

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**Facultad 2**



**Sistema de gestión de información sobre las ventas  
por suscripciones de los productos de la cartera de  
Desoft**

**Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas**

**Autor:** Anabel Figueroa García

**Tutores:** Dra. Ana María García Pérez

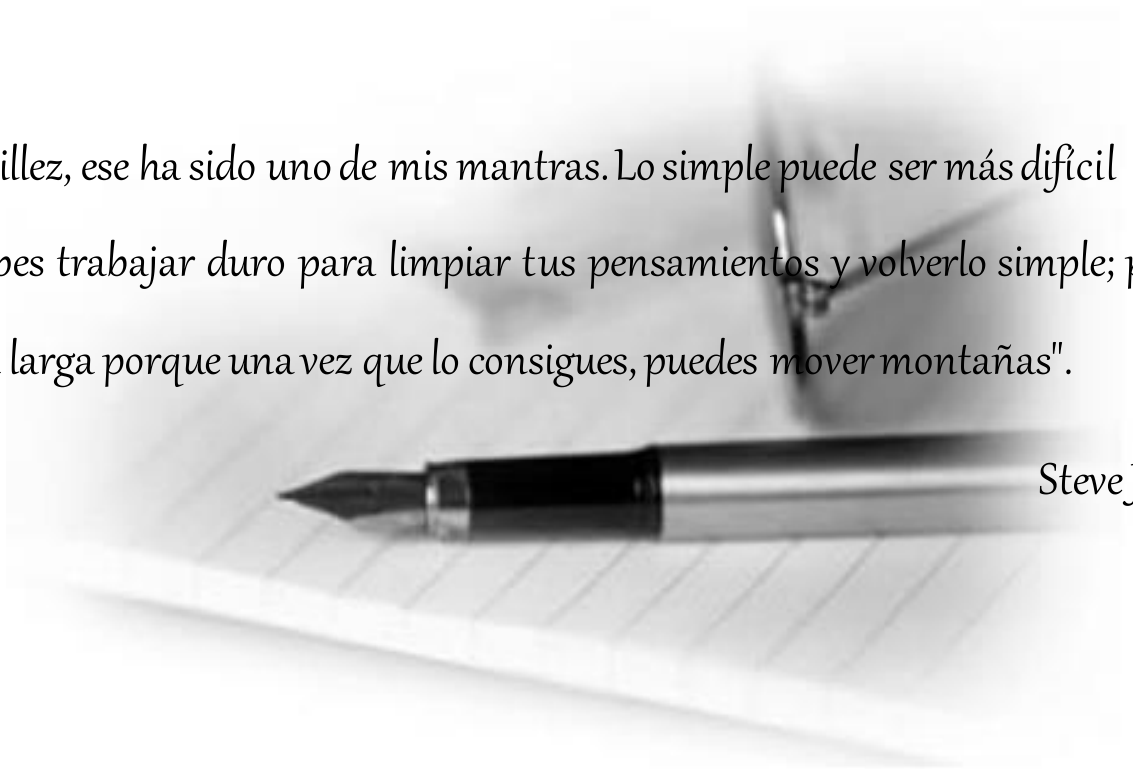
Ing. Pavel Mena Nodal

**La Habana, 2016**

**“Año 58 de la Revolución”**

"Enfoque y sencillez, ese ha sido uno de mis mantras. Lo simple puede ser más difícil que lo complejo: debes trabajar duro para limpiar tus pensamientos y volverlo simple; pero vale la pena a la larga porque una vez que lo consigues, puedes mover montañas".

Steve Jobs



## Declaración de autoría

---

Declaro ser única autora del trabajo **Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft** y autorizo a la Universidad de las Ciencias Informáticas hacer el uso que estimen pertinente con el mismo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año 2016.

---

Anabel Figueroa García

Firma del Autor

---

Dra. Ana María García Pérez

Firma del Tutor

---

Ing. Pavel Mena Nodal

Firma del Tutor

## Datos de contacto

---

### Tutor

**Nombre y apellidos:** Ana María García Pérez.

**Centro de trabajo:** UEB División Desoft Villa Clara

**Cargo:** Sub Directora de Innovación y Desarrollo

**Título de la especialidad de graduado:** Licenciada en Cibernética-Matemática

**Año de graduación:** 1982

**Institución en la que se graduó:** Universidad Central "Marta Abreu" de Las Villas

**Correo electrónico:** anamaria.garcia@vcl.desoft.cu

### Tutor

**Nombre y apellidos:** Pavel Mena Nodal.

**Centro de trabajo:** Universidad de las Ciencias Informáticas.

**Cargo:** Desarrollador.

**Título de la especialidad de graduado:** Ingeniero en Ciencias Informáticas.

**Año de graduación:** 2014.

**Institución en la que se graduó:** Universidad de las Ciencias Informáticas.

**Correo electrónico:** pmena@uci.cu

## Agradecimientos

---

Gracias, de corazón, a mis tutores, la Dra. Ana María García y al Ing. Pavel Mena Nodal por guiarme en esta gran tarea, darme todo su apoyo y si hoy me encuentro aquí es gracias a ustedes. Les estaré eternamente agradecida.

A mis padres por enseñarme que las metas son alcanzables y que una caída no es una derrota sino el principio de una lucha que siempre termina en Logros y Éxitos. Gracias por siempre orientarme en todo lo que se y ayudarme a salir adelante a pesar de los inconvenientes. Este triunfo también es de ustedes. Mami y Papi los quiero.

A mis abuelitos queridos por estar siempre pendientes de mí y por ser este logro su mejor regalo.

A mis dos hermanos por ser mi guía a seguir, por estar siempre presentes para mí, ayudarme y apoyarme en mis decisiones.

A toda mi familia por su guía, amor y apoyo incondicional en cada momento de mi vida y los valores que han sembrado en mí.

A mi novio por permanecer a mi lado en estos 3 años, 9 meses y 1 día, soportándome cuando más difícil era y por mostrarme que existe una forma más pura de amar.

A la familia de mi novio, en especial a sus padres y su hermana por acogerme en su familia y por el afecto que me han dado.

A todos los profesores que me acompañaron en toda mi vida como estudiante, a mi oponente y miembros del tribunal, por su ayuda y buenos consejos; gracias a ellos hoy he podido superar una de mis mayores metas.

## Agradecimientos

---

A mis compañeros de grupo, por ser mis salvadores en muchísimas ocasiones y hacer más  
amenas las clases.

A todas mis amistades por ayudarme en la realización de esta tarea y estar siempre ahí  
cuando acudo a ellos: Yudenia, Yenisel, Enier, Marisbelsy, gracias por dedicar su tiempo  
para que todo saliera de la mejor manera posible.

A todas las personas que conocí a lo largo de mi carrera en la universidad, todo el grupo de  
amistades de mi novio y mías por compartir buenos momentos de fiestas: Yudenia, Batista,  
la Pili, Enier, Ana, Raine, Lorena, Wilito, Yuset (Mon Negron) a mis compañeros de  
apartamento: el Pina y Arianna, a las personas que no están presentes pero que igualmente  
ocupan un lugar muy especial en mi corazón y que ya son parte de mi vida, muchas gracias  
por permitirme conocerlos.

Sepan todos lo que no he mencionado que NO los olvidé, simplemente resulta imposible  
nombrar a tantas personas especiales ya que no podría acabar nunca. No obstante, siempre  
estarán en mi corazón.

Muchas gracias a todos.

## Dedicatoria

---

Para triunfar en la vida no es importante llegar primero, para triunfar simplemente hay que llegar. Al culminar unos de mis objetivos dedico la presente tesis de grado a:

Mis padres, que hicieron todo en la vida para que yo pudiera lograr mis sueños, por motivarme y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Mi novio, porque sin su ayuda este trabajo hubiera sido mucho más largo, complicado y menos rico, por ser parte de mi vida, compartir los buenos y malos momentos y ayudarme a seguir siempre adelante.

Hoy retribuyo parte de sus esfuerzos con este logro que no es mío sino de ustedes, por lo cual viviré eternamente agradecida.

## Resumen

---

La Empresa de Aplicaciones Informáticas Desoft, tiene como misión desarrollar y comercializar productos y servicios informáticos integrales asociados a las tecnologías de la información, aportando al desarrollo sostenible de la sociedad cubana. Actualmente Desoft se encuentra en proceso de perfeccionamiento empresarial y como parte de este proceso ha cambiado su modelo de negocios hacia el “software como servicio” que se basa en la venta de suscripciones a las aplicaciones, centrándose en brindar un servicio de mejora y capacitación continua al cliente en el uso del producto al que éste se suscribe. Sin embargo, en este modelo aumenta la cantidad de información que se genera en cada venta, y al no existir un sistema automatizado que permita controlar esta información, se pierde tiempo y cometen errores, retrasando el proceso de otorgamiento de las licencias. El objetivo del presente trabajo es desarrollar un sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft, que posibilite el manejo de información de las transferencias entre divisiones que se derivan de estas ventas y automatiza el proceso de generación y entrega de licencias. Los principales resultados de la investigación se relacionan con el estudio de sistemas y tecnologías existentes que pudieran cubrir las expectativas y requisitos de la empresa, la elaboración de los artefactos según la metodología Open Up, la implementación del sistema y su validación en la división Desoft de Villa Clara.

**Palabras clave:** comercializar, sistema de gestión de información, suscripciones, software como servicio.



## Abstract

---

Desoft Enterprise has the mission of developing computer products and offering integral services associated with information technology, contributing to the sustainable development of Cuban society. Currently Desoft is in the process of business improvement and as a part of this process it has changed its business model to "software as a service", which is based on selling subscriptions to applications, focusing on providing a service improvement and continuous training to customer for the use of the product to which it subscribes. However, in this model the amount of information generated on each sale increases, and due to the absence of an automated system to monitor this information, time is wasted and humans make mistakes, producing delaying in the process of granting licenses. The objective of this work is to develop a management information system about sales subscription for the product portfolio of Desoft, enabling management information of transfers derived from these sales between divisions, and automating the process of generating and delivering licenses. The main results of research are related to the study of systems and technologies that could meet the expectations and requirements of the company, the development of software artefacts using the methodology Open Up, and system implementation and its validation in the Desoft Division Villa Clara.

**Keywords:** information management system, marketing, software as a service, subscriptions.

# Índice

---

Introducción.....	1
Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones .....	6
1.1 Conceptos asociados al dominio del problema.....	6
1.2 Soluciones informáticas sobre la gestión de información de las ventas por suscripciones .....	8
1.2.1 Sistemas CRM (Gestión de las relaciones con el cliente).....	8
1.2.2 Sistemas ERP (Planificación de Recursos Empresariales) .....	9
1.2.3 Odoos .....	9
1.2.4 Comercializador de la OMPI .....	10
1.2.5 Comercializador de CSI S.A.....	10
1.2.6 Comercializador de Datazucar.....	10
1.3 La gestión de información sobre las ventas por suscripciones en Desoft.....	11
1.4 Análisis de las soluciones anteriores sobre la gestión de información de las ventas por suscripciones.....	12
1.5 Herramientas y tecnologías para el desarrollo del sistema .....	13
1.5.1 Metodologías de desarrollo de Software .....	13
1.5.2 Lenguaje de modelado UML .....	14
1.5.3 Herramientas CASE .....	15
1.5.4 Notación de modelado de Procesos de negocio BPMN .....	15
1.5.5 CMS (Sistema de Gestión de Contenidos) Drupal 7.26 .....	15
1.5.6 SGBD (Sistema Gestor de Base de Datos) PostgreSQL 9.4.....	16
1.5.7 PgAdmin 3 .....	17
1.5.8 Desarrollo web.....	17
1.5.9 Lenguaje de programación al lado del cliente .....	17
1.5.10 Lenguaje de programación del lado del servidor .....	18
1.5.11 IDE (Entorno de Desarrollo Integrado) .....	18
1.5.12 SVN (Subversion).....	19

# Índice

---

1.6 Conclusiones del capítulo .....	19
Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.....	21
2.1 Propuesta del sistema .....	21
2.2 Modelo de negocio.....	22
2.2.1 Descripción del negocio .....	22
2.3 Modelo conceptual .....	23
2.3.1 Descripción de los conceptos que intervienen en el dominio del problema .....	24
2.4 Modelo del sistema .....	24
2.4.1 Requisitos funcionales .....	25
2.4.2 Requisitos no funcionales .....	30
2.4.3 Definición de los actores del sistema.....	31
2.4.4 Diagrama de casos de uso del sistema.....	32
2.4.5 Descripción de casos de uso del sistema.....	32
2.5 Modelo de diseño .....	39
2.5.1 Descripción de los patrones arquitectónicos y de diseño .....	39
2.5.2 Patrones arquitectónicos .....	39
2.5.3 Patrones de diseño.....	40
2.6 Diagrama de clases de diseño .....	42
2.7 Diagramas de interacción .....	43
2.7.1 Diagrama de secuencia.....	43
2.8 Modelo de datos.....	44
2.9 Conclusiones del capítulo .....	45
Capítulo 3: Implementación, pruebas y resumen del Manual de Usuario .....	46
3.1 Modelo de implementación .....	46
3.1.1 Diagrama de componentes .....	46

# Índice

---

3.3.2 Modelo de despliegue .....	47
3.2 Modelo de pruebas .....	48
3.2.1 Pruebas de caja negra .....	48
3.2.2 Pruebas de rendimiento .....	52
3.3 Manual de Usuario .....	52
3.4 Validación del sistema en Desoft Villa Clara .....	55
3.5 Conclusiones del capítulo .....	56
Conclusiones generales .....	57
Recomendaciones .....	58
Referencias bibliográficas.....	59
Bibliografía .....	63

## Índice de Tablas

<b>Tabla 1: Definición de los actores del sistema .....</b>	<b>32</b>
<b>Tabla 2: Descripción del caso de uso "Gestionar Cliente" .....</b>	<b>33</b>
<b>Tabla 3: Diseño de caso de prueba para el CU Gestionar Cliente.....</b>	<b>48</b>
<b>Tabla 4: Descripción de variables.....</b>	<b>50</b>
<b>Tabla 5: Matriz de datos para el escenario adicionar cliente .....</b>	<b>50</b>

### Índice de Figuras

Figura 1. Diagrama de venta por suscripción de un producto de la cartera de Desoft .....	23
Figura 2: Modelo Conceptual .....	24
Figura 3: Diagrama de Casos de Uso del Sistema .....	32
Figura 4: Patrón arquitectónico Modelo-Vista-Controlador .....	40
Figura 5: DCD del CU Gestionar cliente.....	43
Figura 6: Diagrama de secuencia del CU Gestionar cliente: Escenario adicionar cliente .....	44
Figura 7: Modelo de datos del Sistema de información sobre las ventas de los productos de la cartera de Desoft.....	45
Figura 8: Diagrama de componentes .....	47
Figura 9: Diagrama de despliegue.....	47
Figura 10: Prueba de caja negra 1era iteración .....	51
Figura 12: Página Principal.....	52
Figura 13: Menú de gestión.....	53
Figura 14: Gestionar Cliente.....	53
Figura 15: Reporte de deudas.....	54
Figura 16: Vista de Pagos .....	54
Figura 17: Generación de Licencias.....	55

## Introducción

Cuba se encuentra inmersa en la informatización de la sociedad, utilizando de forma ordenada y masiva las Tecnologías de la Información y las Comunicaciones (TIC's), contribuyendo a la satisfacción de las necesidades de múltiples esferas. Ejemplo de ello, es la creación del Ministerio de Comunicaciones (MINCOM). Este, junto con el Grupo Empresarial de la Informática y las Comunicaciones (GEIC), tienen como objetivo impulsar, facilitar y ordenar el uso masivo de servicios y productos de las TIC, para mejorar la calidad de vida de la población y como fuente de riquezas al país (1).

Entre las unidades subordinadas al GEIC se encuentra la Empresa de Aplicaciones Informáticas Desoft, que tiene como misión: desarrollar y comercializar productos y servicios informáticos integrales asociados a las tecnologías de la información, aportando al desarrollo sostenible de la sociedad. (2) Esta organización se encuentra actualmente en proceso de perfeccionamiento empresarial, lo cual significa que está determinada a diseñar un sistema integrado de gestión empresarial<sup>1</sup>, donde la gestión comercial<sup>2</sup> es uno de los procesos operacionales clave, el cual debe garantizar la mejora continua de la satisfacción de sus clientes.

Desoft, está representada por divisiones en todas las provincias de Cuba y el municipio especial de la Isla de la Juventud. Cada división de la empresa ha desarrollado para su comercialización uno o varios productos, que no son más que aplicaciones informáticas pre-empaquetadas que se configuran y adaptan a los clientes. Estas aplicaciones son conocidas como “los productos de la cartera de Desoft”, y se dedican parte de las utilidades a la actualización constante de los productos. Cada división no solo comercializa los productos propios que ella mantiene, sino además los productos de las restantes divisiones.

Existen dos maneras de comercializar estos productos de software, la primera es vendiendo licencias, las cuales se cobran en su totalidad antes de desplegar el producto. Una licencia otorga permiso de uso del producto por tiempo indefinido y la segunda forma es vendiendo suscripciones, que no son más que derechos de usar el producto por un tiempo determinado. Durante el tiempo pactado, el

---

<sup>1</sup> Gestión empresarial: medidas y estrategias llevadas a cabo con la finalidad de que la empresa sea viable económicamente. (59).

<sup>2</sup> Gestión comercial: gestión de las relaciones de una empresa con sus clientes, para regir la manera en que la empresa se dirige y se enfoca en sus clientes. (57)

cliente usa el producto con las actualizaciones que se le vayan realizando por el equipo que lo mantiene.

Desoft ha decidido cambiar su modelo de negocios hacia la venta por suscripciones para los productos de su cartera. Con ello pretende afianzar el modelo de “software como servicio”, la suscripción enfatiza la necesidad de brindar un servicio de mejora y capacitación continua al cliente en el uso del producto, que es el aceptado por el mercado. Cuando el cliente se suscribe y el producto se instala en la infraestructura física del cliente, se le entrega una licencia para impedir las copias ilegales del producto. Para la mejora de este proceso, está prevista una estrategia de Desoft para migrar, en la medida de lo posible, los servicios hacia la nube, una infraestructura tecnológica que se está adquiriendo para que el servicio de uso del producto se produzca en base a una real suscripción y no a una instalación en la infraestructura física del cliente.

Cuando en una división se produce la venta de un producto mantenido por otra división, se estipula que un por ciento del monto ingresado se transfiera hacia la división dueña del producto. Este monto transferido es la forma de financiar parte de los gastos en que incurre la división dueña en las actualizaciones que desarrolla con equipos de trabajo dedicados a esta actividad.

Para realizar estas transferencias se confecciona hoy una hoja Excel, denominada “Reporte de Ventas de Licencias de Otros” (VLO). Este reporte almacena diversos datos con información sobre las ventas de los productos, a saber: el cliente, el producto, la fecha de la venta, el ingreso obtenido y el por ciento a transferir. Toda esta información se controla de forma manual en carpetas compartidas, ocasionando inexactitudes, además de que tiene dispersos los datos que necesitan ser analizados.

La división dueña del producto es la encargada de generar las licencias a partir de las solicitudes del grupo de despliegue de los productos. Para ello el gestor comercial encargado de generar esa licencia sólo procede a entregar la licencia si existe al menos una transferencia efectuada desde la división que vendió el producto hacia la división dueña; si no existe tal transferencia no se entrega la licencia. Debido a fuertes compromisos este paso a veces se viola y como consecuencia, la división dueña pierde dinero, así como también sucede que un cliente de cualquier división deja de pagar el monto pactado y es difícil de detectar, y en general los tiempos de espera por estos procedimientos de comprobación se alargan.

Con el modelo de venta de licencias, las transferencias entre divisiones se realizaban una sola vez por cada venta de producto, sin embargo, a partir del esquema de suscripciones, el cliente puede pagar en plazos y por ende los reportes VLO deben contener no solo la información de una venta nueva sino la información de todos los ingresos obtenidos mediante los pagos a plazos definidos con el cliente en el contrato. Esta problemática aumenta la cantidad de información a controlar y también aumentan los tiempos de espera en la obtención de las licencias, que son necesarias para lograr instalar el producto, donde la búsqueda de las transferencias hay que hacerla de forma manual. Todo ello significa que la



información para la toma de decisiones se vuelve difícil de obtener y, la empresa en su totalidad puede perder recursos y, sobre todo, tiempo.

