

Universidad de las Ciencias Informáticas

Facultad 3



**Análisis y diseño de los procesos Administración de Compra y Venta
para el Sistema de Administración de Relaciones con el Cliente.**

Autor:

Armando Machado Machado

Tutoras:

Ing. Leidy Ramos González

Ing. Olga Yarisbel Rojas Grass

La Habana, Julio 2012

“Año 54 de la Revolución”



“El mundo camina hacia la era electrónica... Todo indica que esta ciencia se constituirá en algo así como una medida del desarrollo; quien la domine será un país de vanguardia. Vamos a volcar nuestros esfuerzos en este sentido con audacia revolucionaria”

Ernesto Che Guevara

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Centro de Informatización de la Gestión de Entidades (CEIGE) de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los ____ días del mes de Julio del año 2012.

Armando Machado Machado

Firma del Autor

Ing. Leidy Ramos González

Firma del Tutor

Ing. Olga Yarisbel Rojas Grass

Firma del Tutor

AGRADECIMIENTOS

Le agradezco a todos los que me han apoyado para lograr el sueño de verme graduado.

Quiero agradecer en primer lugar a mis padres por quererme con tanta intensidad y darme todo lo que tuvieron a su alcance, a quienes no solo agradezco sus consejos y el apoyo constante, sino además, el privilegio de ser su hijo.

A mi hermana por todo el cariño y apoyo que siempre me ha dado.

A mi familia en general que todos se han preocupado por mí.

A todos mis amigos por compartir alegres y tristes momentos conmigo, les agradezco a todos por darme la oportunidad de ser su amigo.

A mis tutoras Olga y Leidy por su apoyo y dedicación, y en especial a Olga por mostrarme un rayo de luz cuando la oscuridad casi se apodera de mí.

Al tribunal por sus señalamientos oportunos.

A todos ustedes muchas gracias.

DEDICATORIA

A mis padres que siempre confiaron en mí y me brindaron su apoyo para que viera realizados mis sueños, quiero regalarles este momento y honrarlos por tanto amor y dedicación. Los quiero mucho.

Mantener un cliente satisfecho es lo primordial para cualquier empresa, pues estos son los que adquieren, consumen y demandan los productos que son fabricados por ellas. El surgimiento de nuevas formas y técnicas para el trabajo posibilitan obtener software que ayuden a viabilizar los procesos empresariales donde juega un papel fundamental los Sistemas de Administración de Relaciones con el Cliente o Customer Relationship Management (CRM). En la Universidad de las Ciencias Informáticas (UCI) no existe un software de producción nacional de este tipo, el presente trabajo está enfocado en realizar el análisis y diseño de los procesos de Compra y Venta que formará parte del CRM a realizar por el Centro de Informatización de la Gestión de Entidades, el cual se desarrollará usando como base la suite OpenERP; agregándole nuevas funcionalidades. Este documento consta de tres capítulos: el primero aborda la fundamentación teórica; en el segundo se realiza la descripción del sistema y se validan los requisitos y en el tercero se elabora el diseño y se valida.

Palabras claves: administración de relaciones con el cliente, compra, venta.

Introducción	1
Capítulo 1: Fundamentación teórica de la investigación	5
1.1 Introducción.....	5
1.2 Marco conceptual.....	5
1.3 Análisis de los principales sistemas	7
1.4 Valoración crítica de los sistemas estudiados	12
1.5 Metodologías, lenguajes y herramientas.....	13
1.5.1 Modelo de desarrollo orientado a componentes.....	13
1.5.2 Notación de modelado de procesos de negocio BPMN	14
1.5.3 Lenguaje Unificado de Modelado UML.....	14
1.5.4 Herramienta CASE.....	15
1.5.5 Sistema gestor de base de datos	16
1.6 Ingeniería de requisitos (IR)	16
1.6.1 Actividades de la Ingeniería de Requisitos.....	17
1.6.2 Técnicas para levantamiento de requisitos	18
1.7 Patrones de diseño.....	19
1.8 Lenguajes de programación	21
1.9 Conclusiones.....	22
Capítulo 2: Descripción del sistema.....	23
2.1 Introducción.....	23
2.2 Modelo de Negocio.....	23
2.3 Mapa de procesos y descripción del negocio.....	23
2.4 Representación del Modelo Conceptual.....	26

