "Gestionar vendedor".

CU #3	Gestionar vendedor.
Actores	Especialista.
Propósito	Ingresar, modificar ó eliminar una unidad de procedencia en el centro.
<p>Resumen</p> <p>El Caso de uso se inicia cuando el Especialista accede al sistema para registrar, modificar o eliminar la información correspondiente a un vendedor. Para registrar un nuevo vendedor el especialista ingresa en una ventana los datos requeridos. Para modificar los datos de un vendedor el especialista debe seleccionarlo para modificar sus datos por medio de una búsqueda en todos los vendedores de la base de datos para conocer su existencia. Para eliminar se debe seleccionar el vendedor que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>	
Referencias	RF-3
Prototipo	Anexo III.4

ANEXO II.4 Descripción del CU "Gestionar comprador".

CU #4	Gestionar comprador.	
Actores	Especialista.	
Propósito	Ingresar, modificar o eliminar un comprador en el centro.	
Resumen		
<p>El Caso de uso se inicia cuando el Especialista accede al sistema para registrar, modificar o eliminar la información correspondiente a un comprador. Para registrar un nuevo comprador el especialista ingresa en una ventana los datos requeridos. Para modificar los datos del comprador el especialista debe seleccionarlo para modificar sus datos por medio de una búsqueda en todos los compradores de la base de datos para conocer su existencia. Para eliminar se debe seleccionar el comprador que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>		
Referencias	RF-4	
Prototipo	Anexo III.5	

ANEXO II.5 Descripción del CU "Gestionar usuario".

CU #5	Gestionar usuario.	
Actores	Administrador.	
Propósito	Ingresar, modificar o eliminar un usuario.	
Resumen		
<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar,</p>		

modificar o eliminar la información correspondiente al usuario. Para registrar un nuevo usuario debe incluirse las características como su nombre, la contraseña y su tipo de rol. Para modificar los datos del usuario el administrador debe seleccionar el usuario al que le desea modificar sus datos por medio de una búsqueda en todos los usuarios de la base de datos para conocer su existencia. Para eliminar se debe seleccionar al usuario que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.

Referencias	RF-5
Prototipo	Anexo III.6

ANEXO II.6 Descripción del CU "Gestionar venta por matadero".

CU #6	Gestionar venta por matadero.
Actores	Técnico.
Propósito	Ingresar, modificar o eliminar una venta por matadero.
Resumen	
<p>El Caso de uso se inicia cuando el Técnico accede al sistema para registrar, modificar o eliminar la información correspondiente a una venta por matadero. Para registrar una nueva venta por matadero el Técnico ingresa en una ventana los datos requeridos. Para modificar los datos de la venta por matadero el Técnico debe seleccionarla para modificar sus datos por medio de una búsqueda en todas las ventas de la base de datos para conocer su existencia. Para eliminar se debe seleccionar la venta por matadero que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>	
Referencias	RF-6

Prototipo	Anexo III.7
------------------	-------------

ANEXO II.7 Descripción del CU "Gestionar venta por nevera".

CU #7	Gestionar venta por nevera.
Actores	Técnico.
Propósito	Ingresar, modificar o eliminar una venta por nevera.
Resumen	
<p>El Caso de uso se inicia cuando el Técnico accede al sistema para registrar, modificar o eliminar la información correspondiente a una venta por nevera. Para registrar una nueva venta por nevera el Técnico ingresa en una ventana los datos requeridos. Para modificar los datos de la venta por nevera el Técnico debe seleccionarla para modificar sus datos por medio de una búsqueda en todas las ventas de la base de datos para conocer su existencia. Para eliminar se debe seleccionar la venta por nevera que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>	
Referencias	RF-7
Prototipo	Anexo III.8

ANEXO II.8 Descripción del CU "Gestionar venta por municipio".

CU #8	Gestionar venta por municipio.
Actores	Técnico.
Propósito	Ingresar, modificar o eliminar una venta por municipio.

Resumen	
<p>El Caso de uso se inicia cuando el Técnico accede al sistema para registrar, modificar o eliminar la información correspondiente a una venta por municipio. Para registrar una nueva venta por municipio el Técnico ingresa en una ventana los datos requeridos. Para modificar los datos de la venta por municipio el Técnico debe seleccionarla para modificar sus datos por medio de una búsqueda en todas las ventas de la base de datos para conocer su existencia. Para eliminar se debe seleccionar la venta por municipio que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>	
Referencias	RF-8
Prototipo	Anexo III.9

ANEXO II.9 Descripción del CU "Mostrar las ventas realizadas en el Municipio".