A partir de la situación problemática descrita se plantea el siguiente **problema**:

El cambio de modelo de negocios de venta de licencias hacia suscripciones ha aumentado la cantidad de información que se debe controlar en Desoft, afectando el proceso de entrega de licencias para la implantación de los productos de la cartera en los clientes y de transferencias entre divisiones.

El problema definido se enmarca en el siguiente **objeto de estudio**: modelo de negocio de ventas por suscripciones.

Se define como **objetivo general** desarrollar un sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft, que posibilite el manejo de información de los estados de las suscripciones con los clientes y las transferencias entre divisiones que se derivan de estas ventas, y automatice el proceso de generación y entrega de licencias, delimitando como **campo de acción**: sistema de gestión de información sobre las ventas por suscripciones.

Un sistema de gestión de información que logre producir automáticamente los reportes VLO a partir de las ventas de productos ajenos, genere las licencias y las envíe a los contactos pertinentes después de que se haya producido al menos una transferencia hacia la división dueña del producto, y emita reportes variados para controlar los estados de las transferencias, reportará beneficios, pues ahorrará tiempo y recursos que se deben poner en función de la gestión de ventas en sí, o sea, de la labor directa de gestión con los clientes.

Las **preguntas de investigación** que se formulan al respecto de la solución a la problemática son:

1. ¿Cuáles son los principales fundamentos teóricos y metodológicos acerca de la gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft?
2. ¿Qué componentes tendrá la aplicación informática que contribuya a la gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft?
3. ¿Qué resultados tendrá en la práctica la aplicación informática desarrollada?

Para dar cumplimiento al objetivo anteriormente planteado se definen las siguientes **tareas de la investigación**:

- Análisis de las características y limitaciones de las soluciones existentes referentes a los sistemas de información de ventas, decidiendo si construir o aprovechar algunas de éstas.
- Caracterización de las herramientas y tecnologías que son adecuadas para la elaboración del sistema.
- Elaboración de los artefactos y la documentación según la metodología de desarrollo seleccionada, para lograr documentar la aplicación que se propone.

- Implementación del sistema para la gestión de información de ventas y entregas de licencias de los productos de la cartera de Desoft.
- Validación del sistema propuesto a partir de la realización de pruebas con Desoft.

Para desarrollar la investigación se utilizan los siguientes **métodos de investigación científica**:

### **Métodos teóricos**

Análisis histórico-lógico: Este método se empleó para la realización de un estudio tanto a nivel internacional como nacional de los diferentes sistemas creados hasta el momento que realizan funcionalidades para la gestión de la información sobre las ventas por suscripciones en empresas de software, así como la evolución de los mismos y sus principales características. (3)

Análisis sintético: Este método permitió analizar el proceso de la gestión de la información sobre las ventas por suscripciones en empresas de software, logrando sintetizar la información adquirida durante el análisis para el entendimiento del funcionamiento de este proceso. (3)

Inductivo-Deductivo: Se utilizó para el planteamiento del objetivo y la extracción de las ideas fundamentales para la elaboración y fundamentación del trabajo de diploma. Se emplea para la identificación de la problemática y de las soluciones. (4)

Modelación: Este método se utilizó en los procesos del negocio, para comprender y describir los principales procesos de la gestión de la información sobre las ventas por suscripciones en la empresa de software Desoft. Se utiliza en el diseño del sistema mediante el esbozo de los diferentes diagramas definidos en la metodología seleccionada, permitiendo así un mejor entendimiento de las funcionalidades de la aplicación. (3)

### **Métodos empíricos**

Entrevista: La entrevista es una técnica de recopilación de información mediante una conversación profesional. Esta técnica permite adquirir información acerca de lo que se investiga y depende en gran medida del nivel de comunicación entre el investigador y los participantes en la misma. (3). Se utilizó con el objetivo de conocer la estructura organizativa del proceso de gestión comercial de la empresa de desarrollo de software Desoft, Villa Clara. Se le realiza la entrevista a la Sub Directora de Innovación y Desarrollo de la empresa. Ver anexo 1.

Observación: Se utilizó para obtener una información más precisa de cómo se realiza el proceso de gestión comercial de la empresa de desarrollo de software Desoft.

Este trabajo de diploma se encuentra estructurado de la siguiente forma:

El **Capítulo 1** describe las características inherentes al proceso de la gestión de información sobre las ventas por suscripciones en la empresa de software Desoft, analizando conceptos, herramientas y tecnologías útiles en la solución de la problemática planteada, se describen también las características

y limitaciones de las soluciones existentes concluyendo que se debe desarrollar una aplicación a la medida, y se caracterizan las herramientas y tecnologías adecuadas para la elaboración del sistema.

En el **Capítulo 2** se documentan los principales artefactos de la solución que se diseñó para la gestión de la información de ventas por suscripciones y entrega de licencias de los productos de la cartera de Desoft, incluidos artefactos de pruebas.

El **Capítulo 3** muestra un conjunto de artefactos como el diagrama de componentes, además se describen las pruebas realizadas al sistema y los resultados alcanzados. En dicho capítulo también se describe el Manual de Usuario de la aplicación, tal y como quedó implementada en su versión 1.0

# **Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones**

El presente capítulo tiene por objetivo describir las características inherentes al proceso de la gestión de información de ventas en instituciones diversas. Analizando soluciones informáticas empleadas en el mundo, específicamente para obtener un control automatizado sobre la información de ventas y entregas de licencias de productos de software. Se describen también las características y limitaciones de las soluciones existentes concluyendo que se debe desarrollar una aplicación a la medida para lo cual se exponen varias tecnologías y herramientas adecuadas.

Al final del capítulo se justifican las herramientas seleccionadas para construir una aplicación que apoye la gestión de la información de ventas por suscripciones y entrega de licencias de los productos de la cartera de Desoft, de acuerdo con la situación actual de la empresa.

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

Para una mejor comprensión de lo planteado en la situación problemática es importante describir algunos conceptos que se utilizan en la investigación.

### **Suscripción**

Suscripción es el “acto y el resultado de suscribirse o de suscribir”. Este verbo, por su parte, puede emplearse para nombrar a la decisión de una persona de registrarse o abonarse a algún tipo de servicio. (5)

Puede entenderse la suscripción como un modelo de comercialización que consiste en la distribución o entrega de ciertos productos de forma periódica, a cambio del pago de un abono mensual o anual. La suscripción, de este modo, es una especie de venta que se repite periódicamente. (6)

Las principales redes sociales usualmente recurren a un tipo de estrategia de comercialización en que el usuario no está obligado a adquirir ni a instalar el software, sino que se limita a iniciar sesión y a utilizarlo directamente en Internet. Empresas como Microsoft y Adobe también han optado por poner a la venta por suscripción un gran número de aplicaciones informáticas, un modelo que goza de creciente aceptación en la industria del software. Transcurrido un período de prueba gratuito, los usuarios deben pagar una cuota mensual si desean seguir utilizando el software, pero el código fuente escrito para crear la aplicación sigue siendo confidencial. (7)

En la presente investigación la suscripción consiste en que el cliente pagará un determinado monto mensual o anual, en dependencia del periodo de contratación del producto solicitado.

### Licencia

Una licencia es un “permiso para hacer algo. El término también permite nombrar al documento o contrato en que consta la licencia en cuestión”. (8)

Lo habitual es que se pague un monto estipulado para hacer uso de la licencia. En el caso del software, el pago de una licencia autoriza el uso de un programa sin que el usuario sea el dueño del mismo (por eso sólo puede usarlo y no modificarlo o venderlo). (9)

Una licencia es una cadena de caracteres que se genera a partir de otra cadena de caracteres de entrada que se llama semilla. Se envía la cadena generada a la dirección de correo de contacto del cliente cuando ha sido efectuada la venta del producto, la cual se registra en el producto y permite que el mismo se ejecute en la infraestructura física seleccionada por el cliente. (10)

### Reporte

Un reporte es un “informe o una noticia. Este tipo de documento (que puede ser impreso, digital, audiovisual, etc.) pretende transmitir una información, aunque puede tener diversos objetivos”. (11)

En el ámbito de la informática, los reportes son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios. Se entiende que estos documentos sean tan importantes en cualquier empresa pues cuentan con sus propias bases de datos (de trabajadores, de clientes, de proveedores, etc.), realiza recopilaciones de las ventas acometidas en un periodo determinado. (12)

En la presente investigación un reporte es la generación de la información necesaria para el intercambio de datos sobre las ventas entre las divisiones, ayudando en la toma de decisiones.

## 1.2 Soluciones informáticas sobre la gestión de información de las ventas por suscripciones

### 1.2.1 Sistemas CRM (Gestión de las relaciones con el cliente)

El CRM corresponde a las siglas Customer Relationship Management, gestión de las relaciones con el cliente. El máximo objetivo del CRM es de disponer en cualquier momento de toda la información sobre cualquier cliente, tanto para satisfacer las necesidades del cliente, como para obtener estudios de mercado que permitan unas mejores estrategias comerciales. El CRM hace tanto referencia a la estrategia de negocio focalizada hacia el cliente, como a todas las aplicaciones informáticas, tanto software como hardware, necesarias para procesar, analizar y exponer la información resultante para medir y retroalimentar la estrategia de negocio desarrollada. (13)

La función de ventas ha sido siempre una actividad central en todo desarrollo empresarial, ya sea contratando personal clave, formando alianzas, o utilizando herramientas de software como los CRM, lo cual constituye el set de herramientas utilizadas para crear y gerenciar un efectivo y eficiente proceso de ventas.

En un nivel más amplio, el proceso de ventas cubre todas las metodologías, herramientas y prácticas que integra las aplicaciones CRM existentes con los sistemas de distribución. Con apropiadas herramientas y metodologías, la empresa quedaría en posición de identificar las tendencias comerciales, con una evaluación temprana de destrezas en ventas. (14)

Los sistemas CRM tienen distintos módulos y categorías de programación desde JavaScript para programaciones del lado del cliente, Plugins que funcionan de manera sincrónica realizando acciones durante la pre y post creación y actualización de registros que realizan tareas de manera asincrónica. (15). Con un módulo de CRM funcionando en la organización se pueden añadir funcionalidades siempre y cuando sea de código abierto.

### 1.2.2 Sistemas ERP (Planificación de Recursos Empresariales)

Un ERP (Enterprise Resource Planning – Planificación de Recursos Empresariales) es un conjunto de sistemas de información que permite la integración de ciertas operaciones de una empresa, especialmente las que tienen que ver con la producción, la logística, el inventario, los envíos y la contabilidad. Básicamente es un software desarrollado para el manejo eficaz de la información de las empresas, que permite tomar decisiones acertadas en los momentos oportunos, gracias a la veracidad de los datos que se manejan mediante el ERP. Están orientados a facilitar la integración de los sistemas de las empresas, asegurar la comunicación y mejorar su productividad industrial o la eficiencia de la gestión financiera, ventas, reportes y atención al cliente. (16)

Las características que distinguen a un ERP de cualquier otro software empresarial son que deben ser modulares y configurables, poseen una base de datos centralizada a partir de la cual los componentes del ERP interactúan entre sí consolidando las operaciones. Los datos se capturan una sola vez y deben ser consistentes, completos y comunes a todos los procesos. Las empresas que lo implanten suelen tener que modificar alguno de sus procesos para alinearlos con los del sistema ERP, aunque no siempre es necesario. Cabe destacar que entre todas las alternativas que existen en la actualidad en el campo de las herramientas de gestión empresarial, los sistemas ERP son sin lugar a dudas los que ofrecen mayores posibilidades y se adecuan mejor a las distintas necesidades de cada empresa. (16)

### 1.2.3 Odoo

Odoo (conocido anteriormente como OpenERP) es un sistema de ERP integrado de código abierto y multiplataforma, es uno de los principales sistemas de gestión empresarial de software libre, muchas

## Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones

---

empresas utilizan este tipo de software que permiten llevar la contabilidad, la gestión comercial, la gestión de relaciones con clientes y la facturación. (17)

Los módulos de Odoo, en su mayoría, están cubiertos por la licencia AGPL (en inglés, Affero General Public License, licencia derivada de la Licencia Pública General de GNU diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que corra en servidores de red), posee una arquitectura cliente-servidor y utiliza como sistema gestor de bases de datos a PostgreSQL. Odoo es sencillo, ligero y tiene todas sus funcionalidades integradas. (18)

Odoo integra los módulos de CRM, facturación, contabilidad y gestión de proyectos. Siendo de código abierto, tiene posibilidades de ser implantado en una nube empresarial, y con una base de datos centralizada reunir la información de todas las sucursales de una empresa.

### 1.2.4 Comercializador de la OMPI

La OMPI, Organización Mundial de la Propiedad Intelectual, es el foro mundial de servicios, políticas, cooperación e información en materia de propiedad intelectual (P.I.). Es un organismo de las Naciones Unidas, autofinanciado, que cuenta con 188 estados miembros y radica en Italia.

Esta empresa cuenta con un comercializador HSCO, que comercializa productos adaptados a las necesidades específicas de sus clientes. Los productos finales de software se entregan a los clientes para que sean ellos quienes los incorporen a su conveniencia a su sitio web, Intranet o infraestructura de tecnología de la información (T.I.). Además, diseñan productos de software para la venta por suscripción. (19)

### 1.2.5 Comercializador de CSI S.A.

CSI S.A. es una compañía que brinda soluciones informáticas para la gestión comercial de los negocios, en especial aquellos que manejan puntos de venta o facturación en general.

Posee un Sistema POS que le permite manejar los procesos de ventas, inventarios, compras, clientes, créditos y contabilidad, de forma centralizada y en tiempo real logrando así tener el control del negocio de una manera fácil y eficiente. (20)

### 1.2.6 Comercializador de Datazucar

Es una aplicación desktop para viabilizar el proceso de negociación y solicitud de las licencias de uso del Sistema de Gestión Contable VERSAT SARASOLA, con cada uno de los comercializadores y con sus clientes. En el Sistema se realizan los procesos tanto de negociación como de solicitud y entrega de las licencias automáticamente mediante el correo electrónico.

A partir de la definición de los “Codificadores” y de “importar la información necesaria” en el sistema se puede proceder a la captura de la información de negocios para ser enviada a través del correo

## Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones

---

electrónico a Datazucar, quedando en espera de la aceptación del negocio. Siempre que ya un negocio haya sido aceptado, se pueden hacer solicitudes de módulos de VERSAT instalados, con su código de registro y enviarlas, quedando en espera de su licencia de uso siempre y cuando se hayan efectuado los pagos correspondientes.

A continuación, se detallan términos necesarios en la asimilación de este producto:

Convenios: Negocios pactados con los clientes por cada comercializador de software.

Solicitud: Solicitud de generación de licencias de los clientes a su Implantador.

En su configuración se ofrece la posibilidad de entrar la información necesaria para el correcto funcionamiento del sistema y principalmente del envío y recepción de los correos.

Cada vez que se vaya a entrar un nuevo negocio o una solicitud es necesario verificar si el cliente ya existe, de no ser así es necesario incluirlo.

Solo cuando coinciden el importe de los plazos con el del convenio, así como el porcentaje de todos los plazos coincide con el Importe según el convenio, se puede dar por terminado un convenio.

Al enviar solicitudes de licencias, el sistema permite modificar de quién es el correo y el destinatario de la licencia. (21)

### **1.3 La gestión de información sobre las ventas por suscripciones en Desoft**

En Desoft se utiliza actualmente un módulo de CRM correspondiente a Odoo versión 8.0-20150526 (anteriormente conocido como Open-ERP), para manejar iniciativas, oportunidades de negocios y reflejar los datos correspondientes a las entidades con las que la empresa se relaciona, ya sean empresas, unidades presupuestadas, CPA (Cooperativas de Producción Agropecuarias), UBPC (Unidades Básicas de Producción Cooperativa), etc. En este módulo se guardan datos de los contactos, se guardan llamadas y se programan visitas y reuniones.

Los prospectos de venta también se registran en Odoo. Un prospecto no es más que la colección de datos que caracterizan al futuro servicio, en específico el ingreso proyectado y que aún no se ha contratado.

Un posible servicio es la venta de un producto de la cartera por lo que la "Oferta de un Producto" incluye los posibles plazos de pago de la suscripción a la aplicación por parte del cliente y sus montos, donde se pueden aplicar descuentos de acuerdo con la política comercial de la Empresa relacionada con la venta de varios productos diferentes a la vez o del mismo producto en varias unidades del cliente.

La aceptación de una oferta por el cliente precede a la realización del contrato. Cuando ambas partes, cliente y Desoft, firman un contrato (contrato en formato impreso) se formaliza la relación entre el



cliente y el proveedor del producto, donde se fijan los compromisos para los pagos (fechas y montos) de los servicios contratados. En Odoo se guardan los contratos y es sencilla la recuperación de los datos de cualquier contrato.

Sin embargo, en Odoo no es posible controlar la información de ventas de los productos de otras divisiones pues no es una solución en la nube y cada división posee su propia base de datos, sólo tres divisiones utilizan Odoo pues no se encuentra totalmente implantado en la empresa. Por ello las ventas de otras divisiones se conocen a través de los reportes VLO que incluye información sobre las ventas de los productos.

La implementación en las instalaciones físicas del cliente supone el uso de la licencia del producto correspondiente, proceso que comienza por la solicitud del gestor comercial que hizo la venta hacia el gestor comercial de la división dueña del producto, quien entrega la licencia después de comprobar que ha recibido al menos una transferencia de ingresos por medio de un reporte VLO.

Los ingresos se van obteniendo a medida que los clientes cumplen con los pagos de los montos pactados en los plazos que se definen en los contratos, por lo que los reportes VLO deben contener no solo la información de una venta nueva sino la información de cada uno de los ingresos obtenidos mediante los pagos de los montos de los plazos definidos con el cliente en el contrato, pues para cada monto hay que transferir el por ciento que le corresponde a la división dueña. Lo ideal sería contar con la información sobre las ventas centralizada en una nube empresarial, donde las transferencias se produjeran de manera automática a partir de los datos primarios.

Ante la imposibilidad técnica de la nube, una solución parcial es contar con alguna aplicación que complemente los datos de la contratación local con los datos de la contratación externa a la división, así como integrar en esa solución la generación de licencias con comprobación automatizada de las condiciones necesarias para proceder a su generación y envío.

### **1.4 Análisis de las soluciones anteriores sobre la gestión de información de las ventas por suscripciones**

Al analizar todas las soluciones se observa que las mismas no incluyen varias de las funcionalidades que se requieren por la empresa:

- Los CRM se dedican a almacenar la información de los clientes y los contratos de una entidad, no incluyen la posibilidad de controlar información de los contratos de entidades externas ni tampoco están concebidos para controlar entregas de licencias.

- Los ERP son sistemas integrales que sustentan la automatización de los flujos de información de los procesos de la gestión empresarial, incluyen módulos de CRM más gestión financiera contable, de proyectos, etc. Pero no incluyen controlar información sobre entrega de licencias a los clientes.
- Odoó es un ERP específico y por tanto posee la misma limitación anterior.
- Las soluciones de la OMPI, CSI y otras empresas de software internacionales manejan servicios en la nube y por tanto no tienen necesidad de otorgar licencias, como sí lo necesita Desoft en la actualidad.
- La solución de Datazucar maneja información de ventas y licencias de un solo producto (VERSAT) y es un software propietario de Datazucar. (21). Esta aplicación cuenta con funcionalidades importantes que se pudieran emplear, pero Desoft comercializa en el territorio nacional más de un producto y se requieren automatizar en todo lo posible las transferencias de ingresos entre divisiones, cuyos fondos se emplean por cada una de ellas para dar sostenibilidad a la mejora continua de cada uno de los productos que se venden.