ÍNDICE

2.5 Especificación de requisitos	27
2.5.1 Técnicas utilizadas para la recopilación de requisitos	27
2.5.2 Requisitos funcionales	28
2.5.3 Requisitos no funcionales	32
2.6 Especificaciones de requisitos	37
2.6 Prototipos de interfaz de usuario	44
2.7 Validación de requisitos	46
2.8 Validación de los resultados por métricas	46
2.9 Administración de requisitos	47
2.10 Conclusiones	49
Capítulo 3: Diseño de la solución	50
3.1 Introducción	50
3.2 Diagrama de componentes	50
3.3 Patrones de diseño	51
3.4 Patrón de arquitectura Modelo Vista Controlador	52
3.5 Diagrama de clases del diseño	53
3.6 Modelado de datos	55
3.7 Descripción de las tablas	55
3.8 Métricas para validar el diseño	60
3.9 Conclusiones	66
Conclusiones generales	67
Recomendaciones	67
Bibliografía	68
Anexos	¡Error! Marcador no definido.

Introducción

En los últimos años, las empresas han comprendido que el cliente es el principal motor de la empresa, pues son ellos quienes adquieren, consumen y demandan los productos que se fabrican. Para lograr que las relaciones con los clientes sean más duraderas surge el concepto de marketing relacional.

Los grandes adelantos tecnológicos han puesto en las manos de las empresas herramientas de gestión empresarial, la unión de estas herramientas y la teoría del marketing relacional dan lugar al surgimiento de la Administración de la Relación con el Cliente (CRM¹). Los recientes cambios tecnológicos, la innovación y las exigencias de un mercado globalizado han dado lugar a la creación y el mejoramiento de nuevos productos y servicios; por lo que el desarrollo de una ventaja competitiva resulta vital para las organizaciones cubanas.

La informática en Cuba se emplea bastante extendida en la automatización de las funciones más convencionales del control empresarial, como son: la contabilidad, las nóminas, los activos fijos, los inventarios, las compras, las ventas entre otras. Particularmente, en organizaciones complejas la posibilidad de captar datos primarios, almacenarlos correctamente, procesarlos y convertirlos en información para los diferentes niveles de dirección resulta de un valor inestimable.

El país puede y debe usar con eficiencia las tecnologías de la información y las comunicaciones para mejorar los procesos de gestión, tanto de las empresas como del gobierno, a partir de la disponibilidad de recursos financieros y materiales que exista. Las mayores fortalezas radican en los cientos de especialistas graduados en los últimos años, que están en condiciones de acompañar y conducir estos esfuerzos, y en la voluntad de impulsar la informatización, como soporte al desarrollo integral de la sociedad.

Cuba, bajo el bloqueo económico al cual ha estado sometida hace más de 50 años, se ha visto en la necesidad de reorganizar su modelo económico cubano en pos de mejorar procesos sustantivos, de forma tal que se logre un sistema empresarial constituido por empresas eficientes, organizadas y eficaces. El Sexto Congreso del Partido Comunista de Cuba tuvo entre sus principales objetivos evaluar y aprobar el proyecto final de los lineamientos de la Política Económica y Social del Partido y la Revolución, para actualizar dicho modelo económico cubano. Muchos son los lineamientos que argumentan los procesos de compras y

¹ Siglas en inglés de Customer Relationship Management

ventas en las empresas con el propósito de elevar el rendimiento empresarial y enfrentar los problemas económicos internos y externos que atraviesan las instituciones del país.

Muchas instituciones, incluyendo la Universidad de las Ciencias Informáticas (UCI) y su empresa comercializadora de software Albet SA, no cuentan con la información adecuada de cada uno de sus clientes como: sus características, sus tendencias, su fidelidad; y en muchos casos no se tiene una cultura de orientación hacia ellos. El personal de la dirección comercial invierte demasiado tiempo en tareas administrativas elaborando informes de visitas, perfiles de clientes, seguimiento, haciendo reuniones internas y otras actividades que implican un desgaste humano. Además no hay una comunicación entre mercadotecnia, ventas y atención al cliente, por lo que las oportunidades generadas por unos no son aprovechadas por otros.