CU #9	Mostrar las ventas realizadas en el Municipio
Actores	Usuario del sistema.
Propósito	Mostrar las ventas realizadas en un municipio hacia un destino en un mes y año especificado por el usuario.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener información acerca de las ventas que se han realizado por un municipio hacia un destino en un mes y año especificado. El Usuario le especifica al sistema el municipio, mes y el año requerido para el reporte. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.</p>	

Referencias	RF-9
Prototipo	Anexo III.10

ANEXO II.10 Descripción del CU "Mostrar las ventas realizadas en la Nevera".

CU #10	Mostrar las ventas realizadas en la Nevera.
Actores	Usuario del sistema.
Propósito	Mostrar las ventas que se han realizado por la Nevera hacia un destino en un mes y año determinado.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener información acerca de las ventas que se han realizado por la Nevera hacia un destino en un mes y año especificado. El Usuario le especifica al sistema el destino, mes y el año requerido para el reporte. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.</p>	
Referencias	RF-10
Prototipo	Anexo III.11

ANEXO II.11 Descripción del CU "Mostrar las ventas realizadas en el Matadero".

CU #11	Mostrar las ventas realizadas en el Matadero.
Actores	Usuario del sistema.
Propósito	Mostrar las ventas que se han realizado por el Matadero desde un

	vendedor especializado hacia un destino en un mes y año determinado.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener información acerca de las ventas que se han realizado en el Matadero desde un vendedor especializado hacia destino en un mes y año especificado. El Usuario le especifica al sistema el vendedor especializado, mes y el año requerido para el reporte. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.</p>	
Referencias	RF-11
Prototipo	Anexo III.12

ANEXO II.12 Descripción del CU "Mostrar las ventas realizadas en el matadero a MINAL-DIVISA en un mes y año determinado".

CU #12	Mostrar las ventas realizadas en el matadero a MINAL-DIVISA en un mes y año determinado.
Actores	Usuario del sistema.
Propósito	Mostrar las ventas realizadas en el matadero a MINAL-DIVISA en un mes y año determinado.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener información acerca de las ventas realizadas en el matadero a MINAL-DIVISA en un mes y año especificado. El Usuario le especifica al sistema el mes y el año requerido para el reporte. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.</p>	

Referencias	RF-12
Prototipo	Anexo III.13

ANEXO II.13 Descripción del CU "Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento".

CU #13	Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento.
Actores	Usuario del sistema.
Propósito	Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento.
Resumen	
El Caso de uso se inicia cuando el Usuario accede al sistema para obtener una comparación de las ventas realizadas por vendedores y compradores desde el Municipio hasta el momento. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.	
Referencias	RF-13
Prototipo	Anexo III.14

ANEXO II.14 Descripción del CU "Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio".

CU #14	Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio.
---------------	--

Actores	Usuario del sistema.
Propósito	Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio hasta el momento.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener una comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio por vendedores especializados-compradores y vendedores especializados hasta el momento. El caso de uso finaliza cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.</p>	
Referencias	RF-14
Prototipo	Anexo III.15