Se concluye que se debe desarrollar un sistema de información sobre las ventas de los productos de la cartera de Desoft, que posibilite el manejo de información de las transferencias entre divisiones que se derivan de estas ventas y automatice el proceso de generación y entrega de licencias, solución que necesita ser alojada en un servidor y de manera centralizada poder disponer de la información derivada de los procesos de contratación de suscripciones de productos de todas las divisiones.

### 1.5 Herramientas y tecnologías para el desarrollo del sistema

Para obtener un sistema con calidad es necesario utilizar herramientas y tecnologías que agilicen el proceso de desarrollo de software y que estén enfocadas en la obtención de un producto. En el presente epígrafe se describen las herramientas y tecnologías que se emplean en el desarrollo e implementación del sistema. Se argumenta la utilización de la metodología de desarrollo de software, herramienta CASE (Ingeniería de Software Asistida por Ordenador), lenguaje de modelado, y otros, conformando así la propuesta de solución de esta investigación.

#### 1.5.1 Metodologías de desarrollo de Software

Las metodologías para el desarrollo del software proponen la utilización de procedimientos, guías y técnicas que establezcan el modo de construcción del software (22). Una metodología representa un marco de trabajo que tiene entre sus funciones guiar, planificar, estructurar, controlar, manipular y dirigir el proceso de desarrollo de sistemas de información. El uso de estas metodologías proporciona un proceso más predecible y eficiente en la confección de un producto. Además, es sencillo para aumentar la calidad del software en todas las fases del desarrollo del mismo (23).

Las metodologías ágiles de desarrollo están especialmente indicadas en proyectos con requisitos poco definidos o cambiantes. Estas metodologías se aplican bien en equipos pequeños que resuelven

problemas concretos donde una correcta modularización de los mismos es fundamental para su exitosa implantación. (24)

### **Open UP (Open Unified Process)**

Es un proceso extensible, dirigido a gestión y desarrollo de proyectos de software basados en un desarrollo iterativo, ágil e incremental apropiado para proyectos pequeños y de bajos recursos; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo. (25)

Open UP permite disminuir las probabilidades de fracaso en los proyectos pequeños e incrementar las probabilidades de éxito. Permite detectar errores tempranos a través de un ciclo iterativo, evita la elaboración de documentación, diagramas e iteraciones innecesarias. Por ser una metodología ágil tiene un enfoque centrado al cliente, con iteraciones cortas y es basa en casos de uso, además de ser libre. (25)

Teniendo en cuenta las necesidades para la solución del problema investigativo, la línea de desarrollo de la empresa Desoft y la sugerencia por parte del cliente, se utiliza la metodología de desarrollo ágil Open UP, además funcionan bien para equipos pequeños localizados en un mismo sitio, características que cumple el equipo de desarrollo, siendo de un solo integrante, la aplicación deberá ser evolucionada desde un planteamiento de requisitos preliminares hasta su ajuste a una versión 1.0 trabajando de conjunto con los clientes finales. También por ser una metodología centrada al cliente y con iteraciones cortas.

### **1.5.2 Lenguaje de modelado UML**

Un lenguaje de modelado es un conjunto estándar de símbolos y de formas que facilita la modelación de un diseño de software. UML, son las siglas de Unified Modeling Language (Lenguaje Unificado de Construcción de Modelos), es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML permite una forma de modelar conceptualmente procesos de negocio y funciones de sistema, además de elementos concretos como son esquemas de base de datos, componentes de software reutilizables y escribir clases en un lenguaje determinado. Es interactivo, orientado a objetos, es altamente usado por la comunidad internacional y se ha convertido en un estándar a la hora de desarrollar sistemas informáticos. (26)

El lenguaje UML se elige puesto que puede ser adaptado para satisfacer necesidades de un proyecto específico. UML es gratuito, accesible a todos, y conforma la colección de varias técnicas de ingeniería que han probado ser un éxito en el modelamiento de sistemas grandes y complejos.

En la actualidad existen varias herramientas CASE (por sus siglas en inglés, Computer Aided Software Engineering, en español Ingeniería de Software Asistida por Ordenador) que permiten realizar estos diagramas. A continuación, se define la herramienta CASE utilizada.

### 1.5.3 Herramientas CASE

De acuerdo con (Kendall y Kendall, 1998) (27) las herramientas CASE son diseñadas para automatizar o apoyar una o más fases del ciclo de vida de un software. Mediante el uso de estas herramientas se puede reducir el consumo de tiempo y dinero en la confección de un sistema. Una de sus principales ventajas es la ayuda para aumentar la calidad de un software.

#### Visual Paradigm for UML 8.0

Visual Paradigm es una herramienta CASE multiplataforma y con licencia gratuita que propicia un conjunto de ayudas para el desarrollo de programas informáticos dando soporte al modelado visual con UML. Sustenta todas las fases de desarrollo de un software, desde la planificación, pasando por el análisis y el diseño. Permite la integración con diferentes entornos de desarrollo integrado, entre los que se encuentran NetBeans, JDeveloper y Eclipse. Visual Paradigm brinda la posibilidad de exportar los diagramas realizados en imágenes jpg, png, etc. Esta herramienta fue diseñada para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque orientado a objetos. (28)

Se selecciona Visual Paradigm en su versión 8.0 debido a que es una herramienta que brinda una interfaz de usuario amigable y fácil de usar. Proporciona al desarrollador un ambiente completamente funcional y permite ahorrar tiempo durante el proceso de desarrollo del software, principalmente durante la fase de construcción. De manera general se considera un estándar ampliamente utilizado en el mundo por las empresas, para el modelado en el proceso de desarrollo de software. Suministra facilidades en el trabajo colaborativo y la integración, soportando la confección de todos los artefactos que propone la metodología de software seleccionada.

### 1.5.4 Notación de modelado de Procesos de negocio BPMN

La notación de modelado de procesos de negocio empleada ha sido Business Process Modeling Notation (BPMN, por sus siglas en inglés). Proporciona la comprensión de los procesos de negocio en una notación gráfica, permitiendo la comunicación de estos procedimientos de manera estándar. Esta notación ha sido diseñada específicamente para coordinar la secuencia de procesos y los mensajes que fluyen entre los diferentes procesos participantes. (29)

### 1.5.5 CMS (Sistema de Gestión de Contenidos) Drupal 7.26

Drupal es un sistema de gestión de contenidos por módulos y muy configurable, es de código abierto y está escrito en PHP. Es desarrollado y mantenido por una activa comunidad de usuarios garantizando su evolución. Destaca por la calidad de su código, el respeto de los estándares web, la usabilidad, flexibilidad y adaptabilidad. Admite crear, clasificar y publicar casi cualquier tipo de información en una

página web. Trabaja con una base de datos, de modo que el editor solo actualiza la misma con los cambios realizados, adicionando la nueva información o modificando la existente. (30)

Drupal puede ser instalado con diferentes gestores de bases de datos, como MySQL, SQLite, PostgreSQL, Oracle, etc. La agregación de una gran cantidad de módulos existentes, hace que sea adecuado para la realización de muchos tipos de sitios web. (31)

El CMS Drupal posee además características como:

- Fácilmente extensible a través de módulos que permiten adaptar rápidamente la aplicación a necesidades específicas o cambiantes.
- Administración de usuarios y permisos basados en roles.
- Independencia de la base de datos, multiplataforma y múltiples idiomas.
- Sistema de búsqueda integrada.
- Administración vía web, registro e informes de errores y estadísticas.
- Sistema de Cache que elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor. (31)

Para la instalación de Drupal se selecciona entre los sistemas gestores de base de datos mencionados anteriormente a PostgreSQL.

### 1.5.6 SGBD (Sistema Gestor de Base de Datos) PostgreSQL 9.4

Según (Riquelme y Morales, 2010) (32) un Sistema Gestor de Bases de Datos es un conjunto de programas, procedimientos y lenguajes que proporcionan las herramientas necesarias para trabajar con una base de datos. Incorpora una serie de funciones que permite definir los registros, sus campos, sus relaciones, insertar, modificar, eliminar y consultar los datos.

PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos. Es uno de los sistemas de gestión de base de datos de código abierto más potentes. El uso de multiprocesos en vez de multihilos garantiza la estabilidad del sistema donde un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. (33)

Presenta características como:

- Alta concurrencia: Permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.
- Amplio soporte de datos: Números de precisión arbitraria, texto de largo ilimitado, etc.
- Implementa el uso de claves ajenas también denominadas llaves foráneas.
- Herencia de tablas.

## Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones

---

Se escoge PostgreSQL como SGBD debido a que posee una gran escalabilidad. Es capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta. PostgreSQL implementa el uso de sub-consultas y transacciones, haciendo su funcionamiento mucho más eficaz. De forma general PostgreSQL es un SGBD confiable, robusto y estable, que corre en la mayoría de los sistemas operativos actuales, posee un buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas, así como gran capacidad de almacenamiento.

Para su uso se seleccionan herramientas de administración para PostgreSQL entre las que se encuentran PgAdmin3, PgAccess, PhpPgAdmin y Database Master. (34)

### 1.5.7 PgAdmin 3

PgAdmin 3 es una herramienta de administración para PostgreSQL de código abierto y multiplataforma. (35). Tiene una interfaz administrativa gráfica que soporta todas las características del SGBD utilizado en el sistema a desarrollar y hace simple la administración de dicho gestor.

PgAdmin 3 se selecciona por las ventajas que brinda en la administración del SGBD. Además, está diseñado para responder a varias necesidades que van desde escribir simples consultas hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y la aplicación también incluye un resaltado de sintaxis SQL, un editor de código del lado del servidor, y mucho más. Otra característica a tener en cuenta en su selección es la licencia bajo la que se distribuye el producto la cual permite la fácil adquisición del mismo. (36)

### 1.5.8 Desarrollo web

Los lenguajes de programación a utilizar son otros de los aspectos fundamentales a tener en cuenta para la confección del sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft. Alguno de los lenguajes de programación utilizados para el desarrollo web son ASP.NET, PHP y JavaScript. El desarrollo web es un conjunto de tecnologías de software del lado del servidor y del cliente con el objetivo de mostrar información en un navegador web o realizar un proceso determinado. (37)

### 1.5.9 Lenguaje de programación al lado del cliente

#### JavaScript

Es un lenguaje de programación interpretado utilizado para construir sitios web más interactivos. Aunque comparte muchas de las características y de las estructuras del lenguaje Java, fue desarrollado independientemente. El lenguaje JavaScript puede interactuar con el código HTML (Hypertext Markup Language, en español Hipertexto Lenguaje de marcado), permitiendo a los programadores web utilizar contenido dinámico. Su utilización alivia la sobrecarga del servidor siendo un lenguaje del lado del cliente. (38)

JavaScript es de gran utilidad en el manejo de eventos sin usar recursos del servidor y posee alta compatibilidad con la mayoría de los navegadores web modernos. Su empleo es recomendable aún para personas con poca experiencia en programación, pues su sencillez permite asimilarlo y practicarlo con facilidad. Este lenguaje se encuentra vinculado a las tecnologías AJAX (Asynchronous JavaScript + XML) para enviar y recibir peticiones al servidor y aligerar este intercambio de información.

### **AJAX**

AJAX es un lenguaje que está fundamentado en una técnica de programación basada en estándares, diseñada para hacer aplicaciones web más sencillas, interactivas y personalizables. Entre las tecnologías que la constituye se encuentran XHTML, CSS, JSON, DOM y JavaScript. (39). Es una tecnología multiplataforma basada en estándares abiertos como JavaScript por lo que evita recargas en el servidor. Mediante el uso de AJAX en el sistema se aumenta la interacción del usuario con la aplicación, evitando las recargas constantes de páginas, lo cual propicia que la comunicación con el usuario no sea interrumpida.

#### **1.5.10 Lenguaje de programación del lado del servidor**

##### **PHP 5**

PHP (Hypertext Preprocessor, en español Procesador de hipertexto) es un lenguaje interpretado y multiplataforma de alto nivel. PHP presenta interfaces para una gran cantidad de sistemas de base de datos diferentes y dispone de conexión propia para estos sistemas. (40)

Por otra parte, el uso de PHP garantiza el uso de técnicas de programación orientada a objetos. Entre sus ventajas se encuentra la facilidad de aprendizaje, la incorporación de bibliotecas para facilitar el trabajo, portabilidad y licencia libre. (40)

#### **1.5.11 IDE (Entorno de Desarrollo Integrado)**

##### **NetBeans IDE 7.4**

NetBeans IDE (Entorno de Desarrollo Integrado por sus siglas en inglés) es una herramienta de código abierto distribuida bajo la licencia pública GPL (General Public License, en español Licencia Pública General). Proporciona la creación de aplicaciones de escritorio, móviles y web, así como aplicaciones HTML5 con HTML, JavaScript y CSS. Entre sus funcionalidades permite escribir, depurar, compilar y ejecutar programas. Además, se destacan sus ventajas de generación automática de códigos para los lenguajes que soporta. Es gratuito y tiene una gran comunidad de usuario y desarrolladores a nivel mundial. (41)

NetBeans por sus principales características es el IDE de desarrollo que se emplea para la programación web durante el proceso de desarrollo del sistema. Además, su selección también está



dada porque soporta el lenguaje de programación PHP, presenta una potente integración con el CMS Drupal y con el sistema de control de versiones Subversion.

### 1.5.12 SVN (Subversion)

SVN es un sistema de control de versiones desarrollado bajo la licencia de código abierto Apache/BSD. Estos sistemas permiten el regreso a un estado anterior del código fuente que almacena en caso de la ocurrencia de un estado no esperado. El mismo permite el uso del repositorio a través de redes por lo que puede ser usado por personas ubicadas en distintas computadoras. El uso de Subversion fomenta la colaboración y agiliza el trabajo pues no se precisa de un único flujo de modificaciones. (42)

La utilización de Subversion está dada por su condición de software libre y la alta disponibilidad de clientes, tanto para distintos sistemas operativos como plataformas.

## 1.6 Conclusiones del capítulo

- El cambio del modelo de negocios en Desoft, desde la venta de las licencias de las aplicaciones hacia la venta de suscripciones aumenta la cantidad de información a manejar sobre los clientes, al dar la posibilidad de que el cliente se suscriba por el tiempo que desee, en comparación con la venta de licencias, donde el cliente paga la licencia de una vez y le sirve de por vida.
- El mecanismo de entrega de licencias tiene que seguir funcionando pues los productos de la cartera de Desoft no son libres de precio, y hay que seguirlos instalando en las infraestructuras de los clientes mientras no sea posible utilizar una nube pública.
- De las soluciones para comercializar productos existentes, las mismas no incluyen varias de las funcionalidades que se requieren por la empresa.
- La solución que demanda Desoft para comercializar sus productos de la cartera en el territorio nacional necesita ser alojada en un servidor y de manera centralizada poder disponer de la información derivada de los procesos de contratación de todas las divisiones, puesto que los clientes tienen que ser atendidos en primer lugar de manera rápida al respecto de la entrega de las licencias de los productos. Por ello, un sistema de manejo de contenidos (CMS) resulta una herramienta útil donde los CMS permiten desarrollar aplicaciones web, con potentes sistemas de gestión de bases de datos integrados.
- En particular, Drupal es un CMS fácilmente extensible a través de módulos que permiten adaptar rápidamente la aplicación a necesidades específicas o cambiantes. Por otra parte, gestiona la administración de usuarios y permisos basados en roles permitiéndole adaptabilidad a las diferentes divisiones existentes en Desoft. Funciona con varios sistemas de gestión de bases de datos y es multiplataforma, aumentando la flexibilidad de la aplicación.



## Capítulo 1: Soluciones y herramientas informáticas para la gestión de ventas por suscripciones

---

- OdoO es un ERP que incluye, de manera integrada, módulos para la gestión comercial, la gestión de relaciones con clientes, la facturación, la contabilidad, el manejo de los recursos humanos, las compras, las ventas y la gestión de proyectos.
- Desoft hoy está en proceso de implantación de OdoO (la versión 8 de Open ERP), que es un ERP libre, y para el cual habría que programar, sobre su módulo de CRM, las funcionalidades requeridas, pero los flujos de eventos de los objetos de OdoO tendrían que ser modificados para resolver esta necesidad, y ante la ausencia de una implantación completa del ERP en Desoft (lo cual demanda bastante tiempo) se decidió implementar la solución que se va a describir en el Capítulo 2, que está preparada para su integración con el VERSAT que es hoy el sistema de gestión contable financiera que se está usando, y se usará aproximadamente hasta el 2018.
- La metodología de desarrollo escogida fue Open Unified Process (Open UP) por la necesidad de intercambiar sistemáticamente con los usuarios finales y adaptarse a requisitos cambiantes.

## **Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft**

En el presente capítulo se define la propuesta del sistema a desarrollar, abordando los aspectos relacionados con el proceso del negocio y las características del sistema. Para optimizar la comprensión de la solución propuesta se realiza el análisis y diseño utilizando la metodología ágil Open Up. Se enuncian los requisitos funcionales y no funcionales de la aplicación; se describe la propuesta de solución, mostrando el modelo del negocio, el modelo conceptual y los diagramas de casos de uso del sistema, su priorización y la descripción de los mismos. Como parte del diseño de la solución se desarrollan los diagramas de clases y de secuencia, que constituyen los artefactos principales generados para esta etapa.

### **2.1 Propuesta del sistema**

Con la implementación de la solución planteada se pretende:

1. Reunir información sobre las ventas de los productos de la cartera de Desoft, que permita producir con rapidez y calidad los datos sobre las transferencias que se deben efectuar desde las divisiones que venden hacia las divisiones dueñas de productos.
2. Advertir las deudas entre divisiones, a fin de que se hagan las gestiones de facturación y cobro a los clientes cuando corresponde según lo contratado.
3. Generar con mayor rapidez las licencias de los productos haciendo de manera automatizada las verificaciones sobre los ingresos recibidos por el producto y enviándolas por correo electrónico a los gestores comerciales.

Para representar estos objetivos la aplicación debe contar con la posibilidad de gestionar productos, precios, plazos de pago, clientes, divisiones, provincias y municipios.

Debe permitir definir roles para usuarios, a fin de restringir el uso de los datos de la aplicación que sean previstos para el personal autorizado.

Las licencias se controlarán por la aplicación eliminando la posibilidad de que un gestor comercial de una división genere una licencia de un producto que no es mantenido por su división. Se podrán copiar en ficheros de texto para trasmitirlas vía telefónica en los casos donde no se pueda establecer comunicación por correo electrónico con el grupo de despliegue del producto. Así mismo las licencias se conservarán en la base de datos y se deben poder localizar a través del sistema haciendo búsquedas, para permitir que se vuelvan a enviar cuando sea necesario recuperarlas.

### 2.2 Modelo de negocio

Con el modelado de negocio se identifican los procesos, la información con la que se va a trabajar, los actores que participan en dichos procesos y los papeles que representan cada uno de ellos con respecto a la información para lograr comprender el entorno en que va a funcionar el sistema. Como resultado de la identificación, captura y documentación de la información, se lleva a cabo la base para especificar los requisitos de un sistema (43). Para modelar los procesos de negocio identificados se utilizará la Notación para el Modelado de Procesos de Negocios (BPMN, por sus siglas en inglés).

#### 2.2.1 Descripción del negocio

En la figura 1 se aprecian cuatro actores del negocio: el gestor comercial de la división que vende el producto, el gestor comercial de la división dueña del producto, el contador y el grupo de despliegue.

El proceso comienza por la confección del contrato con el cliente por parte del gestor comercial de la división que vende el producto. Este mismo gestor factura cada plazo acordado en el contrato de manera que la división obtiene el ingreso planificado para ese plazo. A fin de mes elabora el reporte VLO y lo envía hacia la división dueña del producto, con una o varias ventas de los productos de los que es dueña esa división.

Al recibir el VLO, el mismo es guardado por el gestor comercial, y el contador de la división dueña contabiliza el por ciento de los ingresos que le corresponde.