En varias entrevistas realizadas a especialistas comerciales de las empresas Albet SA y Correos de Cuba se observó que se les daba el mismo tratamiento a clientes actuales que a potenciales cuando obviamente, las circunstancias de ambos son totalmente distintas y también debería ser distinta la manera de comunicarse con ellos. Además, las ventas y las compras no son realizadas por las mismas áreas, lo que trae consigo demora en la atención de las solicitudes y el incumplimiento de contratos, propiciando así la insatisfacción de los clientes. La información referente a los procesos de Compras y Ventas no se encuentra estandarizada por lo que los subprocesos y actividades que se realizan en estos procesos no están lo suficientemente descritos y documentados.

La situación expuesta desencadena el siguiente **problema a resolver**: la información que provee el cliente no está en un lenguaje técnico comprensible para los desarrolladores que facilite el desarrollo de los procesos Compra y Venta para el Sistema de Administración de Relaciones con el Cliente.

Para ello la investigación centra su **objeto de estudio** en los procesos de Administración de Relaciones con el Cliente y el **campo de acción** en los procesos de Compra y Venta.

El **objetivo general** perseguido es realizar el análisis y diseño de los procesos Compra y Venta para el Sistema de Administración de Relaciones con el Cliente.

Para cumplir con este objetivo se plantean una serie de **Objetivos Específicos**:

- Elaborar el marco teórico de la investigación.

- Realizar el análisis de los procesos de Compra y Venta para el Sistema de Administración de Relaciones con el Cliente.
- Elaborar el diseño de los procesos de Compra y Venta para el Sistema de Administración de Relaciones con el Cliente.
- Validar la solución propuesta.

Para dar cumplimiento a los objetivos trazados es necesario cumplir las siguientes **tareas de investigación:**

- Análisis del estado del arte de las herramientas CRM, específicamente cómo realizan la administración de compras y ventas.
- Descripción de los procesos de Compra y Venta.
- Definición de los requisitos funcionales y no funcionales de los procesos de Compra y Venta.
- Realización de la trazabilidad procesos-requisitos.
- Análisis de los patrones de diseño.
- Elaboración del diseño para los procesos de Compra y Venta del sistema CRM.
- Validación del diseño propuesto.

Capítulo 1. Fundamentación teórica de la investigación: constituye la fundamentación teórica del presente trabajo. Su lectura le ofrecerá información referente a los principales conceptos tratados. Lo pondrá al tanto del estado del arte del tema y por último podrá conocer la herramienta, la metodología y los lenguajes de modelado a utilizar para dar solución al problema en cuestión.

Capítulo 2. Descripción del sistema: se ofrece el mapa de procesos de negocio, el modelo conceptual, la especificación de los requerimientos de software que están presente para el modelado del sistema y la descripción de los reportes que pueden ser generados a partir del sistema. Finalmente se realiza una breve explicación de las técnicas y métricas utilizadas en la validación de los requisitos.

Capítulo 3. Diseño del sistema: se inicia con una breve conceptualización del diseño y de los patrones a utilizar, mostrando luego la realización de los diagramas de clases con sus

INTRODUCCIÓN

descripciones y el diagrama de componentes. Además se ofrecen las métricas utilizadas para la validación del diseño así como sus resultados.

Capítulo 1: Fundamentación teórica de la investigación

1.1 Introducción

En el presente capítulo se abordan los principales conceptos a tratar relacionados con los Sistemas de Administración de Relaciones con los Clientes y principalmente los referentes a los procesos de Compra y Venta. Se realizará un estudio del estado del arte de los distintos sistemas de Administración de Relaciones con los Clientes y tecnologías relacionados con el tema así como la definición de las herramientas, metodología y lenguajes que se utilizarán en la propuesta del sistema.