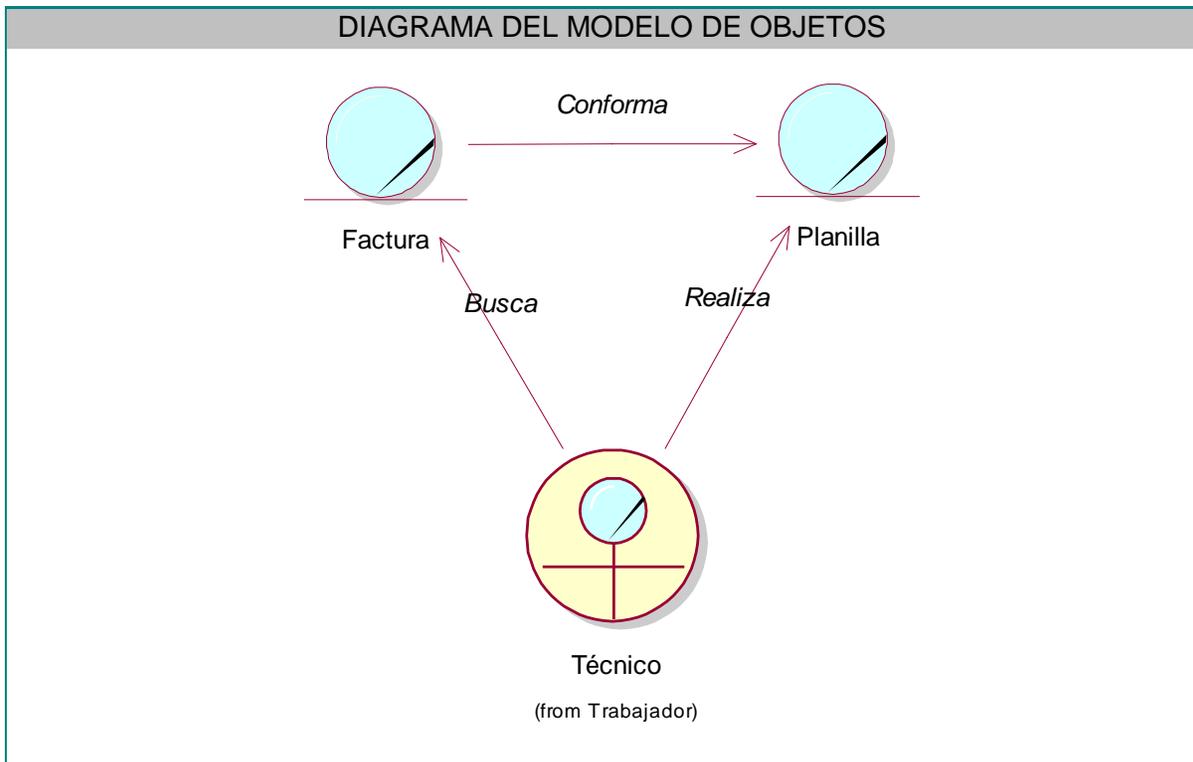
ANEXO II.15 Descripción del CU "Mostrar comparación del mayor importe de venta realizadas".

CU #15	Mostrar comparación del mayor importe de venta realizadas.
Actores	Usuario del sistema.
Propósito	Mostrar comparación del mayor importe de ventas realizadas hasta el momento.
Resumen	
<p>El Caso de uso se inicia cuando el Usuario accede al sistema para obtener una comparación del mayor importe de ventas realizadas por vendedores especializados-compradores y vendedores especializados hasta el momento. El caso de uso finaliza</p>	