El grupo de despliegue al instalar el producto en las instalaciones del cliente, obtienen una “semilla” que es enviada hacia el gestor comercial de la división dueña del producto, quien comprueba que al menos ha recibido la primera transferencia de ingresos consultando los reportes VLO que tiene en su poder (estos reportes se almacenan hoy en carpetas compartidas de un servidor).

Si en efecto, existe al menos una transferencia efectuada hacia la división, el gestor comercial de la división dueña del producto procede a generar la licencia del producto y la envía por correo a la dirección de contacto de la división que solicita la licencia, para finalmente ser concedida por éste al grupo de despliegue para que el producto funcione correctamente durante todo el periodo de suscripción contratado.

Si no se encuentra al menos una transferencia efectuada hacia la división, el gestor comercial de la división dueña del producto no debe generar la licencia del producto.

La situación actual de grandes volúmenes de venta del mismo producto por muchas divisiones genera una cantidad de información muy difícil de controlar de manera manual y por ello las reclamaciones de transferencias dejadas de efectuar se vuelven una actividad cuello de botella en la empresa.

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

En el siguiente diagrama se muestra cómo se lleva a cabo el proceso de venta por suscripción de un producto de la cartera de Desoft. Se muestra el flujo de actividades que realizan los actores del negocio para lograr la venta de un producto mediante suscripciones y la generación de la licencia del producto para su uso por el cliente, exponiendo además la manera en que se realizan las transferencias acordadas entre divisiones.

### Diagrama del proceso de negocio: venta por suscripción de un producto de la cartera de Desoft

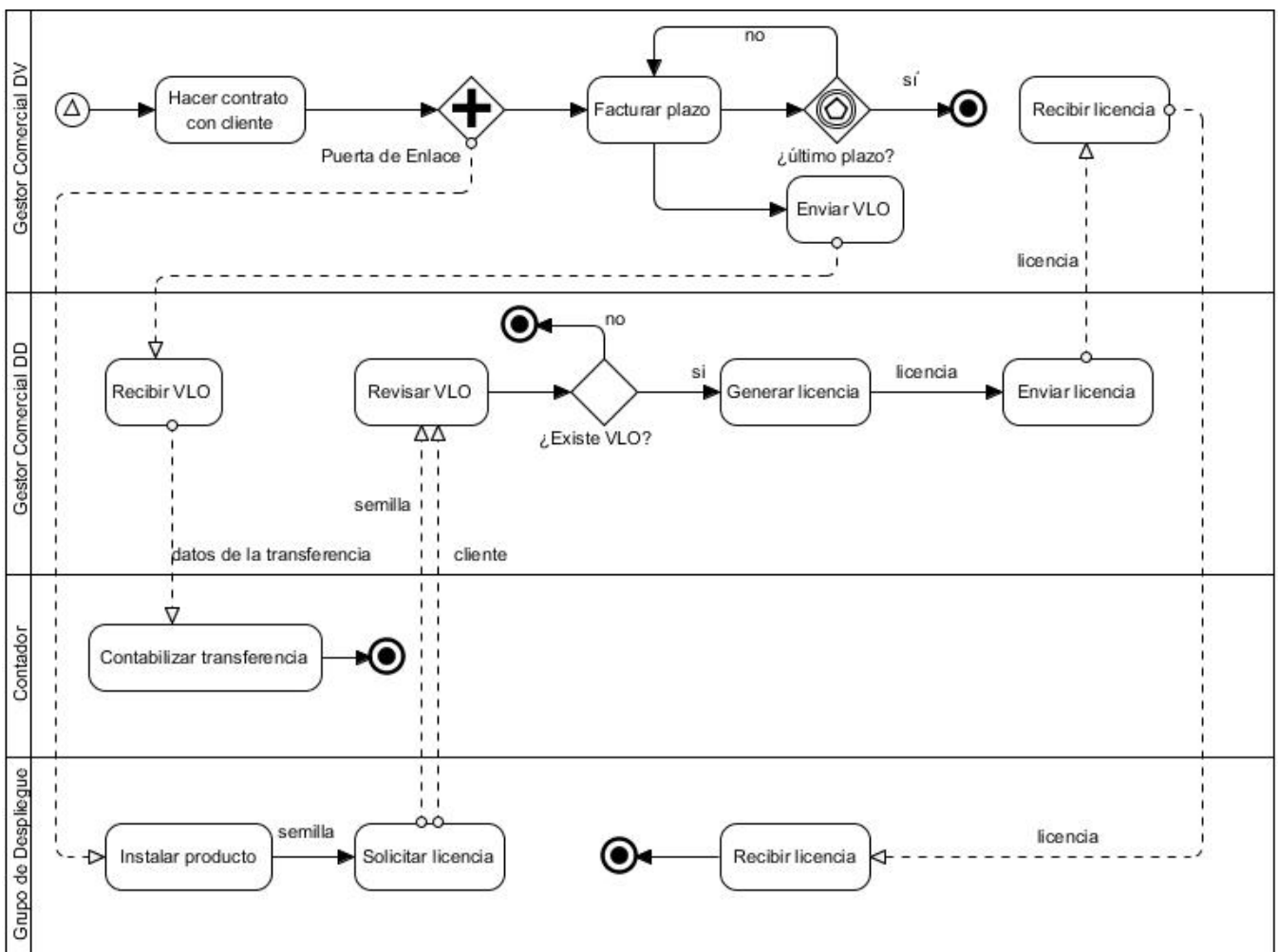


Figura 1. Diagrama de venta por suscripción de un producto de la cartera de Desoft

## 2.3 Modelo conceptual

El modelo conceptual es una representación de conceptos en un dominio del problema. Este modelo muestra asociaciones entre conceptos y atributos de conceptos. Se puede ver como un modelo que comunica los términos importantes y cómo se relacionan entre sí (44). En el siguiente diagrama se representan los conceptos que intervienen en el negocio actual de la presente investigación, así como las relaciones entre ellos.

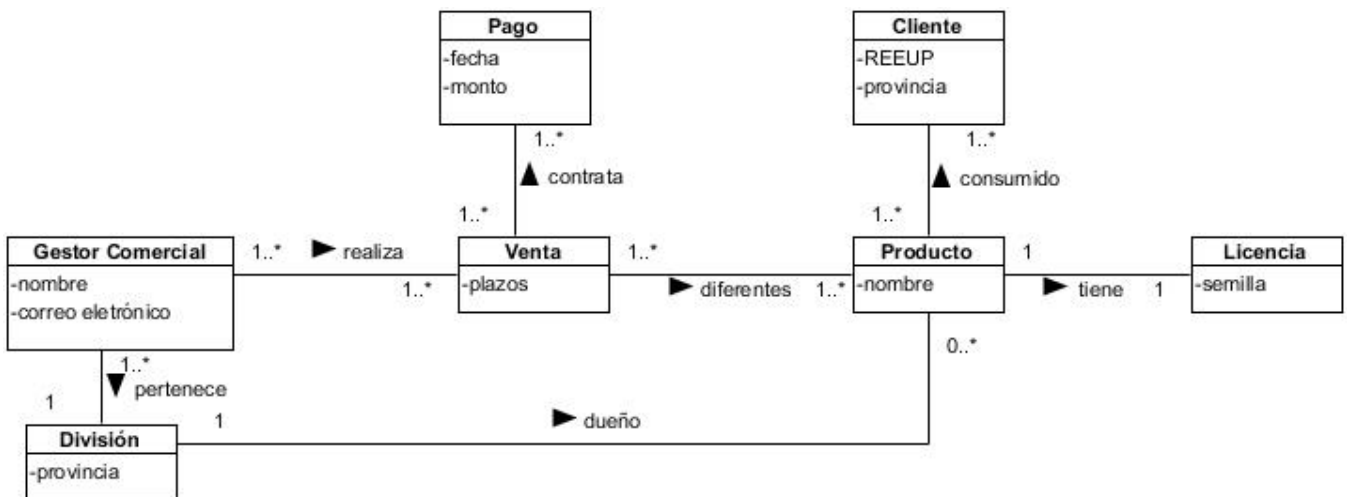


Figura 2: Modelo Conceptual

### 2.3.1 Descripción de los conceptos que intervienen en el dominio del problema

**Cliente:** es la empresa o entidad presupuestada que contrata un servicio, interesándose por uno o varios productos de la cartera de la empresa.

**División:** unidad organizativa básica en la estructura de la empresa Desoft.

**Producto:** aplicación que se le brinda al cliente para la satisfacción de sus necesidades.

**Gestor Comercial:** es la persona de la división encargada de hacer la venta del servicio o producto solicitado por el cliente.

**Licencia:** cadena de caracteres que se genera a partir de otra cadena de caracteres de entrada que se llama semilla, la cual se registra en el producto y permite que el mismo se ejecute en la infraestructura física seleccionada por el cliente.

**Pago:** es la operación que se realiza cuando se efectúa la venta de un producto contratado. Pueden existir uno o varios pagos en dependencia de los plazos que se fijen en el contrato con el cliente.

**Venta:** es la transferencia de un producto al cliente mediante el pago de un precio convenido.

### 2.4 Modelo del sistema

La especificación de requisitos en el proceso de desarrollo del software es de vital importancia. Tener los requisitos bien claros y definidos permite comprender desde un inicio la línea a seguir en el desarrollo y así garantizar la eficiencia y calidad del software. (45)

### 2.4.1 Requisitos funcionales

Para poder identificar que debe hacer el sistema y entender su funcionamiento, es fundamental conocer los requisitos funcionales que el sistema debe cumplir. A continuación, se muestran los requisitos funcionales (RF) identificados con una breve descripción:

#### Listado de requisitos funcionales

**RF 1:** Autenticar usuario.

**Descripción:** El sistema debe permitir que los usuarios previamente registrados se autenticuen en el mismo.

**RF 2:** Salir del sistema.

**Descripción:** El sistema debe permitir a los usuarios previamente autenticados salir del mismo.

**RF 3:** Gestionar usuario.

**RF 3.1:** Insertar usuario.

**Descripción:** El sistema debe permitir que los administradores inserten usuarios.

**RF 3.2:** Editar usuario.

**Descripción:** El sistema debe permitir que los administradores editen los datos asociados al usuario, luego de ser seleccionado.

**RF 3.3:** Eliminar usuario.

**Descripción:** El sistema debe permitir que los administradores eliminen el usuario seleccionado.

**RF 4:** Gestionar rol.

**RF4.1:** Insertar rol.

**Descripción:** El sistema debe permitir al administrador insertar un nuevo rol.

**RF4.2:** Editar rol.

**Descripción:** El sistema debe ser capaz de permitir al administrador editar el nombre del rol una vez creado.

**RF 4.3:** Eliminar rol.

**Descripción:** El sistema debe brindar la opción al administrador de eliminar un rol ya creado.

**RF 5:** Gestionar cliente.

**RF 5.1:** Adicionar cliente.

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

**Descripción:** El sistema debe permitir al administrador adicionar un cliente, con los siguientes atributos:

- Nombre
- REEUP
- Dirección
- Provincia
- Municipio
- Teléfono
- Correo electrónico

**RF 5.2:** Editar datos del cliente.

**Descripción:** El sistema debe permitir al administrador editar los datos del cliente seleccionado (Remitirse a RF 5.1).

**RF 5.3:** Eliminar cliente.

**Descripción:** El sistema debe permitir al administrador eliminar el cliente seleccionado.

**RF 5.4:** Mostrar información del cliente.

**Descripción:** El sistema debe permitir visualizar los datos de los clientes registrados (Remitirse a RF 5.1).

**RF 6:** Gestionar provincia.

**RF 6.1:** Adicionar provincia.

**Descripción:** El sistema debe permitir al administrador crear provincias con los siguientes atributos:

- Nombre de provincia
- Municipios

**RF 6.2:** Editar provincia.

**Descripción:** El sistema debe permitir al administrador editar la provincia seleccionada (Remitirse a RF 6.1).

**RF 6.3:** Eliminar provincia.

**Descripción:** El sistema debe permitir al administrador eliminar la provincia seleccionada.

**RF 6.4:** Mostrar información de la provincia.

**Descripción:** El sistema debe permitir visualizar las provincias creadas (Remitirse a RF 6.1).

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

### RF 7: Gestionar división.

#### RF 7.1: Adicionar división.

**Descripción:** El sistema debe permitir al administrador adicionar una división, con los siguientes atributos:

- Nombre
- Provincia
- Comercial Nombre
- Comercial Correo

#### RF 7.2: Editar división.

**Descripción:** El sistema debe permitir al administrador editar los datos de la división seleccionada (Remitirse a RF 7.1).

#### RF 7.3: Eliminar división.

**Descripción:** El sistema debe permitir al administrador eliminar la división seleccionada.

#### RF 7.4: Mostrar información de la división.

**Descripción:** El sistema debe permitir visualizar los datos de las divisiones creadas (Remitirse a RF 7.1).

### RF 8: Gestionar producto.

#### RF 8.1: Adicionar producto.

**Descripción:** El sistema debe permitir al administrador adicionar un producto, con los siguientes atributos:

- Nombre
- Versión
- Descripción
- Precio Suscripción
- Precio Total
- Por Ciento Dueño
- Por Ciento Vendedor
- División Id
- Imagen



## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

**RF 8.2:** Editar producto.

**Descripción:** El sistema debe permitir al administrador editar los datos del producto seleccionado (Remitirse a RF 8.1).

**RF 8.3:** Eliminar producto.

**Descripción:** El sistema debe permitir al administrador eliminar el producto seleccionado.

**RF 8.4:** Mostrar información de los productos.

**Descripción:** El sistema debe permitir visualizar los datos de los productos creados (Remitirse a RF 8.1).

**RF 9:** Gestionar venta.

**RF 9.1:** Adicionar venta.

**Descripción:** El sistema debe permitir al gestor comercial adicionar una venta.

- Producto Id
- División
- Cliente Id
- Fecha de venta
- Periodo suscripción: fecha de inicio y fecha de fin.
- Plazos de pago: una serie de fechas y montos.

**RF 9.2:** Editar venta.

**Descripción:** El sistema debe permitir al gestor comercial editar los datos de la venta seleccionada (Remitirse a RF 9.1).

**RF 9.3:** Eliminar venta.

**Descripción:** El sistema debe permitir al gestor comercial eliminar la venta seleccionada.

**RF 9.4:** Mostrar información de las ventas.

**Descripción:** El sistema debe permitir visualizar los datos de las ventas creadas (Remitirse a RF 9.1).

**RF 10:** Administrar licencia.

**RF 10.1:** Mostrar información de las licencias.

**Descripción:** El sistema debe permitir visualizar los datos de las licencias con los siguientes atributos:

- División vendedora

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

- Cliente
- Producto
- Licencia
- Fin de suscripción

**RF 10.2:** Generar licencia del producto.

**Descripción:** El sistema debe permitir al gestor comercial de la división dueña del producto generar la licencia para el producto seleccionado si antes se efectuó al menos una transferencia hacia la división dueña del producto.

**RF 11:** Enviar licencia por correo electrónico.

**Descripción:** El sistema debe permitir al gestor comercial de la división dueña del producto enviar la licencia del producto seleccionado por correo electrónico hacia la dirección de contacto del comercial de la división que solicita la licencia.

**RF 12:** Gestionar pago

**RF 12.1:** Adicionar pago del producto.

**Descripción:** El sistema debe permitir al gestor comercial de la división dueña del producto adicionar el pago del producto.

- Tipo de Pago
- Fecha de pago
- Monto pagado

**RF 12.2:** Editar pago del producto.

**Descripción:** El sistema debe permitir al gestor comercial de la división dueña del producto editar los datos del pago seleccionado (Remitirse a RF 12.1).

**RF 12.3:** Eliminar pago del producto.

**Descripción:** El sistema debe permitir al gestor comercial de la división dueña del producto eliminar el pago seleccionado.

**RF 12.4:** Mostrar información de los pagos realizados.

**Descripción:** El sistema debe permitir visualizar los datos de los pagos realizados (Remitirse a RF 12.1).

**RF 13:** Visualizar reporte

**RF 13.1:** Generar reporte de deudas.

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

**Descripción:** El sistema debe permitir al gestor comercial generar reporte de deudas que tienen las otras divisiones, con los atributos:

- División
- Producto
- Monto de la deuda
- Fecha del reporte

**RF 13.2:** Generar reporte "VLO".

**Descripción:** El sistema debe permitir al gestor comercial generar reporte de venta de productos de los que son dueñas otras divisiones.

- División
- Producto
- Cliente
- Fecha de venta
- Ingreso
- Por ciento a enviar

**RF 14:** Enviar reporte VLO por correo.

**Descripción:** El sistema debe permitir al gestor comercial enviar el reporte VLO por correo electrónico hacia las direcciones de contacto de los comerciales de las divisiones dueñas de los productos que se vendieron.

### 2.4.2 Requisitos no funcionales

Los requisitos no funcionales detallan las propiedades o cualidades que el producto debe tener, aumentándole funcionalidad al sistema, pues hacen al producto atractivo, fácil de usar, rápido y confiable, los cuales se encuentran separados por categorías (45) que ahora se mencionarán.

#### **Usabilidad:**

**RNF 1:** En el sistema podrán trabajar, desde usuarios con conocimientos avanzados de informática hasta usuarios inexpertos; esto se logrará a partir de una correcta estructura de la información, con el empleo de menús, que proporcionan una navegación sencilla, la cual no sobrepasa dos niveles de profundidad.

**RNF 2:** En el sistema se debe mostrar información al usuario de las acciones a realizar, esto se logra a partir de mensajes en los iconos que representa dichas acciones.

#### **Fiabilidad:**

**RNF 3:** El sistema debe ser robusto, porque se puede necesitar hacer un cambio en la programación ante nuevas orientaciones de la empresa.

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

---

**RNF 4:** Debe estar disponible en horario laborable.

**RNF 5:** En caso de que el sistema presente alguna falla, los errores se deben mostrar sin detalles de información que puedan comprometer la seguridad e integridad del mismo.

### **Eficiencia:**

**RNF 6:** El sistema deberá tener un nivel de respuesta aceptable, tanto para los accesos a la base de datos, como para el proceso de administración de tareas.

**RNF 7:** El sistema debe permitir la navegación de varios usuarios simultáneamente sin que el rendimiento del sitio se vea afectado drásticamente.

### **Software:**

**RNF 8:** Se debe emplear un sistema gestor de bases de datos para el almacenamiento de los datos que manejará el sistema.

### **Seguridad**

**RNF 9: Confidencialidad:** en el sistema solo podrá gestionar información el personal autorizado, con los permisos correspondientes, por lo que será necesaria una autenticación previa.

**RNF 10: Integridad:** el software deberá estar protegido contra acciones no autorizadas o que puedan afectar la integridad de los datos.

**RNF 11: Disponibilidad:** el sistema deberá estar disponible en horario laborable para todos los usuarios de la empresa Desoft, exceptuándose los días que se encuentre en actualización, corrección de errores, o soporte.

### **Hardware mínimo para el servidor web:**

**RNF 12:** El servidor debe tener al menos 1 GB de memoria RAM siendo el mínimo recomendado para sistemas desarrollados con el CMS Drupal.

**RNF 13:** Debe contar con 30 GB de espacio en el disco duro.

**RNF 14:** Como procesador debe tener un Intel Pentium IV o mayor.

### **Requisitos de soporte:**

**RNF 15:** Para que los usuarios finales sean productivos en el manejo del sistema, se les dará una capacitación donde adquieran las habilidades necesarias para que trabajen eficientemente con el sistema.

### **2.4.3 Definición de los actores del sistema**

Los actores del sistema intercambian información con él, aunque no forman parte de éste. Pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado (46).

A continuación, se muestran los actores y la justificación que tienen en el sistema:

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

Tabla 1: Definición de los actores del sistema

Actor	Descripción
Usuario	Usuario no registrado con permisos para autenticarse en el sistema.
Administrador del sistema	Es un usuario del sistema con todos los permisos para trabajar sobre él. Es el encargado de gestionar los usuarios, los roles y las divisiones, así como modificar la información del sistema.
Gestor comercial	Usuario registrado con permisos para gestionar la información referente a las ventas de un producto, los clientes, los pagos, las provincias, los reportes del sistema y genera las licencias de los productos propios.