1.2 Marco conceptual

Ventas:

La American Marketing Association, define la venta como "el proceso personal o impersonal por el que el vendedor comprueba, activa y satisface las necesidades del comprador para el mutuo y continuo beneficio de ambos (del vendedor y el comprador)". (1)

El concepto de venta supone que es preciso estimular a los consumidores para que compren. Para ello, las empresas que ponen en práctica este concepto, utilizan todo un arsenal de herramientas de venta y promoción para estimular más compras. (2)

En general, ventas para una empresa u organización es vender lo que produce. Para ello, dirige sus esfuerzos hacia todas las actividades que le permitan estimular a sus prospectos para que tomen una decisión favorable como inscribirse o comprar.

Compras:

Comprar supone el proceso de localización y selección de proveedores, adquisición de productos, luego de negociaciones sobre el precio y condiciones de pago, así como el acompañamiento de dicho proceso para garantizar su cumplimiento de las condiciones pactadas; y en términos de mercadotecnia, comprar es adquirir por un precio en dinero algún bien, derecho o mercancía. (3)

Comprar es la acción de adquirir un producto ofrecido por un vendedor, a través de un contrato de compra-venta, a cambio de un precio en dinero, cierto y no simulado. (4)

Comprar es la acción de adquirir un producto ofrecido por un vendedor, a través de un contrato o a cambio de un precio en dinero.

Administración de Relaciones con el Cliente (CRM)

El CRM consiste en una estrategia de la organización en la cual centra sus esfuerzos en el conocimiento de sus clientes, detectando sus necesidades, aumentando su grado de satisfacción, incrementando su fidelidad a la empresa e incrementando la rentabilidad o beneficios del cliente a la empresa, mediante el análisis de las informaciones extraídas por los clientes desde los diferentes canales o medios de comunicación. (5)

Un CRM es una herramienta de negocios que a partir de soluciones tecnológicas, se pueden mejorar las relaciones con los clientes. Para esto el sistema brinda un grupo de datos acerca de los clientes que son usados para realizar un estudio previo del mercado para conocer los gustos de los clientes, adaptando los productos a sus necesidades y lograr que queden totalmente satisfechos.

ERP (Enterprise Resource Planning)

El sistema de planificación de recursos empresariales, más conocido como ERP (Enterprise Resource Planning) es un sistema estructurado que busca satisfacer la demanda de soluciones de gestión empresarial. Se basa en el concepto de una solución completa que permita a las empresas unificar las diferentes áreas de productividad de la misma. (6)

Los ERP son sistemas que le permiten a las empresas gestionar toda la información en tiempo real, permitiendo una mejor planificación, análisis y gestión de sus recursos. En ocasiones estos son complejos y difíciles de implantar debido a que necesitan un desarrollo personalizado para cada empresa partiendo de la parametrización inicial de la aplicación que es común para todas.

Clientes

En toda estrategia empresarial, el cliente es una figura que siempre está presente, es lógico, es quien demanda los productos y servicios que las empresas ofrecen, por lo que consiguen consolidarse en el mercado para obtener rendimientos e ingresos para posicionarse y sobrevivir. (7)

Los clientes son aquellos que le hacen compras a la empresa de forma periódica o que lo hicieron en una fecha reciente.

Cliente Potencial

Es aquella (persona, empresa u organización) que no le realiza compras a la empresa en la actualidad pero que son visualizados como posibles clientes en el futuro porque tienen la disposición necesaria, el poder de compra y la autoridad para comprar. Este tipo de clientes es el que podría dar lugar a un determinado volumen de ventas en el futuro (a corto, mediano o largo plazo) y por tanto, se les puede considerar como la fuente de ingresos futuros. (8)

Cliente potencial es una persona, empresa u organización que aún no se relaciona con la empresa pero pueden ser un posible cliente en el futuro.

1.3 Análisis de los principales sistemas

Con el uso de los CRM las empresas mejoran las relaciones con sus clientes, obtienen más información acerca de los clientes aumentado su grado de fidelidad con la empresa. A continuación se relacionan algunos de los CRM más utilizados en el mundo empresarial.