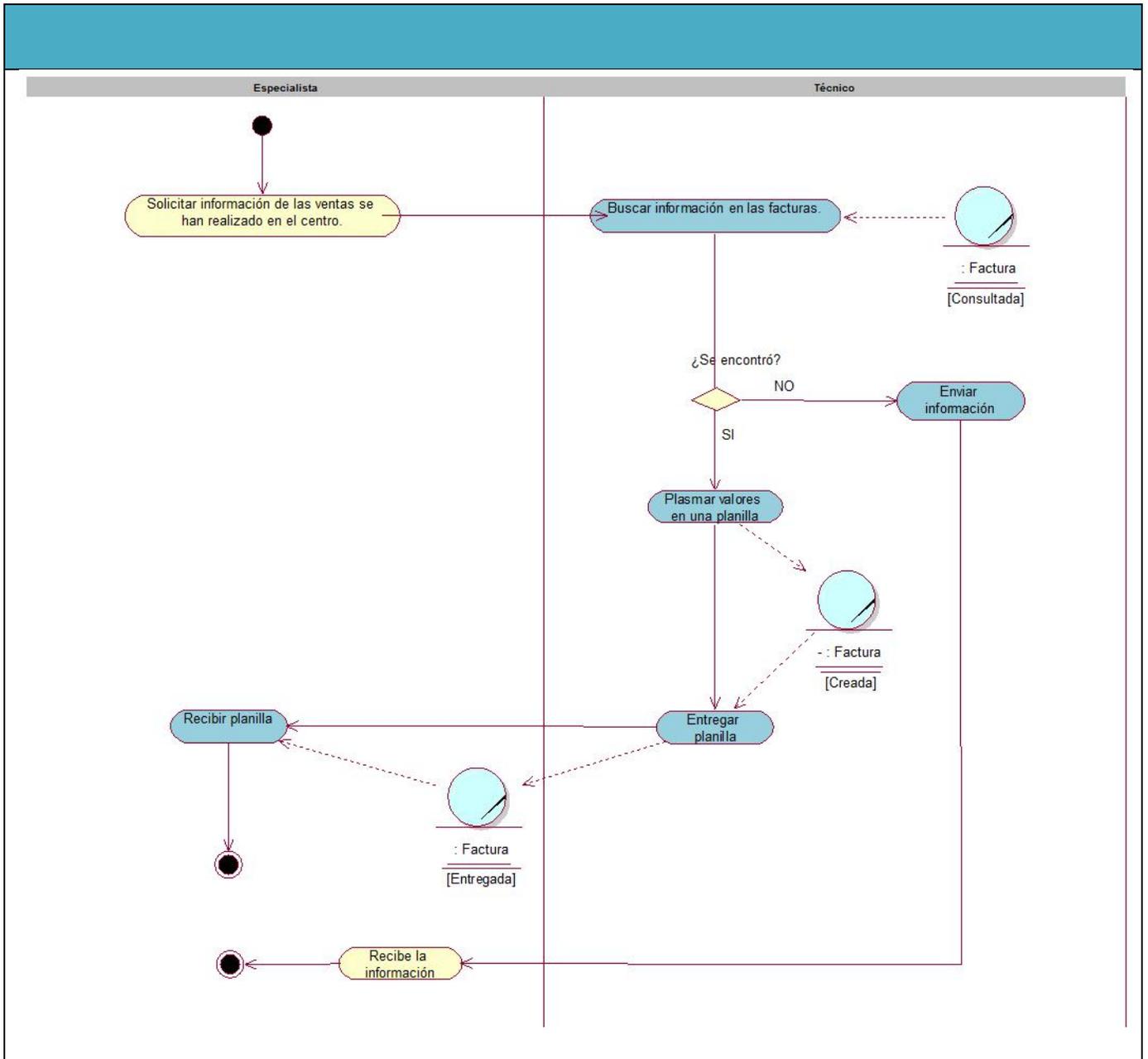
cuando se ejecuta la operación antes descrita y el sistema le da la respuesta al Usuario por medio de una planilla.

Referencias	RF-15
Prototipo	Anexo III.16

ANEXO II.16 Modelo de objetos.