### 2.4.4 Diagrama de casos de uso del sistema

El diagrama de casos de uso del sistema ayuda a comprender gráficamente los procesos del sistema y su interacción con los actores.

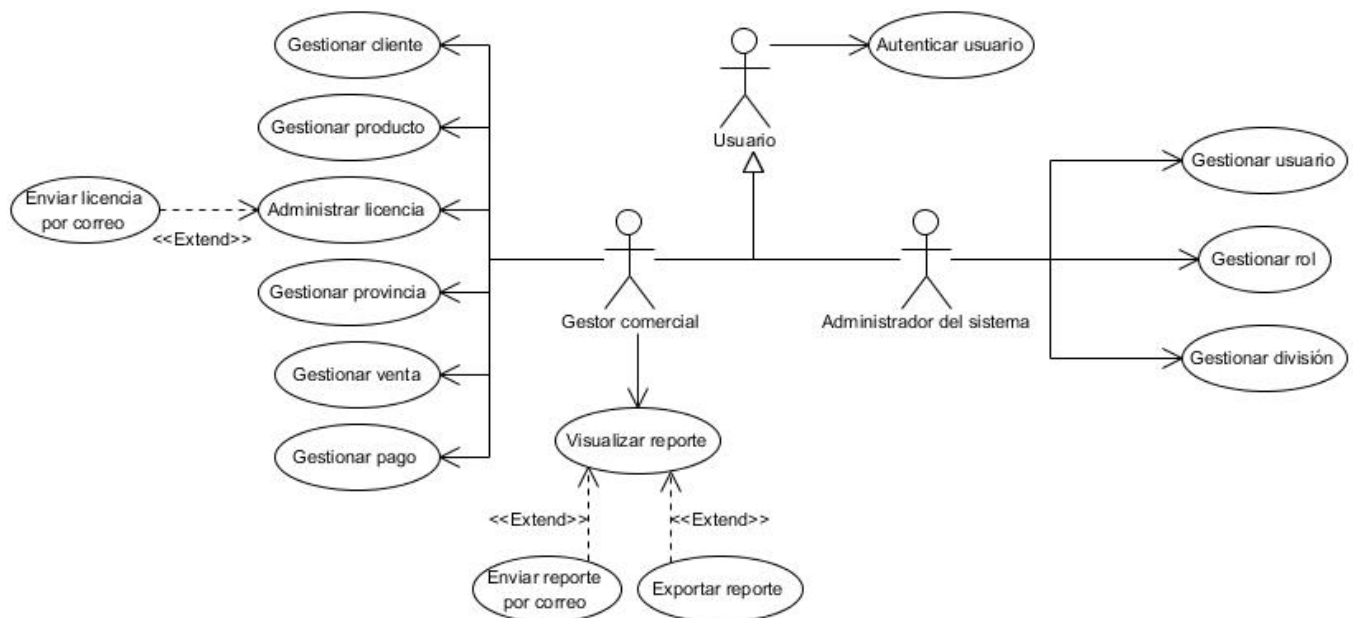


Figura 3: Diagrama de Casos de Uso del Sistema

### 2.4.5 Descripción de casos de uso del sistema

La descripción amplia de los casos de uso del sistema permite realizar el diseño, implementación y las pruebas de la interfaz. A continuación, se realiza la descripción expandida de uno de los casos de uso del sistema:

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

Tabla 2: Descripción del caso de uso "Gestionar cliente"

<b>Caso de uso</b>	Gestionar cliente	
<b>Objetivo</b>	Adiciona un nuevo cliente en el sistema, edita, muestra o elimina los que ya estén previamente almacenados.	
<b>Actores</b>	Gestor comercial (Inicia): Adiciona, edita, muestra y elimina los datos de los clientes en el sistema.	
<b>Resumen</b>	El caso de uso se inicia cuando el actor selecciona la Gestionar Cliente del menú "GESTIÓN". La aplicación muestra la lista de clientes almacenados y se le brinda al actor la posibilidad de adicionar un nuevo cliente, editar o eliminar los ya existentes.	
<b>Complejidad</b>	Alta	
<b>Prioridad</b>	Crítico	
<b>Precondiciones</b>	Gestor comercial autenticado.	
<b>Postcondiciones</b>	Se adiciona, edita, muestra o elimina el cliente.	
<b>Referencias</b>	RF 9, RF 10, RF 11, RF 12	
<b>Flujo de eventos</b>		
<b>Flujo básico: Gestionar cliente</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	Selecciona la opción "GESTION" donde selecciona el submenú Gestionar cliente. (A)	
2.		<p>Muestra un listado con los clientes almacenados y posibilita al actor ejecutar las siguientes acciones:</p> <ol style="list-style-type: none"> <li>1. Adicionar cliente (B). Ver Sección 1: "Adicionar cliente".</li> <li>2. Muestra clientes adicionados (C). Ver Sección 2: "Mostrar clientes".</li> <li>3. Editar cliente (D). Ver Sección 3: "Editar cliente"</li> <li>4. Eliminar cliente (E). Ver Sección 4: "Eliminar cliente"</li> </ol> <p>(Ver prototipo de interfaz 1)</p>
3.		Termina el CU

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

Prototipo de interfaz 1		
Sección 1: “Adicionar cliente”		
1.	Selecciona la opción “Adicionar cliente”.	
2.		Muestra una interfaz con los campos necesarios para adicionar los datos del cliente.
3.	<p>Inserta los datos teniendo en cuenta los siguientes atributos:</p> <ul style="list-style-type: none"> <li>- Nombre (obligatorio). (A)</li> <li>- REEUP (obligatorio). (B)</li> <li>- Dirección. (C)</li> <li>- Teléfono. (D)</li> <li>- Correo electrónico (obligatorio). (E)</li> <li>- Provincia. (F)</li> <li>- Municipio. (G)</li> </ul> <p>Luego de llenar los datos selecciona la opción “Guardar”. (H)</p> <p>(Ver prototipo de interfaz 2)</p>	
4.		Valida que no existan campos obligatorios vacíos.
5.		Valida que no se introduzcan letras en el campo del teléfono.

**Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.**

6.		Inserta los datos del cliente en la base de datos.
7.		Muestra un mensaje de notificación indicando que los datos se guardaron satisfactoriamente.  (Ver prototipo de interfaz 3)
8.		Termina el CU.

**Prototipo de interfaz 2**

Diagrama de un formulario de registro de cliente con los siguientes elementos etiquetados:

- (A) Nombre
- (B) REEUP
- (C) Dirección
- (D) Teléfono
- (E) Correo electrónico
- (F) Provincia
- (G) Municipio
- (H) Guardar
- (I) Cancelar

**Prototipo de interfaz 3**

Información: La acción fue realizada correctamente

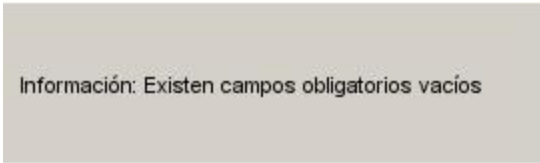
**Flujos alternos**

**3a. Evento: Selecciona la opción "Cancelar".**

	Actor	Sistema
3a	Selecciona la opción "Cancelar". (I)  (Ver prototipo de interfaz 2)	
3b		Regresa al paso 2 del Flujo Básico: Gestionar cliente.  Termina el CU.



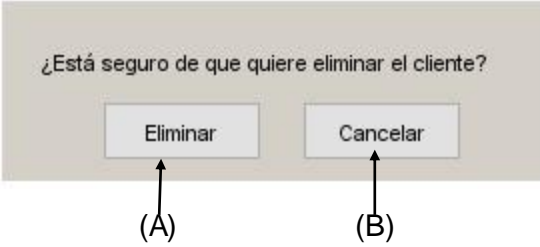
## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

Flujos alternos		
<b>4a. Evento: Campo obligatorio vacío.</b>		
	<b>Actor</b>	<b>Sistema</b>
4a		<p>Muestra un mensaje de error indicando que hay campos obligatorios vacíos.</p> <p>(Ver prototipo de interfaz 4)</p> <p>Regresa al paso 3 del Flujo Básico: Adicionar cliente.</p>
Prototipo de interfaz 4		
 <p>Información: Existen campos obligatorios vacíos</p>		
Flujos alternos		
<b>5a. Evento: No se permiten letras en el campo del Teléfono</b>		
	<b>Actor</b>	<b>Sistema</b>
5a		<p>Muestra un mensaje de error indicando que no se permiten letras en el campo del Teléfono.</p> <p>Regresa al paso 2 del Flujo Básico: Gestionar cliente.</p> <p>Termina el CU.</p>
Sección 2: "Editar datos del cliente"		
Flujo básico: Editar datos del cliente		
	<b>Actor</b>	<b>Sistema</b>
1.	Selecciona la opción editar datos del cliente que desee.	
2.		Carga los datos del cliente seleccionado y los muestra en una interfaz permitiendo su edición.
3.	<p>Cambia los atributos:</p> <ul style="list-style-type: none"> <li>- Nombre (obligatorio). (A)</li> <li>- REEUP (obligatorio). (B)</li> </ul>	

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

	<ul style="list-style-type: none"> <li>- Dirección. (C)</li> <li>- Teléfono. (D)</li> <li>- Correo electrónico (obligatorio). (E)</li> <li>- Provincia. (F)</li> <li>- Municipio. (G)</li> </ul> <p>Luego presiona la opción "Guardar". (H)</p> <p>(Ver prototipo de interfaz 2)</p>	
4.		Valida que no existan campos obligatorios vacíos.
5.		Busca en la base de datos los datos del cliente de acuerdo a su identificador y modifica los datos.
6.		Muestra un mensaje de notificación indicando que los datos se actualizaron satisfactoriamente.
		(Ver prototipo de interfaz 3)
		Termina el CU.
<b>Flujos alternos</b>		
<b>3a. Evento: Selecciona la opción "Cancelar".</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	<p>Selecciona la opción "Cancelar". (I)</p> <p>(Ver prototipo de interfaz 2)</p>	
2.		<p>Regresa al paso 2 del Flujo Básico: Gestionar cliente.</p> <p>Termina el CU.</p>
<b>Flujos alternos</b>		
<b>4a. Evento: Campo obligatorio vacío.</b>		
	<b>Actor</b>	<b>Sistema</b>
1.		<p>Muestra un mensaje de error indicando que hay campos obligatorios vacíos.</p> <p>(Ver prototipo de interfaz 4)</p> <p>Regresa al paso 3 del Flujo Básico: Adicionar</p>

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

		cliente.
<b>Sección 3: “Eliminar cliente”</b>		
<b>Flujo básico: Eliminar cliente</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	Selecciona la opción eliminar para el cliente que desee.	
2.		Muestra un mensaje de confirmación para saber si el usuario desea o no eliminar el cliente seleccionado.
3.	Selecciona la opción “Eliminar”. (A) (Ver prototipo de interfaz 5)	
4.		Elimina el cliente. Termina el CU.
<b>Prototipo de interfaz 5</b>		
 <p>¿Está seguro de que quiere eliminar el cliente?</p> <p>Eliminar (A)      Cancelar (B)</p>		
<b>Flujos alternos</b>		
<b>3a. Evento: Selecciona la opción “Cancelar”.</b>		
	<b>Actor</b>	<b>Sistema</b>
1.	Selecciona la opción “Cancelar”. (B) (Ver prototipo de interfaz 5)	
2.		Regresa al paso 2 del Flujo Básico: Gestionar cliente. Termina el CU.
<b>Relaciones</b>	<b>CU Incluidos</b>	No existen.
	<b>CU Extendidos</b>	No existen.

Se pueden consultar el resto de las descripciones de los casos de uso en el Anexo 2: Especificación de casos de uso.

### 2.5 Modelo de diseño

El Modelo de Diseño es una disciplina que no se puede obviar en el proceso de desarrollo del software. Es imprescindible para comprender la forma en que va a funcionar el sistema de acuerdo con los requisitos, lenguajes de programación, componentes reutilizables y tecnologías de interfaz de usuario que se eligieron para el desarrollo del mismo. Es una representación gráfica, mediante varios diagramas muy explícitos de la implementación del sistema (47).

#### 2.5.1 Descripción de los patrones arquitectónicos y de diseño

Los patrones ayudan al arquitecto a definir la composición y el comportamiento del sistema de software, y una combinación adecuada de ellos permite alcanzar los requerimientos de calidad. (47) A continuación, se describen los utilizados en el desarrollo de la propuesta de solución.

#### 2.5.2 Patrones arquitectónicos

Los patrones arquitectónicos expresan el esquema de organización estructural fundamental para sistemas de software. Provee un conjunto de subsistemas predefinidos, especifica sus responsabilidades e incluye reglas y pautas para la organización de las relaciones entre ellos. Propone que son plantillas para arquitecturas de software concretas, que especifican las propiedades estructurales de una y tienen un impacto en la arquitectura de subsistemas. La selección de un patrón arquitectónico es, por lo tanto, una decisión fundamental de diseño en el desarrollo de un sistema de software. (48)

Los patrones arquitectónicos:

- Definen la estructura básica de una aplicación.
- Pueden contener o estar contenidos en otros patrones.
- Proveen un subconjunto de subsistemas predefinidos, incluyendo reglas y pautas para su organización.
- Son una plantilla de construcción. (49)

El patrón arquitectónico seleccionado para el desarrollo del sistema fue el Modelo - Vista - Controlador (MVC). El mismo separa la lógica de negocio de la interfaz de usuario aumentando la posibilidad de evolución por separado de ambos aspectos en tres componentes diferentes. (Ver figura 4)

#### Patrón arquitectónico Modelo – Vista – Controlador

Mediante el uso de MVC es posible incrementar la reutilización y la flexibilidad de las aplicaciones debido a la separación de los aspectos antes mencionados. Las modificaciones a las vistas no afectan

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

en absoluto a los otros módulos de la aplicación. Este patrón consiste en separar el código de los proyectos en tres partes (50):

**Modelo:** En este componente se coloca todo el código relacionado con las bases de datos, es decir, consultas de entrada, de modificación y de salida retornando los resultados en la forma que se necesiten.

**Vista:** En esta parte se programa la parte visual del software como los formularios, la parte que visualiza el usuario.

**Controlador:** Es el componente que implementa la lógica del programa, encargada de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario. Es aquella que le pide al modelo los datos y los muestra en la vista.

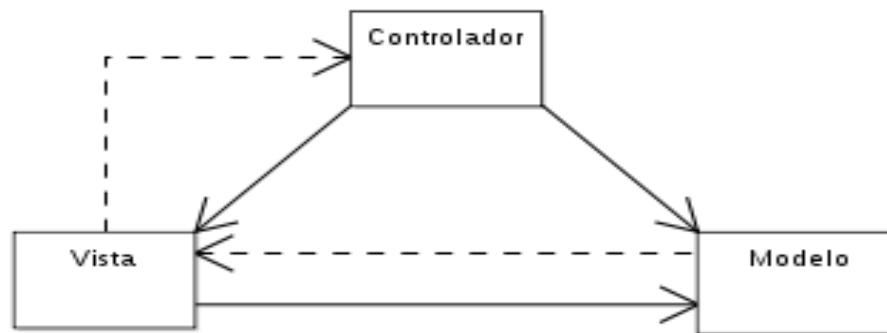


Figura 4: Patrón arquitectónico Modelo-Vista-Controlador

### 2.5.3 Patrones de diseño

Los patrones de diseño son el núcleo en las soluciones a problemas comunes en el desarrollo de software. Brindan una solución exitosamente probada para problemas similares en el desarrollo de software (51). Además, es una técnica muy eficaz para flexibilizar el código haciéndolo satisfacer ciertos criterios, así como permite una manera más práctica de describir ciertos aspectos de la organización de un programa. (52)

Entre los patrones de diseño más utilizados se encuentran los Patrones de Principios Generales para Asignar Responsabilidades (GRASP por sus siglas en inglés) y los GOF (Gang of Four o Banda de los Cuatro en español). Según (Larman, 2003) (53) los patrones GRASP poseen principios relacionados con la asignación de responsabilidades. Por otra parte, los patrones GOF surgen como una forma indispensable de enfrentarse a la programación. Estos se clasifican en patrones creacionales, estructurales y de comportamiento. A continuación, se muestran los patrones GRASP y GOF empleados.

### Patrones GRASP empleados

**Experto:** El uso de este patrón permite asignarle una responsabilidad a la clase que tiene la información necesaria para realizarla (53). En la aplicación el uso del mismo está implementado bajo el patrón arquitectónico MVC, por lo que cada clase tiene responsabilidades específicas donde contiene la información necesaria para realizarla, como acceder a la base de datos, mostrar la información o controlar las operaciones que realiza el sistema.

**Creador:** Este patrón permite asignarle a una clase la responsabilidad de crear una instancia de otra (53). Dentro del sistema este patrón se evidencia en las acciones de los controladores, las cuales del modelo o los formularios que representan las entidades crean objetos.

**Controlador:** Sirve para asignar la responsabilidad del sistema a una clase específica de recibir o manejar un mensaje de evento (53). Ejemplo de la utilización de este patrón se encuentra en la clase `comercializador_controller` que se encarga de la lógica del negocio. Esta clase utiliza información de la clase `comercializador_model` y envía datos a `comercializador_view`.

**Bajo Acoplamiento:** Se utiliza para soportar bajas dependencias entre las clases, o sea, mantener el bajo acoplamiento (53). En el sistema se evidencia el uso de este patrón pues cada clase solo crea instancias de otras que necesite. De esta forma, si se produce algún cambio en algunas de ellas, se afecte lo menos posible las otras clases.

**Alta cohesión:** Se utiliza para asignar una responsabilidad de manera que la cohesión permanezca alta. Un elemento que tiene responsabilidades altamente relacionadas, y que no hace una gran cantidad de trabajo, posee alta cohesión. Estos elementos pueden ser clases, subsistemas, etcétera (53). Un ejemplo de utilización de este patrón es cuando se definen las responsabilidades y tareas de cada clase en el sistema. La clase controlador se encarga de la lógica del negocio, el modelo del acceso a datos y la vista de la presentación de la información. Además, en Drupal cada módulo se encarga de hacer tareas específicas, sin altas dependencias, permitiendo que la modificación de la lógica del negocio de alguno no afecte el funcionamiento del resto.

### Patrones GOF empleados

**Fachada (Facade):** Proporciona una interfaz unificada de alto nivel para representar un subsistema y facilitar su uso (53). El patrón es utilizado en la generación de la licencia de un producto, donde el sistema utiliza el programa de generar licencia sin que el usuario tenga que interactuar con este, actuando como una fachada y haciendo más fácil su uso.

**Puente (Bridge):** Su propósito es separar una abstracción de su implementación para permitir que ambos puedan variar independientemente. Este patrón es usado para evitar que posibles cambios en la implementación de una abstracción afecten a los clientes o cuando se quiere ocultar la implementación de la vista de los clientes (53). La conexión de Drupal con la base de datos es una

abstracción que le permite al cliente ignorar el tipo de almacenamiento que utiliza, he incluso migrar de gestor de base de datos sin tener que modificar su código.

**Instancia única** (Singleton): Garantiza la existencia de una única instancia para una clase proporcionando un mecanismo de acceso global a dicha instancia (53). Se puede apreciar el uso de este patrón en la conexión con la base de datos que utiliza Drupal. El CMS crea un objeto único para la conexión y no se destruye hasta que no se atienden todas las peticiones a la base de datos.

**Acción** (Command): Patrón de comportamiento a nivel de objetos, su principio de funcionamiento es encapsular en un objeto la acción que satisface una petición. Proporciona una separación entre dicha acción y la petición que resuelve lo cual aumenta la flexibilidad en todo lo relativo con la acción (53). La utilización de este patrón se evidencia en la implementación de los ganchos (hook) necesarios para la comunicación del módulo con el núcleo de Drupal.

### 2.6 Diagrama de clases de diseño

Un diagrama de clases de diseño (DCD) representa las especificaciones de las clases e interfaces de software en una aplicación. Muestra las relaciones existentes entre las clases del sistema. Éste contiene además la clase servidora, así como los formularios que se muestran en estas. Sirve para dar una vista más específica de lo que el usuario va a ver y de lo que va a hacer el sistema internamente (53).