Microsoft Dynamics

Microsoft Dynamics CRM permite definir una estrategia de negocio centrada en anticipar, conocer, satisfacer las necesidades y los deseos de sus clientes, incrementando la efectividad de los empleados de ventas y servicios. (9)

Microsoft Dynamics CRM permite que pequeñas y medianas empresas, y áreas específicas de grandes organizaciones, incrementen el éxito de sus ventas, entreguen un mejor servicio al cliente y tomen mejores decisiones, en base al manejo de información de valor. (10)

Acciones de ventas:

- Gestiona tipo de órdenes
- Gestiona tipo de operación
- Genera reportes personalizados para cada etapa
- Gestiona las liquidaciones de comisiones
- Gestiona los pedidos de venta
- Gestiona los descuentos de ventas
- Gestiona las solicitudes de ventas

Acciones de compras:

- Gestiona presupuestos
- Define la cantidad de autorizaciones para generar órdenes de compras
- Gestión de pedidos de compras
- Gestiona las solicitudes de compras

Funcionalidades:

- Administrar sus datos de manera eficaz
- Automatizar los flujos de trabajo
- Obtener conocimientos clave para la toma de decisiones
- Simplificar el seguimiento de los clientes potenciales

SugarCRM

SugarCRM es distinto a las demás soluciones que existían en el mercado antes de su aparición. La gran diferencia es que se trata de la primera aplicación de "código abierto" que consiguió posicionarse como líder de ese segmento. Esto implica que en principio cualquiera puede descargar la versión open source y empezar a utilizarla, por lo que los costos por licencia de software no existen. (11)

SugarCRM está construido utilizando principalmente PHP y MySQL. SugarCRM puede funcionar en un ambiente LAMP (Linux Apache MySQL y Php), WAMP (Windows Apache MySQL Php) e inclusive en uno MAMP (MacOS Apache MySQL Php) lo cual le permite correr en múltiples plataformas sin ningún problema. Estas características han hecho que muchísimas empresas lo hayan adoptado en todo el planeta. (12)

Acciones de ventas:

- Gestión de oportunidades
- Gestión de clientes
- Gestión de clientes potenciales
- Gestión de pronósticos de ventas
- Gestión de contratos

OpenERP (CRM)

OpenERP es un sistema de gestión empresarial (ERP) y de relación con el cliente (CRM) de código abierto. OpenERP cubre las necesidades de las áreas de contabilidad, ventas, compras, y almacén e inventario, entre otras. También soporta múltiples monedas, múltiples compañías y múltiples contabilidades; además incorpora funcionalidades de gestión de documentos para agilizar la colaboración entre departamentos y equipos en la empresa; y permite trabajar remotamente mediante una interfaz web desde una computadora conectada a Internet. (13)

Características:

- OpenERP es un sistema ERP y CRM, utilizado para la gestión integrada de los recursos de la empresa.
- Se integra con distintos software de oficina. Dispone de funcionalidad para la generación de impresos vía PDF, HTML, y permite exportar datos a otros programas como OpenOffice o MS Office (Excel, Word).
- La arquitectura del sistema es cliente – servidor, lo que permite que todos los usuarios trabajen sobre el mismo repositorio de datos. Esto tiene la ventaja de que toda la información está disponible y sincronizada en todo momento además de que descarga la mayor parte del trabajo de procesamiento de datos de las máquinas cliente (donde trabajan efectivamente los usuarios). Dispone de interfaces XML-RPC y SOAP.
- Dentro de la construcción misma del software se hace un uso intensivo de flujos de trabajo (modelo workflow) que se pueden integrar con sus distintos módulos. (14)

Acciones de ventas:

- Gestionar iniciativa de venta
- Gestión de pedido de venta
- Gestión de cliente
- Convertir iniciativa en oportunidad

Acciones de compras:

- Gestión de solicitud de compra
- Gestión de presupuesto
- Gestión de pedido de compra

VtigerCRM

VtigerCRM es una empresa de software CRM de código abierto principalmente para pequeñas y medianas empresas. Vtiger CRM está construido sobre los probados, rápidos, fiables LAMP / WAMP (Linux / Windows, Apache, MySQL y PHP) y tecnologías de otros proyectos de código abierto. (15)

Con VTiger se puede administrar completamente una empresa, a través de los diversos módulos que incluye, con los que se puede manejar las actividades de inventario, incluyendo fuerza de ventas, mercadotecnia, clientes y contactos, productos, presupuestos, facturación y pedidos, soporte y servicio al cliente. Actualmente existen muchas herramientas y plugins para VTigerCRM que asistirán en mejorar la productividad de una organización. No existe límite de usuarios ni tiene costo por licencia y se pueden integrar nuevos módulos de acuerdo a los requerimientos de una empresa. (16)