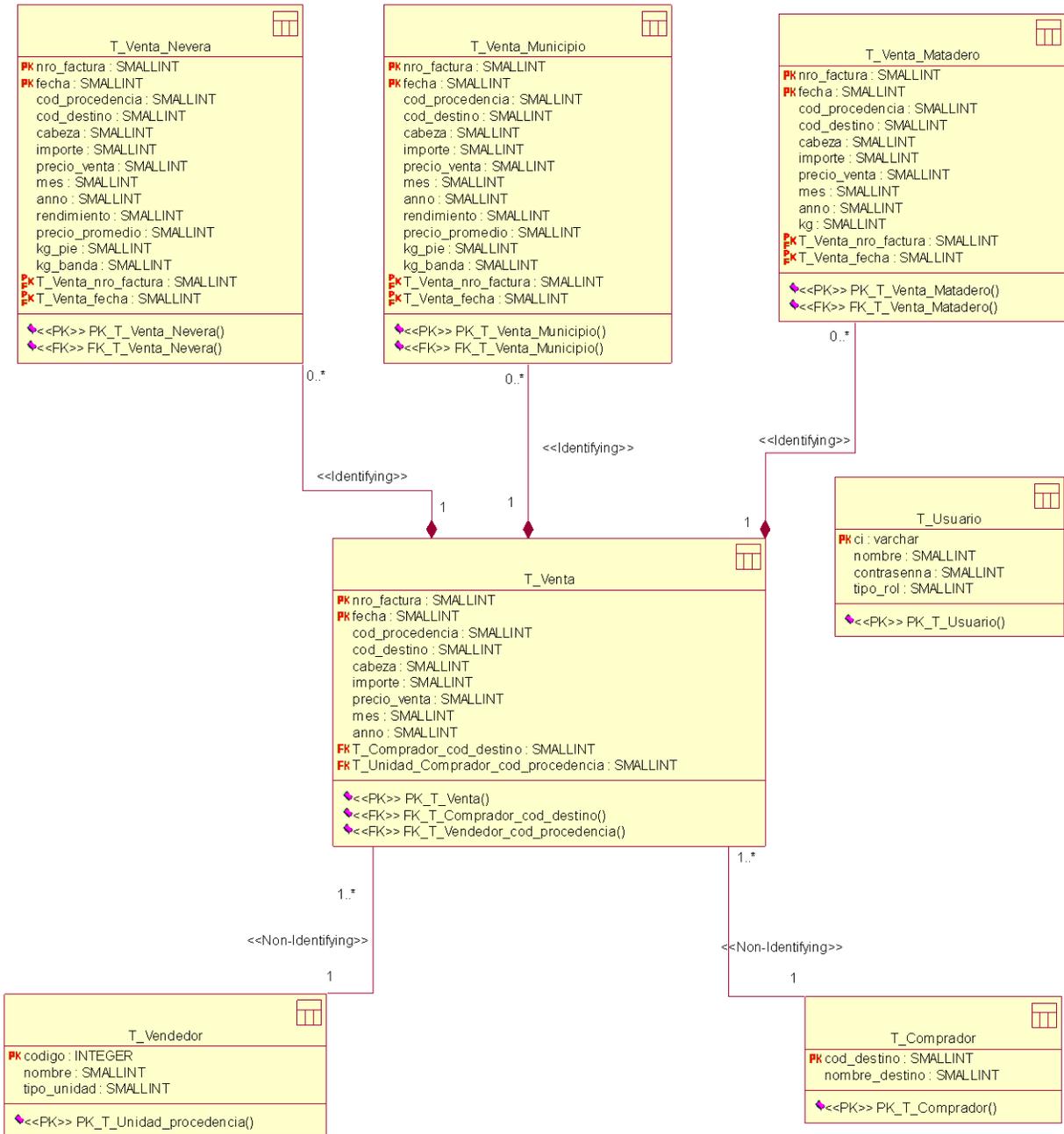


ANEXO II.17 Diagrama de actividad del CU “Solicitar información de las ventas de cerdos realizadas en un mes y año determinado”.



ANEXO III.1 Modelo físico de datos.

MODELO FÍSICO DE DATOS



ANEXO III.2 Prototipo CU “Autenticar usuario”

The screenshot shows a dialog box titled "Autenticación de usuario" with a blue header bar. On the left is a pig icon, and on the right is a document icon. The main area contains the text "PORSOFT" and "AUTENTIFICACIÓN" above the subtitle "SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPIRITUS". Below this is a form with the following fields:

- Usuario: reynel
- Contraseña: [masked]
- Servidor: 127.0.0.1

At the bottom right are two buttons: "Aceptar" (with a green checkmark) and "Cancelar" (with a red prohibition sign).

ANEXO III.3 Prototipo CU “Cambiar contraseña”

The screenshot shows a dialog box titled "Cambiar contraseña" with a blue header bar. On the left is a pig icon, and on the right is a key icon. The main area contains the text "PORSOFT" and "CAMBIO DE CONTRASEÑA" above the subtitle "SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPIRITUS". Below this is a form with the following fields:

- Usuario: reynel
- Contraseña: [masked]
- Contraseña (Nueva): [empty]
- Rectificar Contraseña (Nueva): [empty]

At the bottom right are two buttons: "Aceptar" (with a green checkmark) and "Cancelar" (with a red prohibition sign).

ANEXO III.4 Prototipo CU “Gestionar vendedor”

Gestionar vendedor

PORSOFT ADMINISTRACIÓN

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Vendedor

Código: 63

Nombre: La Sierpe

Tipo de unidad: NE

Vendedor: 1 de 18

Buscar Cancelar

ANEXO III.5 Prototipo CU “Gestionar comprador”

Gestionar comprador

PORSOFT ADMINISTRACIÓN

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Comprador

Código: 2

Nombre: MINAL-DIVISA

Comprador: 1 de 10

Buscar Cancelar

ANEXO III.6 Prototipo CU “Gestionar usuario”

Gestionar usuario

PORSOFT ADMINISTRACIÓN

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Usuario

CI: 91041698764

Nombre: reydel

Contraseña: •

Tipo de rol: Administrador

Usuario: 1 de 8

Buscar Cancelar

ANEXO III.7 Prototipo CU “Gestionar venta por Matadero”

Gestionar venta por Matadero

PORSOFT GESTIÓN

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Venta por Matadero

Fecha: 12-01-2013

Vendedor: Cacahual

Nro de factura: 1234

Comprador: Mercado Agropecuario

Cabezas: 24

Mes: Febrero

Importe(\$): 5.0

Año: 2013

Precio de venta(\$): 7.0

Kg en pie: 56

Venta: 1 de 20

Buscar Cancelar

ANEXO III.8 Prototipo CU “Gestionar venta por Nevera”