A continuación, se muestra el DCD para el CU Gestionar cliente. En este diagrama se encuentra la Server Page Index.php que se encarga de atender todas las peticiones para lo cual establece comunicación con el núcleo de Drupal y comercializador.module. Además, se muestra las clases Client Page que tienen asociados los formularios que envían los datos insertados por los usuarios a la Server Page comercializador.module, la cual se relaciona con las clases que intervienen en el funcionamiento del sistema. Los demás diagramas se encuentran representados en el Anexo 3: Diagramas de Clases del Diseño.

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

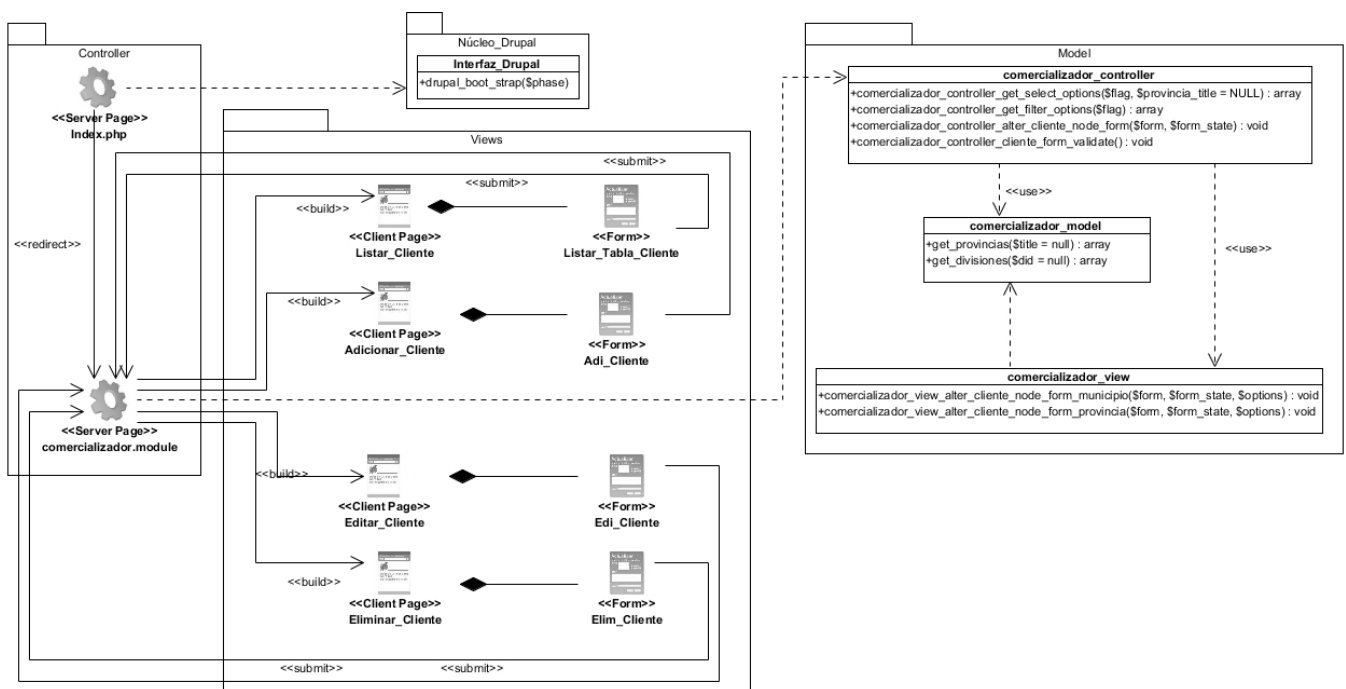


Figura 5: DCD del CU Gestionar cliente

### 2.7 Diagramas de interacción

EL término diagrama de interacción es una generalización de dos tipos de diagramas UML más especializados; los diagramas de colaboración y los diagramas de secuencia. Ambos pueden utilizarse para representar de forma similar interacciones de mensajes. Además, estos diagramas describen la manera en que colaboran grupos de objetos para cierto comportamiento (53). En el diseño según (Jacobson y otros, 2000) (46) se recomienda utilizar los diagramas de secuencias mostrándose la secuencia de interacciones detalladas y ordenadas en el tiempo.

#### 2.7.1 Diagrama de secuencia

Un diagrama de secuencia muestra las interacciones entre objetos, ordenadas en secuencia temporal durante un escenario<sup>3</sup> concreto. Es una representación que ejemplifica, en determinado escenario de un caso de uso, los eventos generados por actores externos, su orden y los eventos internos del sistema. (54). Para una mejor comprensión se continuará mostrando como ejemplo el diagrama de secuencia del CU: Gestionar cliente escenario adicionar cliente

<sup>3</sup> El escenario de un caso de uso es una instancia o trayectoria realizada por medio del uso: un ejemplo real de su ejecución.



## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

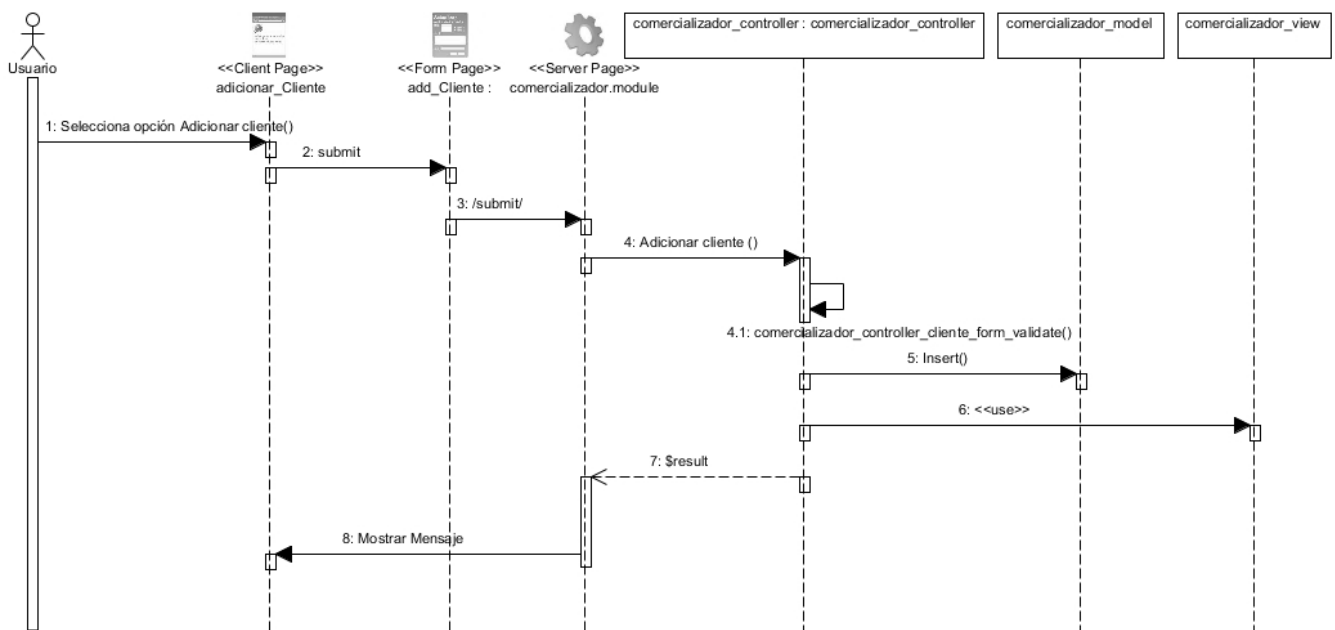


Figura 6: Diagrama de secuencia del CU Gestionar cliente: Escenario adicionar cliente

### 2.8 Modelo de datos

Un modelo de datos es la descripción de una base de datos. Típicamente permite describir las estructuras de datos de la base, su tipo, descripción y la forma en que se relacionan, restricciones de integridad entre otros, es factible pensar que un modelo de datos permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que se relacionan esos elementos entre sí. El modelo de Base de Datos del Sistema de información sobre las ventas de los productos de la cartera de Desoft, está compuesto por 5 tablas y se describe a continuación.

En la figura 7 se puede apreciar que las entidades del dominio del problema: CLIENTE, PRODUCTO y DIVISION se han representado como las entidades fuertes del modelo de datos del sistema. Las VENTAS son una entidad débil que se forma por la relación entre las entidades CLIENTE y PRODUCTO, con los atributos: fecha de la venta, tiempo de suscripción, licencia (atributo para guardar la cadena que se genere y se pueda recuperar cada vez que sea necesario) y la división que efectúa la venta. Cada venta origina la definición de una serie de plazos de pago cada uno con su monto a pagar; y en consecuencia se representan los PAGOS como las transferencias que se han efectuado desde la división que vende hacia la división dueña del producto.

A continuación, se muestra el modelo de datos del sistema propuesto en esta investigación:

## Capítulo 2: Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft.

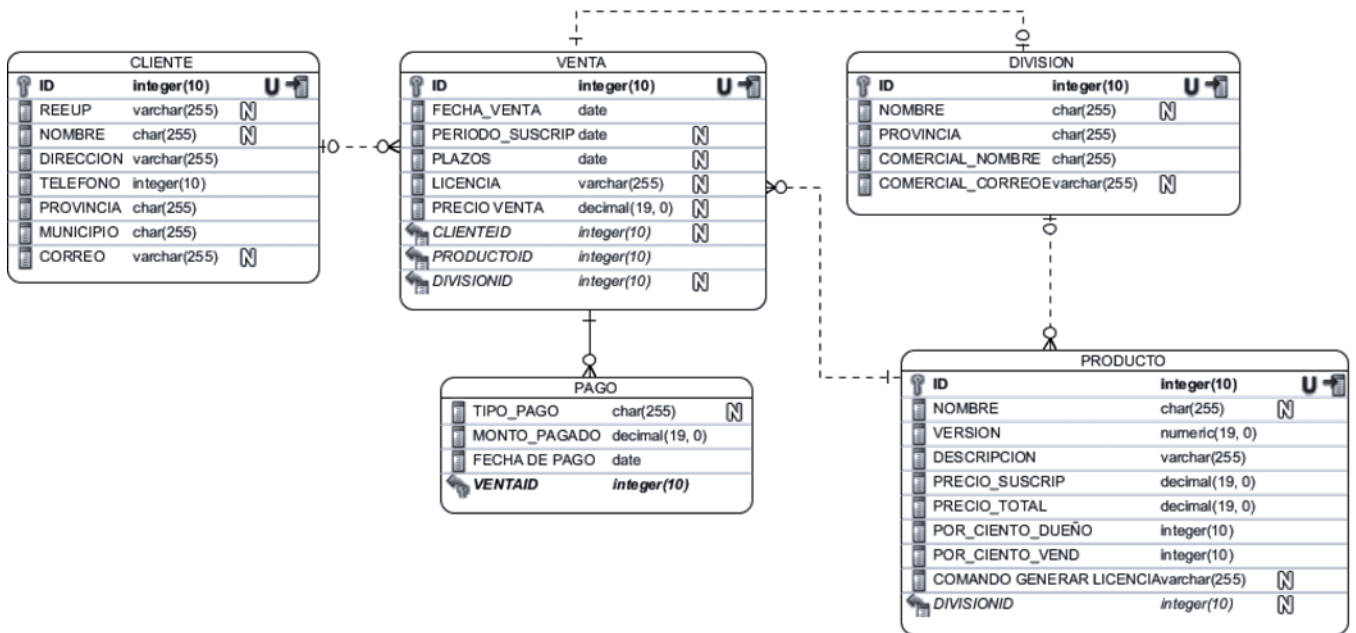


Figura 7: Modelo de datos del Sistema de información sobre las ventas de los productos de la cartera de Desoft

### 2.9 Conclusiones del capítulo

- La realización de la propuesta de un sistema de gestión de información sobre las ventas por suscripciones permite describir con claridad las necesidades del cliente y acotar el alcance de la solución a realizar.
- Mediante la entrevista realizada con el cliente se identificaron sus necesidades, obteniéndose los requisitos funcionales agrupados por un total de 14 casos de uso y 15 requisitos no funcionales.
- Los artefactos relacionados con el flujo de análisis y diseño, teniendo en cuenta la arquitectura MVC propuesta, permitieron obtener una mayor comprensión de cómo debe funcionar la aplicación, así como definir los principios que guían la implementación y organización de la misma.

## Capítulo 3: Implementación, pruebas y resumen del Manual de Usuario

En el presente capítulo se expone la documentación referente a la fase de implementación. Como resultado de esta etapa se muestra el diagrama de componentes. Por otra parte, se desarrollaron pruebas de Caja Negra a las interfaces del sistema y se documentan las mismas. Además, se hace un resumen del manual de usuario para el sistema implementado.

### 3.1 Modelo de implementación

En la etapa de implementación se generan una serie de artefactos como el modelo de implementación. Para ilustrar los detalles de esta etapa UML define varios diagramas, entre ellos el diagrama de componentes y el diagrama de despliegue. Según (Jacobson y otros, 2000) (46) el modelo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes. Además, describe la organización de estos componentes y cómo dependen unos de otros.

#### 3.1.1 Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones que indican la utilización de servicios ofrecidos por otros componentes. Un componente podría ser simples archivos, paquetes, bibliotecas, entre otros. Se realizan con el objetivo de poseer una vista de forma general del sistema a partir de las dependencias e integraciones de los componentes (53). En la siguiente figura se muestra el diagrama de componentes del sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft. Este diagrama contiene los ficheros .php que son las clases como el modelo y la vista implicadas en el funcionamiento del sistema. El fichero .module contiene funciones o ganchos (hooks) predefinidos por Drupal para ser utilizados durante una petición. También se encuentran el fichero .install donde se definen las operaciones necesarias para el funcionamiento inicial del módulo y el .info contiene información general del módulo como el nombre, las dependencias del mismo, las clases con que trabaja, etc. Por último, el archivo .js posee todas las funciones JavaScript del sistema.

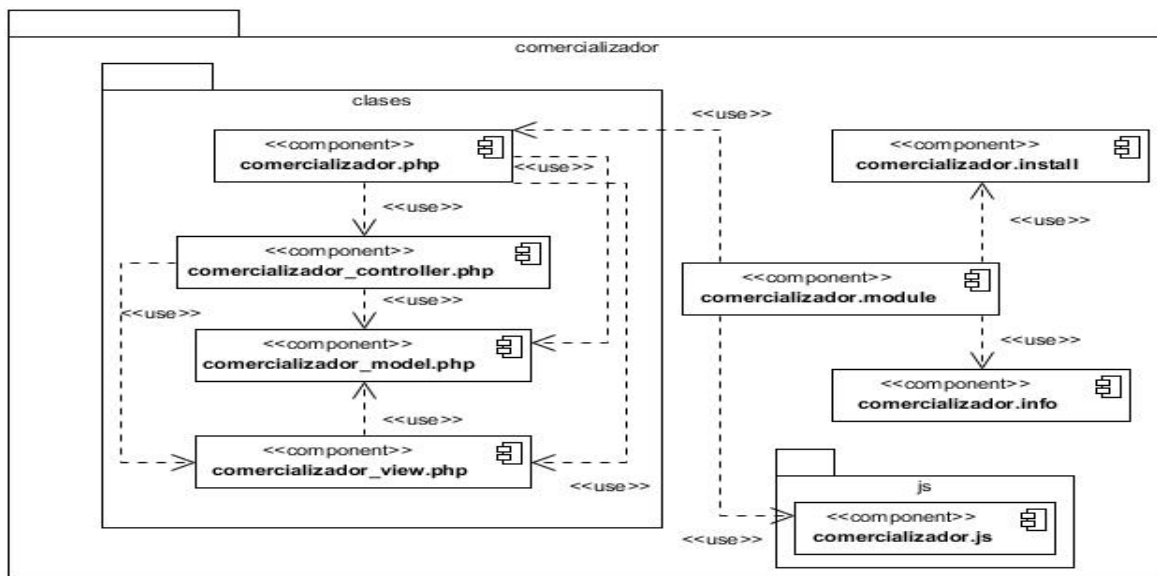


Figura 8: Diagrama de componentes

### 3.3.2 Modelo de despliegue

Un diagrama de despliegue representa los recursos de cómputo que utiliza el sistema. Estos recursos son llamados nodos que muestran su configuración y componentes. Las relaciones entre los nodos constituyen medios de comunicación (53). En la Fig. 8 se muestra el diagrama de despliegue del sistema implementado.

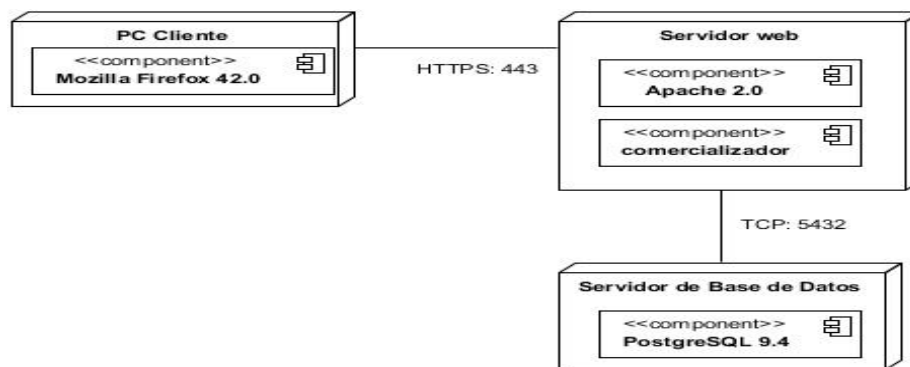


Figura 9: Diagrama de despliegue

**PC Cliente:** Representa las computadoras clientes que se conectan al servidor web mediante el protocolo HTTPS por el puerto 443. Este nodo tiene como componente el navegador web Mozilla Firefox 42.0.

**Servidor web:** Representa el servidor donde se encuentra instalada la aplicación web, debe tener el servidor Apache en su versión 2.0. Este accede al servidor de Base de Datos para el manejo de la información mediante el protocolo TCP por el puerto 5432.

**Servidor de Base de datos:** Es donde se almacena toda la información de la aplicación, administrada por el gestor PostgreSQL 9.4.

## 3.2 Modelo de pruebas

El resultado de la implementación se verifica en la etapa de prueba, por lo que planificar estas pruebas y manejar los resultados de forma que los defectos puedan ser arreglados es de gran importancia. El modelo de pruebas según (Jacobson y otros, 2000) (46) se describe como “una colección de casos de prueba, procedimientos de prueba y componentes de prueba”. Además, expresa que un caso de prueba incluye la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse, mientras que un procedimiento de prueba especifica cómo realizar uno o varios casos de prueba o partes de estos. Por otra parte, existen dos métodos fundamentales que se le aplican a un producto de software que utilizan diseño de casos de prueba, estas son: las pruebas de caja blanca y las pruebas de caja negra.

### 3.2.1 Pruebas de caja negra

Se escogió realizar al sistema las pruebas de caja negra. Éstas detectan errores en la implementación de los requisitos funcionales del software. Para esta prueba se utilizó la técnica de partición equivalente la cual divide el dominio de entrada en clases de datos a partir de las cuales pueden derivarse casos de prueba (55). A continuación, se muestra el caso de prueba para el CU Gestionar cliente.