Acciones de ventas:

- Gestión de contactos
- Gestión de oportunidad de venta
- Gestión de pedido de venta
- Gestión de facturas
- Gestión de tarifas

Ventajas

- Es un programa muy completo, flexible, adaptable
- Es un programa de código abierto
- Excelente herramienta de gestión para empresas que está en castellano

Desventajas

- Es un programa muy cargado, aunque es intuitivo
- No es fácil de manejar porque tiene demasiadas variables
- Requiere cambiar el chip (para mejor) pero te obliga a enfocarte a lo esencial y romper con el enfoque a facturas tradicional

Sap CRM

El software Sap CRM ofrece, en su conjunto de aplicaciones buenas funcionalidad para marketing, ventas y servicios. Soporta procesos de negocios para tratar directamente con los clientes en múltiples canales de interacción, Sap CRM permite que las organizaciones mantengan el foco en estrategias de crecimiento centrado en el cliente y que se diferencien en el mercado por ofrecer una experiencia superior para el cliente.

Sap CRM es capaz de:

Gestionar procesos relacionados con los clientes: SAP CRM asegura la orquestación de todas las tareas relacionadas con los clientes, de un departamento a otro, incorporando, de forma transparente, actividades tales como fulfillment, distribución, facturación y cuentas por cobrar.

Suministrar a toda la organización información de clientes: la solución SAP CRM reúne todas las fuentes relevantes de datos de clientes, distribuidas por todas las empresas, contribuyendo para lograr un mejor proceso de toma de decisiones.

Ofrecer beneficios inmediatos: solo SAP CRM permite que las empresas resuelvan, en primer lugar, las prioridades estratégicas y cumplan con los objetivos más rápidamente. La solución puede expandirse gradualmente y cada etapa traerá, de una manera tangible, el correspondiente retorno sobre la inversión. (17)

Acciones de ventas:

- Gestión de cuentas y contactos
- Gestión de actividades
- Gestión de oportunidades
- Gestión de territorios
- Gestión de pedido de venta
- Gestión de contratos

1.4 Valoración crítica de los sistemas estudiados

Tabla 1. Comparación entre los CRM

CRM	Plataformas	Costo	Funcionalidades	Soporte
Microsoft Dynamic	Windows	Alto	Ventas: gestiona pedidos de venta, descuentos, solicitud y tipo de operación. Compra: gestiona presupuestos, pedidos de compras y solicitudes de compras.	Administra asuntos de soporte desde el contacto inicial hasta la resolución final.
SugarCRM	Windows, Linux, Mac OS X	Bajo	Ventas: gestiona las oportunidades, clientes, clientes potenciales, pronósticos de ventas y contratos.	No hay soporte de una empresa quien certifique o dé seguridad del producto.
OpenERP	Windows, Linux	Bajo	Ventas: gestiona iniciativa de venta, pedido de venta y cliente. Compra: gestiona solicitud de compra, de presupuesto y pedido de compra.	No hay soporte de una empresa quien certifique o dé seguridad del producto.
VTiger	Linux	Bajo	Ventas: gestiona las oportunidades de ventas, contratos, pedido de venta, facturas y tarifas.	No hay soporte de una empresa quien certifique o dé seguridad del producto.
SAP	Windows	Alto	Ventas: gestiona oportunidades, contactos, actividades, territorios, pedido de venta y contratos	Administra asuntos de soporte desde el contacto inicial hasta la resolución final.