Gestionar venta por Nevera

PORSOFT **GESTIÓN**

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Venta por Nevera

Fecha: 01-03-2012

Nro de factura: 504060

Cabezas: 3

Importe(\$): 264.48

Kg en banda: 152.0

Vendedor: Comercializador 1

Comprador: Comedor

Mes: Marzo

Año: 2012

Kg en pie: 244

Indicadores de calidad

Peso promedio: 81.0

Rendimiento(%): 62.3

Precio de venta(\$): 1.74

Venta: 1 de 21

Buscar Cancelar

ANEXO III.9 Prototipo CU “Gestionar venta por Municipio”

Gestionar venta por Municipio

PORSOFT **GESTIÓN**

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Venta por Municipio

Fecha: 05-06-2007

Nro de factura: 2

Cabezas: 12

Importe(\$): 2.36

Kg en banda: 2.365

Vendedor: Carbó

Comprador: Mercado Agropecuario

Mes: Enero

Año: 2012

Kg en pie: 232

Indicadores de calidad

Peso promedio: 19.0

Rendimiento(%): 1.0193

Precio de venta(\$): 0.99788

Venta: 1 de 19

Buscar Cancelar

ANEXO III.10 Prototipo CU “Mostrar las ventas realizadas por cada municipio a Comercio y Gastronomía en un mes y año determinado.”

Mostrar ventas de los municipios hacia Comercio y Gastronomía

100%

LISTADO DE LAS VENTAS REALIZADAS

Municipio: Cabaiguán **Destino:** Comercio y Gastronomía

Mes: Marzo

Año: 2013

Fecha	Número de factura	Cabezas	Importe(\$)	Precio de venta(\$)	Kg en pie	Kg en banda	Peso promedio	Rendimiento(%)
23/06/2005	50	12	4393,00	14,64	200	300,00	16,00	150,00
06/03/2012	5640067	4	5327,00	30,44	282	175,00	70,00	62,06
18/11/2003	75	23	283000,00	832,35	300	340,00	13,00	113,33
Totales		39	292720,0	877.4300000000001	782	815,0	99,0	325,39

28/02/2013 Página 1 de 1

Pagina 1 de 1

ANEXO III.11 Prototipo CU “Mostrar las ventas realizadas en la nevera hacia Venta a trabajadores en un mes y año determinado.”

Mostrar ventas de la Nevera hacia Venta a trabajadores

PORSOFT REPORTES

SOFTWARE PARA EL CONTROL DE LA MASA PORCINA EN LA EMPRESA PROVINCIAL PORCINA EN SANCTI SPÍRITUS

Seleccione el mes y el año a realizar la búsqueda

Mes: Enero Año: 2013

Mostrar

Mostrar ventas de la Nevera hacia Venta a trabajadores

EMPRESA PORCINA SANCTI SPÍRITUS
Elevadas producciones con garantía de calidad

LISTADO DE LAS VENTAS REALIZADAS

Lugar de comercialización: Nevera Destino: Venta a trabajadores
Mes: Marzo
Año: 2012

Fecha	Número de factura	Cabezas	Importe(\$)	Precio de venta(\$)	Kg en pie	Kg en banda	Peso promedio	Rendimiento(%)
06/03/2012	504066	8	142,68	3,48	82	41,00	10,00	50,00
08/03/2012	504071	7	1404,00	4,50	469	312,00	67,00	66,52
05/03/2012	504064	2	1333,50	10,50	178	127,00	89,00	71,35
Totales		17	2880,18	18,48	729	480,00	166,00	187,87

28/02/2013 Página 1 de 1

Página 1 de 1

ANEXO III.12 Prototipo CU “Mostrar las ventas realizadas en las unidades especializadas hacia MINAL-DIVISA desde el Matadero en un mes y año determinado.”

LISTADO DE LAS VENTAS REALIZADAS

Unidad especializada: El Colorado
 Mes: Enero
 Año: 2013

Destino: MINAL-DIVISA

Fecha	Número de factura	Cabezas	Importe(\$)	Precio de venta(\$)	Kg en pie
15/01/2017	34	22	5,00	4,00	3
05/09/2008	5441206	2	56326,54	112,65	256
01/03/2012	5541052	14	42788,16	14,32	2988
Totales		38	99119,70	130,97	3247

28/02/2013 Página 1 de 1

ANEXO III.13 Prototipo CU “Mostrar las ventas realizadas al MINAL-DIVISA en un mes y año determinado.”