Tabla 3: Diseño de caso de prueba para el CU Gestionar Cliente

Caso de prueba: Gestionar Cliente.				
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Resultados Esperados	Resultados Obtenidos
SC 1: Listar los datos de los clientes.	EC 1.1: Listar los datos de los clientes exitosamente.	El usuario selecciona la opción gestionar cliente y el sistema lista todos los datos de los clientes almacenados con las opciones de adicionar, editar y eliminar.	El sistema lista todos los clientes almacenados con las opciones de adicionar, editar y eliminar.	El sistema lista todos los clientes almacenados con las opciones de adicionar, editar y eliminar.
	EC 1.2: Listar fallidamente los datos de los clientes.	El usuario selecciona la opción gestionar cliente y el sistema no tiene ningún cliente almacenado.	El sistema muestra la lista vacía y habilita la opción “Adicionar cliente”	El sistema muestra la lista vacía pero no habilita la opción “Adicionar cliente”.
SC 2: Adicionar cliente	EC 2.1: Adicionar el cliente exitosamente.	El sistema muestra un formulario para adicionar un cliente, el usuario llena los datos y presiona el botón “Guardar”.	El sistema adiciona el cliente, se muestra un mensaje indicando que el cliente ha sido adicionado satisfactoriamente y se actualiza la lista de clientes.	El sistema adiciona el cliente, se muestra un mensaje indicando que el cliente ha sido adicionado satisfactoriamente y se actualiza la lista de clientes.
	EC 2.2: Adicionar el cliente	El usuario presiona el botón	El sistema muestra un	El sistema muestra un

### Capítulo 3: Implementación, pruebas y resumen del Manual de Usuario.

	estando vacíos campos obligatorios.	“Guardar” antes de completar todos los datos obligatorios y el sistema muestra un mensaje de error indicando que existen campos obligatorios por llenar.	mensaje indicando que existen campos obligatorios por llenar y señala en rojo los campos vacíos.	mensaje indicando que existen campos obligatorios por llenar y señala en rojo los campos vacíos.
	EC 2.3 Adicionar el cliente con caracteres no numéricos en el campo teléfono.	El usuario presiona el botón “Guardar” con caracteres no numéricos en el campo teléfono.	El sistema muestra un mensaje indicando que el campo teléfono debe incluir sólo números.	El sistema muestra un mensaje indicando que el campo teléfono debe incluir sólo números.
SC 3: Editar datos del cliente.	EC 3.1: Editar datos del cliente exitosamente.	El sistema muestra un formulario con los datos del cliente seleccionado, el usuario cambia los datos y presiona el botón “Guardar”.	El sistema modifica los datos del cliente, muestra un mensaje indicando que estos datos han sido modificados satisfactoriamente.	El sistema modifica los datos del cliente, muestra un mensaje indicando que estos datos han sido modificados satisfactoriamente.
	EC 3.2: Editar datos del cliente dejando vacíos campos obligatorios.	El usuario presiona el botón “Guardar” antes de completar todos los datos obligatorios y el sistema muestra un mensaje de error indicando que existen campos obligatorios por llenar.	El sistema muestra un mensaje indicando que existen campos obligatorios por llenar.	El sistema muestra un mensaje indicando que existen campos obligatorios por llenar.
	EC 2.3 Editar el cliente insertando caracteres no numéricos en el campo teléfono.	El usuario presiona el botón “Guardar” con caracteres no numéricos en el campo teléfono.	El sistema muestra un mensaje indicando que el campo teléfono debe incluir sólo números.	El sistema muestra un mensaje indicando que el campo teléfono debe incluir sólo números.
SC 4: Eliminar cliente.	EC 4.1: Eliminar el cliente exitosamente.	El usuario selecciona el cliente que desea eliminar y selecciona la opción “Eliminar”, el sistema muestra un mensaje de confirmación de la acción y si el usuario acepta se elimina el cliente, además se muestra un mensaje indicando que la acción fue realizada satisfactoriamente.	El sistema muestra un mensaje de confirmación, se elimina el cliente satisfactoriamente y se actualiza la lista de clientes.	El sistema muestra un mensaje de confirmación, se elimina el cliente satisfactoriamente y se actualiza la lista de clientes.
	EC 4.2: Cancelar la eliminación del cliente.	El usuario selecciona el cliente que desea eliminar y selecciona la opción “Eliminar”, el sistema muestra un mensaje de confirmación de la acción y si el usuario cancela no se realiza la eliminación y se regresa a la vista de gestionar clientes.	El sistema no elimina y muestra la lista de clientes.	El sistema no elimina y muestra la lista de clientes.

## Capítulo 3: Implementación, pruebas y resumen del Manual de Usuario.

**Tabla 4: Descripción de variables**

No	Nombre del campo	Clasificación	Valor nulo	Descripción
1	Nombre	Campo de texto	No	Campo para entrar el nombre del cliente.
2	REEUP	Campo de texto	No	Este campo es para entrar el código REEUP del cliente.
3	Dirección	Campo de texto	Si	Campo para entrar la dirección del cliente.
4	Teléfono	Campo de texto	Si	Campo para entrar el teléfono del cliente.
5	Correo electrónico	Campo de texto	No	Campo para entrar el correo electrónico del cliente.
6	Provincia	Campo seleccionable	Si	Este campo contiene una lista de las provincias existentes para seleccionar la del cliente.
7	Municipio	Campo seleccionable	Si	Este campo contiene una lista de los municipios existentes para seleccionar el del cliente.

La siguiente tabla contiene aspectos como V (Válido), I (Inválido), N/A (No Aplica) que indican el valor del dato a utilizar durante la prueba.

**Tabla 5: Matriz de datos para el escenario adicionar cliente**

Escenario	Nombre	REEUP	Dirección	Teléfono	Correo electrónico	Provincia	Municipio	Respuesta del sistema
Adicionar el cliente exitosamente.	V: Agropecuaria Matanzas	V: 101.20.9	N/A	V:8352101	V: ana@vcl.desoft.cu	N/A	N/A	Se añade el cliente y se muestra un mensaje de confirmación
Adicionar el cliente estando vacíos campos obligatorios.	I: Campo vacío V: Agropecuaria Matanzas	I: Campo vacío V: 101.20.9	N/A N/A	N/A N/A	I: Campo vacío V: ana@vcl.desoft.cu	N/A N/A	N/A N/A	El sistema muestra un mensaje indicando que existen campos obligatorios por llenar y señala en rojo los campos vacíos.
Adicionar el cliente con caracteres no numéricos en el campo	N/A	N/A	N/A	I:835we2	N/A	N/A	N/A	El sistema muestra un mensaje indicando que el campo

### Capítulo 3: Implementación, pruebas y resumen del Manual de Usuario.

teléfono.								teléfono debe incluir sólo números.
-----------	--	--	--	--	--	--	--	-------------------------------------

#### Resultados de las pruebas:

A partir del diseño y ejecución de los casos de prueba, introduciendo juegos de datos, tanto correctos como incorrectos, se detectaron errores tales como: el sistema no presentaba como disponible la opción de adicionar cliente cuando la lista de clientes estaba vacía, errores de idioma, validación de los datos de entrada, errores de redacción y funcionalidades incorrectas. Como resultado de la primera iteración se obtuvieron 7 (siete) no conformidades como se muestra en la figura 10. Luego de corregidas se procedió a realizar una segunda iteración la cual arrojó resultados satisfactorios.

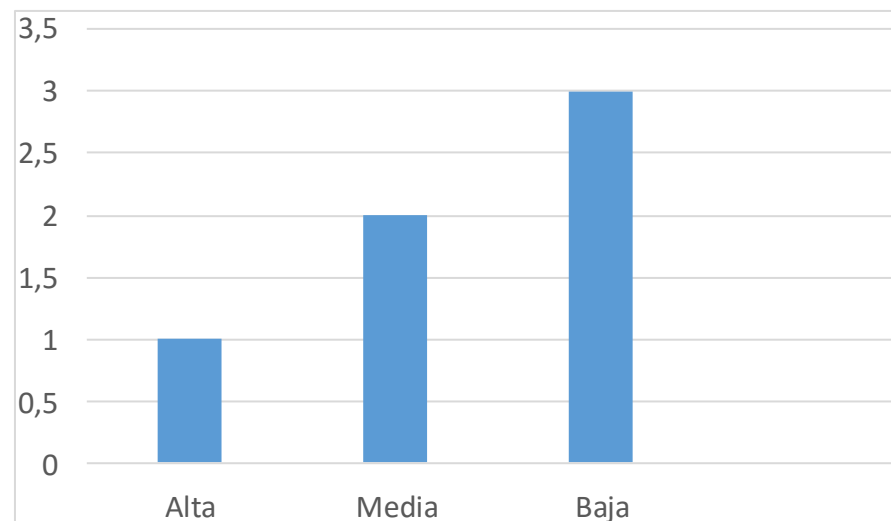


Figura 10: Prueba de caja negra 1era iteración



### 3.2.2 Pruebas de rendimiento

Las pruebas de rendimiento consisten en someter el sistema a altas cargas de trabajo mediante la simulación de la actividad de los usuarios reales en el sistema. (56) Se decidió realizar al sistema las pruebas de rendimiento carga y estrés pues estas están enfocadas en determinar la capacidad de la aplicación de recibir múltiples peticiones sin que se vea afectada. Para la elaboración de estas pruebas se escogió la herramienta JMeter, la cual permite medir la capacidad de carga de una aplicación. Además, con ella es posible determinar dado un número de usuarios los tiempos de repuestas del sistema y un número real de transacciones procesadas por unidad de tiempo.

Estas pruebas se realizaron en un servidor con microprocesador Intel(R) Pentium(R) II a 2.0 MHz, con 4 GB de memoria RAM donde se encontraban instalados el servidor Apache 2 y PostgreSQL 9.4. y con el sistema operativo Windows 10. Para la ejecución de estas se tiene en cuenta el diseño de un plan de pruebas basado en las acciones que el usuario puede realizar al conectarse a la aplicación. Con una muestra de 30 usuarios conectados concurrentemente se obtiene que para un total de 240 muestras que se le realizaron al sistema el rendimiento es de 45,7 peticiones por segundo, con un porcentaje de 0% de errores para cada petición realizada y un tiempo de respuesta de 7196 milisegundos.

Durante la realización de estas pruebas el sistema manifestó un nivel de respuesta aceptable cumpliendo con las características de los requisitos no funcionales de eficiencia especificados en el capítulo 2.

### 3.3 Manual de Usuario

Para acceder al sistema el usuario debe estar autenticado, en la página principal se muestran aleatoriamente los productos adicionados, su versión, la división propietaria y su descripción.



Figura 11: Página Principal

### 1. Contenido de la página principal.

En la figura 12 se muestra el menú de gestión del sistema implementado, el cual contiene las operaciones de gestionar división, gestionar producto, gestionar ventas, gestionar clientes, gestionar provincias y gestionar usuarios, donde el usuario autenticado puede seleccionar la operación deseada.



Figura 12: Menú de gestión

### 1. Menú de Gestión, para seleccionar la operación deseada.

En la figura 13 se muestra la ventana de gestionar clientes, donde aparecen las operaciones de adicionar, editar y eliminar un cliente, además muestra el listado de los clientes ya adicionados.

Inicio / Gestión / Gestionar cliente

## Gestionar cliente

Adicionar cliente 1

Nombre

Buscar Reiniciar

2	Nombre	REEUP	Correo	Dirección	Provincia	Municipio	Teléfono	3	operaciones	4
	Hospital Cienfuegos	32423	lolo@uci.cu	calle 3	Cienfuegos	Cumanayagua	4 365 765		editar   eliminar	
	Agricultura Villa Clara	123456	lolo1@uci.cu	calle 32	Villa Clara	Cifuentes	436 574 222		editar   eliminar	
	Mesa Redonda	9123	afigueroa@vcl.cu	AVE Bolleros	Cienfuegos	Cienfuegos	8 372 101		editar   eliminar	
	Manolo	werty	manolo@uci.cu	calle 4ta	Matanzas	Matanzas	3 456 789		editar   eliminar	
	Pepe	123	pepe@uci.cu	calle 23	Granma	Granma	123 456		editar   eliminar	
	Tabaco Torcido	1234	aut@vcl.cu	adghj45	Villa Clara		123 456		editar   eliminar	

Figura 13: Gestionar Cliente

1. Opción para adicionar un cliente.
2. Muestra la lista de clientes adicionados.
3. Opción de editar el cliente seleccionado.

### 4. Opción de eliminar el cliente seleccionado.

La figura 14 muestra el menú de reportes, el cual contiene el reporte de deudas que tienen las otras divisiones y el reporte VLO, además se visualiza la ventana del reporte de deudas donde se muestran los datos relacionados con este reporte y la opción de exportar el reporte a excel.

División	Producto	Cliente	Total pagado	MONTO DE LA DEUDA
Matanzas	Energux	Manolo	0	136.848
Matanzas	Energux	Agricultura Villa Clara	228.08	-108.08

Figura 14: Reporte de deudas

1. Menú de reportes.
2. Listado de los datos del reporte seleccionado.
3. Opción de exportar a excel.

Se muestra en la figura 15 la vista de los pagos realizados con las operaciones de adicionar, editar y eliminar un nuevo pago.

Tipo de pago	Fecha de Pago (mes/dia/año)	Monto pagado	operaciones
división - división	04/06/2016	200	editar eliminar

Tipo de pago	Fecha de Pago (mes/dia/año)	Monto pagado	operaciones
cliente - división	04/14/2016	100	editar eliminar

Figura 15: Vista de Pagos

1. Muestra los datos de los pagos adicionados.

2. Opción de adicionar un nuevo pago.
3. Opción de editar el pago seleccionado.
4. Opción de eliminar el pago seleccionado.

En la figura 16 se muestra el listado de las ventas de los productos para la generación de licencias y la opción de enviarla por correo electrónico a las personas deseadas, estableciendo por defecto la dirección del comercial de la división que solicita, o del cliente en el caso en que la venta se realizara por la división propietaria.

The screenshot shows a web application interface for 'Listado de ventas'. At the top, there is a navigation bar with 'INICIO', 'GESTIÓN', 'REPORTES', 'PAGOS', and 'LICENCIAS'. Below the navigation bar, the breadcrumb 'Inicio / Listado de ventas' is visible. The main heading is 'Listado de ventas' with a circled '1' next to it. Below the heading, there are two dropdown menus: 'Producto vendido' (set to 'Todos') and 'Division vendedora' (set to 'Todos'). There are also 'Buscar' and 'Reiniciar' buttons. Below these filters is a table with the following data:

División vendedora	Cliente	Producto	Licencia	Fin de suscripción	Operaciones
Matanzas	Manolo	Energux	Sin licencia	07/21/2016	Generar licencia
Desoft Villa Clara	hospital cienfuegos	Energux	Asignada	07/05/2016	Generar licencia   Enviar por correo
Matanzas	Agricultura Villa Clara	Energux	Asignada	07/04/2016	Generar licencia   Enviar por correo
Desoft Villa Clara	Agricultura Villa Clara	Energux	Sin licencia	05/02/2016	Generar licencia
Desoft Villa Clara	hospital cienfuegos	Energux	Asignada	06/30/2016	Generar licencia   Enviar por correo

Numbered callouts in the image: '1' points to the title 'Listado de ventas'; '2' points to the 'Operaciones' column header; '3' points to the 'Enviar por correo' option in the first row of the table.

Figura 16: Generación de Licencias

1. Listado de las ventas realizadas.
2. Opción de generar las licencias para los productos de las ventas realizadas.
3. Opción de enviar por correo electrónico la licencia generada.

### 3.4 Validación del sistema en Desoft Villa Clara

El sistema se instaló en el Departamento Comercial de Desoft Villa Clara y fue probado en todas sus funcionalidades por la gestora comercial Elismagui Pérez Olivera. Esta compañera solicitó la inclusión de un nuevo campo "precio de la venta" que es diferente al precio del producto, para poder reflejar la realización de descuentos comerciales que frecuentemente se utilizan en los procesos de venta, ante esta solicitud se procedió con la actualización del esquema de la base de datos y la aplicación del valor de este nuevo campo en el cálculo de las deudas. También se solicitó que se contemplara la utilización de un generador de licencias diferente para cada producto, aspecto que también se adicionó en el sistema. Se realizaron otras pruebas de comportamiento como el acceso al sistema desde múltiples usuarios simultáneos. Se solicitó un cambio de aspecto en los reportes y se detectó un error de funcionalidad que también fue corregido.

El director del Departamento Comercial dio una opinión satisfactoria del sistema, resaltando la utilidad que tiene el mismo para la gestión comercial y para el control de los ingresos por suscripciones. Los gestores comerciales continuaron con el uso del sistema introduciendo datos reales.

En general fue de buena aceptación y se plantearon realizar en el futuro otras adiciones como un reporte consolidado de ventas realizadas por las otras divisiones, con filtros y búsquedas.

### **3.5 Conclusiones del capítulo**

- En este capítulo se definen temas relacionados con los procesos de implementación y pruebas de la aplicación que posibilitan la realización de los requisitos del sistema y la verificación del correcto funcionamiento de los mismos. La elaboración del diagrama de componente en la investigación proporciona una vista de forma general del sistema a partir de las dependencias e integraciones de los mismos.
- La aplicación de las pruebas de caja negra aplicando la técnica de partición equivalente, favorece la identificación de 7 no conformidades en su primera iteración para poder darles solución a las mismas.
- Someter el sistema a pruebas de rendimiento permitió comprobar la respuesta del mismo ante situaciones extremas.
- La creación del manual de usuario proporciona una guía para usar de manera efectiva las funcionalidades del sistema.
- Mediante la validación del sistema en la división Desoft Villa Clara permitió conocer la aceptación de los usuarios resaltando la utilidad del producto.

### Conclusiones generales

La presente investigación arroja como principal resultado, el desarrollo de un sistema para la gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft, a partir de lo cual se arriban a las siguientes conclusiones:

- Se analizaron las características y limitaciones de las soluciones existentes referentes a los sistemas de información de ventas, evidenciándose la necesidad de construir el sistema, debido a que los existentes no reunían las funcionalidades que se requieren.
- Se caracterizaron las herramientas y tecnologías a utilizar, determinándose la base tecnológica necesaria para desarrollar el sistema propuesto.
- Se implementó el sistema para la gestión de información de ventas y entregas de licencias de los productos de la cartera de Desoft que contribuye a mejorar la velocidad en la entrega de las licencias y la eficiencia en el manejo de los datos sobre las transferencias entre divisiones.
- La aplicación de pruebas permitió validar el correcto funcionamiento del sistema desarrollado simulando ambientes reales en diferentes entornos de ejecución.
- Se realizó la validación del sistema en la división Desoft Villa Clara obteniendo la satisfacción del personal interesado.

### Recomendaciones

Luego de haber concluido con el desarrollo de la primera versión del Sistema de gestión de información sobre las ventas por suscripciones de los productos de la cartera de Desoft, se proponen las siguientes recomendaciones:

- Adicionar al sistema nuevos reportes como el consolidado de ventas realizadas por otras divisiones, con filtros y búsquedas para aumentar la usabilidad del producto.
- Desplegar el sistema en una infraestructura que posibilite el uso del mismo por todas las divisiones de Desoft luego de un periodo de prueba en la división territorial Villa Clara.

## Referencias bibliográficas

1. *PRACTICAS DE APRENDIZAJE ORGANIZACIONAL EN EMPRESAS DEL SECTOR DE LA INFORMÁTICA Y LAS COMUNICACIONES*. Blanco Rosales, Dr.C Humberto , Deulofeu Crespo, Dra.C Maria E. y Rodríguez González, Lic.Gisela .
2. **DESOFT**. *Expediente de Perfeccionamiento Empresarial*. Cuba : s.n., Septiembre 2015.
3. **Alvarez de Zayas, Carlos**. *Metodología de la investigación científica*. Santiago de Cuba, Cuba : s.n., 1995.
4. **Nocedo, Irma y otros**. *Metodología de la investigación educacional. Segunda parte*. La Habana : Félix Varela, 2001.
5. **Villanueva, Darío** . Diccionario de la lengua española | Real Academia Española. [En línea] Madrid: Espasa, octubre de 2014. <http://www.rae.es/obras-academicas/diccionarios/diccionario-de-la-lengua-espanola>.
6. **definiciones-de**. Qué significa suscribir - Información y significado de suscribir. [En línea] Diccionario de ALEGSA, 28 de 08 de 2010. [Citado el: 12 de noviembre de 2015.] <http://www.definiciones-de.com/Definicion/de/suscribir.php>.
7. **De Borja de Carlos Martín-Lagos, Francisco**. *Sistematización de la Función Comercial*. s.l. : Netbiblo, 2008.
8. **conceptodefinicion.de**. ¿Qué es Licencia? - Su Definición, Concepto y Significado. [En línea] Venemedia, 19 de enero de 2015. [Citado el: 16 de noviembre de 2015.] <http://conceptodefinicion.de/licencia/>.
9. **TheFreeDictionary**. Licencia s. f. [En línea] Diccionario Manual de la Lengua Española Vox. Larousse Editorial, S.L., 2007. [Citado el: 16 de noviembre de 2015.] <http://es.thefreedictionary.com/licencia>.
10. **DESOFT**. *PROCEDIMIENTO GENERAL DE MERCADOTECNIA*. 2012.
11. **TheFreeDictionary**. Reporte s. m. [En línea] Diccionario Manual de la Lengua Española Vox. Larousse Editorial, S.L., 2007. [Citado el: 16 de noviembre de 2015.] <http://es.thefreedictionary.com/Reporte>.
12. **Quees.la**. ¿Qué es reporte? Definición, concepto y significado. [En línea] 24 de 04 de 2015. [Citado el: 16 de noviembre de 2015.] <http://quees.la/reporte/>.