A partir del estudio realizado a los sistemas mencionados anteriormente, se observó que a pesar de brindar ventajas en el área Compra/Venta, no cumplen con todas las necesidades de las entidades del país. Muchos de estos sistemas fueron desarrollados bajo herramientas y software propietarios; ejemplo de esto SAP y Microsoft Dynamic, sistemas que a pesar de brindar soporte desde el contacto inicial hasta la resolución final y poseer grandes funcionalidades para el manejo de las relaciones con los clientes, tienen un costo inicial muy alto por lo que el país utilizar alguno de ellos incurriría en grandes gastos de licencia y mantenimiento. Los demás sistemas pertenecen a la familia de los OpenSource, que a pesar de tener un costo inicial bajo, a diferencia de los propietarios, no tienen soporte de una empresa que certifique o garantice la seguridad del producto. De todos los sistemas estudiados el que más se adecua es el OpenERP porque brinda flexibilidad y simplicidad a la hora de realizar modificaciones y adaptaciones, ajustándolo a las necesidades, otra de las posibilidades que brinda es la integración con herramientas de negocio, utiliza un flujo de trabajo flexible y dinámico, pudiéndosele agregar funciones y módulos e integrarlos a los ya existentes, además de soportar plataformas heterogéneas. La propuesta es desarrollar un sistema CRM tomando como base la suite de OpenERP, adicionándole características y funcionalidades que se necesitan para alcanzar el producto deseado.

1.5 Metodologías, lenguajes y herramientas

1.5.1 Modelo de desarrollo orientado a componentes

El modelo de desarrollo orientado a componentes, se utilizará en la adaptación y construcción de las funcionalidades del Sistema de Administración de Relaciones con el Cliente, definido por el Centro de Informatización de la Gestión de Entidades (CEIGE).

Características fundamentales:

- La ingeniería de requisitos es más clara que en las demás metodologías de desarrollo.
- Se modela el negocio por procesos, no por casos de usos.
- Es orientada a componentes, posibilitando la independencia de funciones del sistema a la hora de mantener o modificar el sistema funcional.
- Utiliza como lenguajes de modelado BPMN para el negocio y UML para el diseño.

- Se definen claramente los roles que intervienen y las responsabilidades de cada uno de ellos, las actividades de desarrollo que se deben llevar a cabo, bien descritas y que roles las realizan, así como los artefactos y los roles que lo generan.

Se mantiene el objetivo fundamental de todo proceso en el que se define quién está haciendo qué, cuándo y cómo para alcanzar un determinado objetivo. Un Proceso de Desarrollo de Software es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el proyecto. (18)

1.5.2 Notación de modelado de procesos de negocio BPMN

BPMN (Business Process Management Notation): se decide utilizar BPMN como notación para el modelado del negocio, por ser fácil, legible y entendible. Dicha herramienta está diseñada para cubrir varios tipos de modelado, permite la creación tanto de segmentos de procesos, como procesos de negocio de comienzo a fin en diferentes niveles de representatividad. BPMN es gráficamente más rico, con menos símbolos fundamentales, pero con más variaciones de éstos, lo que facilita su comprensión por parte de personas no expertas.

Provee una notación que es fácilmente entendida por todos los usuarios, desde el analista de negocio, el desarrollador técnico y hasta la propia gente del negocio.

Asegura que los lenguajes para la ejecución de los procesos de negocio puedan ser visualizados con una notación común. (19)

1.5.3 Lenguaje Unificado de Modelado UML

UML (Unified Modeling Language): es un lenguaje que permite visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. (20) UML es un conjunto de herramientas, que permite modelar, analizar y diseñar sistemas orientados a objetos. (21)

Se ha convertido en el estándar de facto de la industria. UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. Su utilización es independiente del lenguaje de programación y de las características de los proyectos, ya que UML ha sido

diseñado para modelar cualquier tipo de proyectos, tanto informáticos como de arquitectura, o de cualquier otra rama. UML cuenta con varios tipos de diagramas que son la representación gráfica de un conjunto de elementos y sus relaciones que visualizan el sistema desde diferentes perspectivas. (22)

1.5.4 Herramienta CASE

La ingeniería de software asistida por computadora (CASE: Computer Aided Software Engineering: Software Asistida por Computación) ayuda a los ingenieros de software en todas las actividades asociadas a los procesos de software. Las herramientas CASE automatizan las actividades de gestión de proyectos, gestionan todos los productos de los trabajos elaborados a través del proceso y ayudan a los ingenieros en el trabajo de análisis, diseño y codificación. La importancia de su uso radica en permitir reducir la cantidad de esfuerzo que se requiere para producir un producto o para alcanzar un hito en el proceso de desarrollo. Además, contribuyen a la calidad del software ya que proporcionan nuevas formas de observar la información de la ingeniería del software. (23)