LISTADO DE LAS VENTAS REALIZADAS

Mes: Enero
Año: 2013

Destino: MINAL-DIVISA

Fecha	Número de factura	Cabezas	Importe(\$)	Precio de venta(\$)
01/03/2012	5541052	14	42788,16	14,32
05/09/2008	5441206	2	56326,54	112,65
15/01/2017	34	22	5,00	4,00
12/01/2013	5541053	46	29892,05	10,16
Totales		84	129011,75	141,13

28/02/2013

Página 1 de 1

ANEXO III.14 Prototipo CU “Mostrar una comparación de las ventas realizadas por vendedores y compradores desde Municipio hasta el momento.”

Análisis económico de las ventas realizadas por vendedores y compradores desde Municipio

100%

ANÁLISIS ECONOMICO ENTRE LAS VENTAS REALIZADAS POR VENDEDOR-COMPRADOR DESDE MUNICIPIO



Vendedor	Comprador	Cabezas	Importe(\$)	Kg en pie	Kg en banda	Rendimiento(%)	Peso promedio
Trinidad	Comercio y Gastronomía	6	321,57	120	12,36	1,89	109,00
Tamarindo	Mercado Agropecuario	8	800,00	100	200,00	200,00	12,00
Carbó	Aseguramiento MINAGRI	2	3865,88	188	127,00	67,55	94,00
Cabaiguán	Comercio y Gastronomía	39	292720,00	782	815,00	325,39	99,00
Comercializador 4	MINAL-DIVISA	6	24075,60	1170	790,92	67,60	195,00
El Colorado	MINAL-DIVISA	6	123,56	653	452,65	69,32	108,00
Cacahual	Mercado Agropecuario	12	800,66	132	350,00	265,15	11,00
Sancti SAP	Mercado Agropecuario	4	7244,72	355	238,00	67,04	88,00
Los Molinos	Venta a trabajadores	2	3500,60	115	171,00	148,70	57,00
Carbó	Mercado Agropecuario	12	2,36	232	2,36	1,02	19,00
Totales		97	333454,95	3847	3159,29	1213,66	792,00

28/02/2013

Página 1 de 1

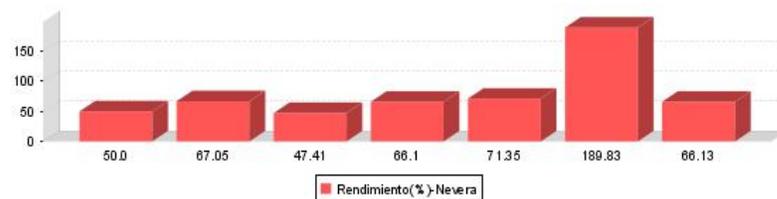
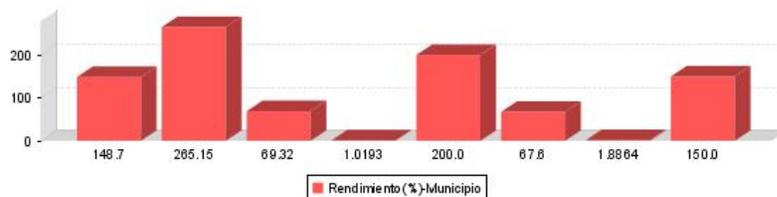
Página 1 de 1

ANEXO III.15 Prototipo CU “Mostrar comparación del mayor rendimiento de ventas realizadas en la Nevera y el Municipio por Vendedores Especializados.”

ANALISIS ECONOMICO DEL RENDIMIENTO DE LAS VENTAS POR VENDEDORES ESPECIALIZADOS REALIZADAS DESDE NEVERA Y MUNICIPIO



Vendedor	Rendimiento(Venta por Municipio)	Rendimiento(Venta por Nevera)
Los Molinos	148,70	50,00
Cacahual	265,15	67,05
El Colorado	69,32	47,41
Carbó	1,02	67,05
Tamarindo	200,00	66,10
Comercializador 4	67,60	71,35
Comercializador 1	1,89	189,83
Venegas	150,00	66,13



ANEXO III.16 Prototipo CU “Mostrar comparación del mayor importe de ventas realizadas por Vendedores Especializados.”