## Referencias bibliográficas

---

13. **Rouse, Margaret.** ¿Qué es CRM (Gestión de relaciones con los clientes)? [En línea] noviembre de 2006. [Citado el: 15 de febrero de 2016.] Whatls.com.
14. *Gestión comercial ventas, procesos, inductores, métricas y efectividad.* **Quiroz Leyton, Osvaldo.** Santiago de Chile, South America : s.n., 2012.
15. *Managing Customer Relationships: A Strategic Framework.* **Don Peppers y Martha Rogers,** **Wiley.** s.l. : Second Edition, 2011.
16. *Metodología para selección de sistemas ERP Reportes Técnicos en Ingeniería de Software.* **Chiesa, Florencia.** 1, 2004, Vol. 6. ISSN: 1668-3137.
17. **Fabien.** OpenERP is an Open Source Business Applications Suite. *This project group contains all OpenERP-related projects.* [En línea] 24 de 07 de 2008. [Citado el: 16 de febrero de 2016.] <https://launchpad.net/openobject>.
18. **Liémance, Laurent.** OpenERP Server, Addons, GTK Client. [En línea] Copyright (C) Free Software Foundation, 29 de June de 2007. [Citado el: 16 de febrero de 2016.] <https://doc.odoo.com/v6.1/legal/license.html>.
19. **Simmons, Geoff , Armstrong, Gillian y Durkin, Mark.** A Conceptualization of the Determinants of Small Business Website Adoption:Setting the Research Agenda. [En línea] 21 de Mayo de 2009. [Citado el: 12 de noviembre de 2015.] <http://studysites.uk.sagepub.com/chaston/20Simmons%20et%20al.pdf>. 389.
20. **SistemasPOS.** Sistemas POS para Alimentos y Bebidas, Hotelería y Retail - CSI SA Soluciones. [En línea] Carrera 43 A # 7-50 Oficina 1403 – Torre Empresarial Dann Carlton – Medellín-Colombia Teléfono (57- 4) 444 3 555. [Citado el: 17 de febrero de 2016.] <http://www.csisasoluciones.com/quienes-somos>.
21. **ManualdeUsuario.** *Manual de Usuario. Sistema de Comercialización de Softwares.* Cuba, Villa Clara : s.n., Octubre del 2007.
22. *Metodología del análisis estructurado de sistemas.* **AREBA, J. B. D.** 8468, Madrid : ALCOBENDAS, 2001, Vol. 84. 043-06 .
23. **Pressman, Roger S.** *"Ingeniería de Software: un enfoque práctico". Quinta Edición.* s.l. : McGraw-Hill, 2001.
24. *"Mantenimiento Perfectivo en un Gestor de Contenidos. Una Experiencia para la Especificación de Consultas Dinámicas".* **Alfonzo, Mariño, Godoy.** 312, Argentina : Revista Latinoamericana de Ingeniería de Software, 2012, Vol. 310. ISSN 2314-2642.

## Referencias bibliográficas

---

25. **Medina González , Luz Edith** . Open Up: Metodología OpenUP. *Metodología Open UP*. [En línea] Blogger, domingo, 23 de febrero de 2014. [Citado el: 14 de febrero de 2016.] <http://openup3.blogspot.com/2014/02/metodologia-open-up.html>.
26. **Booch, Grady**. *Object-Oriented Analysis and Design with Applications*. s.l. : Addison-Wesley, 2004.
27. **Kendall, Kenneth y Kendall, Julie**. *Análisis Y Diseño de Sistemas*. 3ra Edición. México : PrenticeHall, 2003.
28. **International, Visual Paradigm**. Visual Paradigm. *Visual Paradigm*. [En línea] 2008. [Citado el: 11 de 3 de 2015.] <http://www.visual-paradigm.com/product/vpuml/features/>.
29. **Technologies, Abitzar Learning**. Curso Modelado de Negocio con BPMN 2.0 y UML. [En línea] 2011. [Citado el: 3 de 06 de 2016.] <http://www.milestone.com.mx/CursoModeladoNegociosBPMN.htm>.
30. **Gil, Fran**. *Experto en Drupal 7 Nivel Inicial*. 2012.
31. **Hispano, Drupal**. Comunidad de usuarios de Drupal. . [En línea] 11 de Abril de 2005 . [Citado el: 11 de noviembre de 2015.] <http://drupal.org.es/drupa..>
32. **Riquelme, S. L. and Morales, M. C**. *Sistema Gestor de Base de Datos Relacionales*. 2010. 14.
33. **GUERRERO, R. M**. Sobre PostgreSQL. [En línea] 2009. [Citado el: 26 de enero de 2015.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
34. **Geschwinde, E. y Schöenig, HJ**. *Postgresql: Developer's Handbook*. . 2002.
35. **POSTGRESQL**. pgAdmin Introduction. [En línea] 2012. [Citado el: 26 de enero de 2016.] <http://www.pgadmin.org/> 44.
36. **ArPUG**. Grupo de usuarios PosgreSQL de Argentina. [En línea] 2008. [Citado el: 21 de Noviembre de 2015.] [http://www.arpug.com.ar/trac. .](http://www.arpug.com.ar/trac.)
37. **Welling, Luke y Thomson, Laura**. *Desarrollo Web con PHP y MySQL*. España : s.n., 2005.
38. **Sánchez Maza, Miguel Ángel**. *Javascript. Innovacion and cualificacion*. 2001.
39. **Smith, K**. *Ajax-style web development*. 2006.
40. **Converse, Tim, Park, Joyce y Morgan, Clark**. *PHP5 and MySQL Bible*. Indiana: Wiley Publishing. 2004.
41. **NetBeans**. NetBeans. *NetBeans*. [En línea] 2015. [Citado el: 15 de 1 de 2015.] <https://www.netbeans.org/features/index.html>.

## Referencias bibliográficas

---

42. **Collins-Sussman, Ben, Fitzpatrick, Brian W. y Pilato, C. Michael.** Control de versiones con Subversion. [En línea] 2004. [Citado el: 25 de noviembre de 2015.] <http://svnbook.red-bean.com/nightly/es/svn-ch-1-sect-1.html>.
43. **Samespinosa.** Modelo de Negocio. [En línea] Business & Mgmt. S.I., 2009. [Citado el: 31 de 03 de 2016.] <http://www.slideshare.net/samespinosa/mapa-de-procesos.1053479>.
44. **Martínez, Prof. Ivette Carolina.** CLASE 6: Modelo Conceptual/ Modelo de Dominio . [En línea] 2003. [Citado el: 01 de 04 de 2016.] [http://ldc.usb.ve/~martinez/cursos/ci3715/clase6\\_AJ2010.pdf](http://ldc.usb.ve/~martinez/cursos/ci3715/clase6_AJ2010.pdf).
45. **Pressman, Roger S.** *Software engineering: a practitioner's approach. 7th ed.* 2010.
46. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software.* Madrid : Addison Wesley, 2000.
47. **Kruchten.** *The Rational Unified Process: An Introduction.* 2000.
48. **Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P, y Stal, M, Pattern.** *Oriented Software Architecture. A System of Patterns.* Inglaterra : John Wiley & Sons, 1996.
49. **Cardoso, E, Camacho, F, y Nuñez, G.** *ARQUITECTURAS DE SOFTWARE. GUÍA DE ESTUDIO.* 2004.
50. **Venete, Adriana.** *Introducción a los Patrones de Arquitectura.* 2011.
51. **Microsoft.** [En línea] 2013. [Citado el: 01 de 04 de 2016.] <http://msdn.microsoft.com/eses/library/bb972240.aspx> .
52. **Gamma.** *Design patterns.* s.l. : AddisonWesley, 2013.
53. **Larman, Craig.** *UML y patrones. 2da Edición.* México : Prentice Hall, 2003.
54. —. *UML y Patrones. Introducción al análisis y diseño orientado a objetos, 1ra Edición.*
55. **Pressman, Roger S.** *Ingeniería del Software: Un Enfoque Práctico. 6ta Edición.* 2005.
56. —. *Ingeniería del Software. Un enfoque práctico.* España : Mc Graw Hill, 5ta edition, 2001. 8448132149. 323 pp. .
57. **Herrero Palomo, Julián.** *Administración, gestión y comercialización en la pequeña empresa.* s.l. : Editorial Paraninfo, 2001.
58. **Domínguez, Rubio P.** *Introducción a la gestión Empresarial.* s.l. : Edición electrónica, 2006.

## Bibliografía

1. *PRACTICAS DE APRENDIZAJE ORGANIZACIONAL EN EMPRESAS DEL SECTOR DE LA INFORMÁTICA Y LAS COMUNICACIONES*. **Blanco Rosales, Dr.C Humberto , Deulofeu Crespo, Dra.C Maria E. y Rodríguez González, Lic.Gisela .**
2. **DESOFT**. *Expediente de Perfeccionamiento Empresarial*. Cuba : s.n., Septiembre 2015.
3. **Alvarez de Zayas, Carlos**. *Metodología de la investigación científica*. Santiago de Cuba, Cuba : s.n., 1995.
4. **Nocedo, Irma y otros**. *Metodología de la investigación educacional. Segunda parte*. La Habana : Félix Varela, 2001.
5. **Villanueva, Darío .** Diccionario de la lengua española | Real Academia Española. [En línea] Madrid: Espasa, octubre de 2014. <http://www.rae.es/obras-academicas/diccionarios/diccionario-de-la-lengua-espanola>.
6. **definiciones-de**. Qué significa suscribir - Información y significado de suscribir. [En línea] Diccionario de ALEGSA, 28 de 08 de 2010. [Citado el: 12 de noviembre de 2015.] <http://www.definiciones-de.com/Definicion/de/suscribir.php>.
7. **De Borja de Carlos Martín-Lagos, Francisco**. *Sistematización de la Función Comercial*. s.l. : Netbiblo, 2008.
8. **conceptodefinicion.de**. ¿Qué es Licencia? - Su Definición, Concepto y Significado. [En línea] Venemedia, 19 de enero de 2015. [Citado el: 16 de noviembre de 2015.] <http://conceptodefinicion.de/licencia/>.
9. **TheFreeDictionary**. Licencia s. f. [En línea] Diccionario Manual de la Lengua Española Vox. Larousse Editorial, S.L., 2007. [Citado el: 16 de noviembre de 2015.] <http://es.thefreedictionary.com/licencia>.
10. **DESOFT**. *PROCEDIMIENTO GENERAL DE MERCADOTECNIA*. 2012.
11. **TheFreeDictionary**. Reporte s. m. [En línea] Diccionario Manual de la Lengua Española Vox. Larousse Editorial, S.L., 2007. [Citado el: 16 de noviembre de 2015.] <http://es.thefreedictionary.com/Reporte>.
12. **Quees.la**. ¿Qué es reporte? Definición, concepto y significado. [En línea] 24 de 04 de 2015. [Citado el: 16 de noviembre de 2015.] <http://quees.la/reporte/>.
13. **Rouse, Margaret**. ¿Qué es CRM (Gestión de relaciones con los clientes)? [En línea] noviembre de 2006. [Citado el: 15 de febrero de 2016.] Whatls.com.

## Bibliografía

---

14. *Gestión comercial ventas, procesos, inductores, métricas y efectividad*. **Quiroz Leyton, Osvaldo**. Santiago de Chile, South America : s.n., 2012.
15. *Managing Customer Relationships: A Strategic Framework*. **Don Peppers y Martha Rogers, Wiley**. s.l. : Second Edition, 2011.
16. *Metodología para selección de sistemas ERP Reportes Técnicos en Ingeniería de Software*. **Chiesa, Florencia**. 1, 2004, Vol. 6. ISSN: 1668-3137.
17. **Fabien**. OpenERP is an Open Source Business Applications Suite. *This project group contains all OpenERP-related projects*. [En línea] 24 de 07 de 2008. [Citado el: 16 de febrero de 2016.] <https://launchpad.net/openobject>.
18. **Liémance, Laurent**. OpenERP Server, Addons, GTK Client. [En línea] Copyright (C) Free Software Foundation, 29 de June de 2007. [Citado el: 16 de febrero de 2016.] <https://doc.odoo.com/v6.1/legal/license.html>.
19. **Simmons, Geoff , Armstrong, Gillian y Durkin, Mark**. A Conceptualization of the Determinants of Small Business Website Adoption:Setting the Research Agenda. [En línea] 21 de Mayo de 2009. [Citado el: 12 de noviembre de 2015.] <http://studysites.uk.sagepub.com/chaston/20Simmons%20et%20al.pdf>. 389.
20. **SistemasPOS**. Sistemas POS para Alimentos y Bebidas, Hotelería y Retail - CSI SA Soluciones. [En línea] Carrera 43 A # 7-50 Oficina 1403 – Torre Empresarial Dann Carlton – Medellín-Colombia Teléfono (57- 4) 444 3 555. [Citado el: 17 de febrero de 2016.] <http://www.csisasoluciones.com/quienes-somos>.
21. **ManualdeUsuario**. *Manual de Usuario. Sistema de Comercialización de Softwares*. Cuba, Villa Clara : s.n., Octubre del 2007.
22. *Metodología del análisis estructurado de sistemas*. **AREBA, J. B. D.** 8468, Madrid : ALCOBENDAS, 2001, Vol. 84. 043-06 .
23. **Pressman, Roger S.** "*Ingeniería de Software: un enfoque práctico*". Quinta Edición. s.l. : McGraw-Hill, 2001.
24. "*Mantenimiento Perfectivo en un Gestor de Contenidos. Una Experiencia para la Especificación de Consultas Dinámicas*". **Alfonzo, Mariño, Godoy**. 312, Argentina : Revista Latinoamericana de Ingeniería de Software, 2012, Vol. 310. ISSN 2314-2642.
25. **Medina González , Luz Edith** . Open Up: Metodología OpenUP. *Metodología Open UP*. [En línea] Blogger, domingo, 23 de febrero de 2014. [Citado el: 14 de febrero de 2016.] <http://openup3.blogspot.com/2014/02/metodologia-open-up.html>.

26. **Booch, Grady.** *Object-Oriented Analysis and Design with Applications*. s.l. : Addison-Wesley, 2004.
27. **Kendall, Kenneth y Kendall, Julie.** *Análisis Y Diseño de Sistemas. 3ra Edición*. México : PrenticeHall, 2003.
28. **International, Visual Paradigm.** Visual Paradigm. *Visual Paradigm*. [En línea] 2008. [Citado el: 11 de 3 de 2015.] <http://www.visual-paradigm.com/product/vpuml/features/>.
29. **Technologies, Abizar Learning.** Curso Modelado de Negocio con BPMN 2.0 y UML. [En línea] 2011. [Citado el: 3 de 06 de 2016.] <http://www.milestone.com.mx/CursoModeladoNegociosBPMN.htm>.
30. **Gil, Fran.** *Experto en Drupal 7 Nivel Inicial*. 2012.
31. **Hispano, Drupal.** Comunidad de usuarios de Drupal. . [En línea] 11 de Abril de 2005 . [Citado el: 11 de noviembre de 2015.] <http://drupal.org.es/drupa..>
32. **Riquelme, S. L. and Morales, M. C.** *Sistema Gestor de Base de Datos Relacionales*. 2010. 14.
33. **GUERRERO, R. M.** Sobre PostgreSQL. [En línea] 2009. [Citado el: 26 de enero de 2015.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
34. **Geschwinde, E. y Schöenig, HJ.** *Postgresql: Developer's Handbook*. . 2002.
35. **POSTGRESQL.** pgAdmin Introduction. [En línea] 2012. [Citado el: 26 de enero de 2016.] <http://www.pgadmin.org/> 44.
36. **ArPUG.** Grupo de usuarios PosgreSQL de Argentina. [En línea] 2008. [Citado el: 21 de Noviembre de 2015.] [http://www.arpug.com.ar/trac. .](http://www.arpug.com.ar/trac.)
37. **Welling, Luke y Thomson, Laura.** *Desarrollo Web con PHP y MySQL*. España : s.n., 2005.
38. **Sánchez Maza, Miguel Ángel.** *Javascript. Innovacion and cualificacion*. 2001.
39. **Smith, K.** *Ajax-style web development*. 2006.
40. **Converse, Tim, Park, Joyce y Morgan, Clark.** *PHP5 and MySQL Bible*. Indiana: Wiley Publishing. 2004.
41. **NetBeans.** NetBeans. *NetBeans*. [En línea] 2015. [Citado el: 15 de 1 de 2015.] <https://www.netbeans.org/features/index.html>.
42. **Collins-Sussman, Ben, Fitzpatrick, Brian W. y Pilato, C. Michael.** Control de versiones con Subversion. [En línea] 2004. [Citado el: 25 de noviembre de 2015.] <http://svnbook.red-bean.com/nightly/es/svn-ch-1-sect-1.html>.

43. **Samespinosa**. Modelo de Negocio. [En línea] Business & Mgmt. S.l., 2009. [Citado el: 31 de 03 de 2016.] <http://www.slideshare.net/samespinosa/mapa-de-procesos.1053479>.
44. **Martínez, Prof. Ivette Carolina**. CLASE 6: Modelo Conceptual/ Modelo de Dominio . [En línea] 2003. [Citado el: 01 de 04 de 2016.] [http://ldc.usb.ve/~martinez/cursos/ci3715/clase6\\_AJ2010.pdf](http://ldc.usb.ve/~martinez/cursos/ci3715/clase6_AJ2010.pdf).
45. **Pressman, Roger S**. *Software engineering: a practitioner's approach. 7th ed.* 2010.
46. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James**. *El Proceso Unificado de Desarrollo de Software*. Madrid : Addison Wesley, 2000.
47. **Kruchten**. *The Rational Unified Process: An Introduction*. 2000.
48. **Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P, y Stal, M, Pattern**. *Oriented Software Architecture. A System of Patterns*. Inglaterra : John Wiley & Sons, 1996.
49. **Cardoso, E, Camacho, F, y Nuñez, G**. *ARQUITECTURAS DE SOFTWARE. GUÍA DE ESTUDIO*. 2004.
50. **Venete, Adriana**. *Introducción a los Patrones de Arquitectura*. 2011.
51. **Microsoft**. [En línea] 2013. [Citado el: 01 de 04 de 2016.] <http://msdn.microsoft.com/eses/library/bb972240.aspx> .
52. **Gamma**. *Design patterns*. s.l. : AddisonWesley, 2013.
53. **Larman, Craig**. *UML y patrones. 2da Edición*. México : Prentice Hall, 2003.
54. —. *UML y Patrones. Introducción al análisis y diseño orientado a objetos, 1ra Edición*.
55. **Pressman, Roger S**. *Ingeniería del Software: Un Enfoque Práctico. 6ta Edición*. 2005.
56. —. *Ingeniería del Software. Un enfoque práctico*. España : Mc Graw Hill, 5ta edition, 2001. 8448132149. 323 pp. .
57. **Herrero Palomo, Julián**. *Administración, gestión y comercialización en la pequeña empresa*. s.l. : Editorial Paraninfo, 2001.
58. **Domínguez, Rubio P**. *Introducción a la gestión Empresarial*. s.l. : Edición electrónica, 2006.