Visual Paradigm para UML es una herramienta CASE multiplataforma de modelado visual UML, muy potente y fácil de utilizar. Soporta el ciclo de vida completo del desarrollo de software, ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. (24)

Herramienta OSRMT

Es una herramienta de Software Libre pensada para asistir en todo el Ciclo de Vida del Desarrollo del Software. Permite la descripción avanzada de diversos tipos de requisitos y garantiza la trazabilidad entre todos los documentos relacionados con la ingeniería de requisitos (funcionalidades, requisitos, casos de uso, casos de prueba). (25)

Funcionalidades:

- Gestión de la configuración: versionado y registro de los cambios realizados en los diferentes elementos
- Gestión de usuarios y permisos

- Herramientas de migración para los diversos cambios de versiones
- Múltiples idiomas (importación y exportación para dar soporte a diversos idiomas)
- Importar y exportar información en XML y mediante línea de comandos
- Exportar información en HTML mediante línea de comandos (25)

1.5.5 Sistema gestor de base de datos

PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre. Es usado para manejar grandes cantidades de información y está basado en el modelo relacional, aunque incorpora conceptos del modelado orientado a objeto. Se pueden definir consultas anidadas, vistas, crear funciones por el usuario, no sólo en el lenguaje natural SQL, sino en varios más, entre ellos C, lenguaje nativo PostgreSQL, Perl, PHP y Java. Es multiplataforma, soporta múltiples transacciones, integridad de datos, presenta una estabilidad muy alta, gran seguridad de los datos, soporta la réplica y procedimientos almacenados. Propone un tamaño ilimitado para las base de datos, lo que da la medida de un gestor de base de datos robusto, y con grandes funcionalidades. (26)

1.6 Ingeniería de requisitos (IR)

La Ingeniería de requisitos tiene lugar durante todo el ciclo de vida del software, fundamentalmente en las primeras etapas cuando es necesario encontrar y comunicar las necesidades de clientes y usuarios.

Mejora la capacidad de predecir cronogramas de proyecto, así como sus resultados. Realizar un proceso bien detallado de IR disminuye los costos y retrasos del proyecto, influyendo esto en que mejore la calidad del software, que debe cumplir con un conjunto de requisitos como son: la funcionalidad, facilidad de uso, confiabilidad y desempeño. Además de proporcionar un consenso entre clientes y desarrolladores, que es vital para que el proyecto sea exitoso.

La obtención correcta de los requisitos puede llegar a describir con claridad el comportamiento de un sistema. De tal manera que, basarse en la extracción de requisitos y sobre todo en que sean correctos, proporciona minimizar los problemas relacionados al desarrollo de sistemas, logrando la reducción de tiempo en la construcción, así como la reducción de errores. (27)

1.6.1 Actividades de la Ingeniería de Requisitos

El proceso de Ingeniería de Requisitos puede ser descrito a través de 4 actividades fundamentales, las cuales se describen a continuación:

Elicitación de requisitos

Este proceso hace referencia a la captura de requisitos, mediante esta actividad el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. El proceso de elicitación puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él. Por la complejidad que todo esto puede implicar, la IR ha trabajado desde hace años en desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y precisa. Entre las técnicas existentes se encuentran: la Entrevista, el Diseño Común de Aplicaciones o Desarrollo Conjunto de Aplicación (JAD), la Tormenta de Ideas y otras como los Mapas Conceptuales. Estas técnicas mencionadas anteriormente fueron las utilizadas para esta etapa de elicitación en el proceso de desarrollo con el objetivo de captar las necesidades de la parte interesada. (28)

Análisis de requisitos

Sobre la base de la extracción realizada previamente, comienza esta fase. La misma se enfoca en descubrir problemas con los requerimientos del sistema identificados hasta el momento. Usualmente se hace un análisis luego de haber producido un bosquejo inicial del documento de requerimientos; en esta etapa se leen los requerimientos, se conceptúan, se investigan, se intercambian ideas con el resto del equipo, se resaltan los problemas, se buscan alternativas y soluciones. Se trasladan los requisitos de usuario a requisitos del software (implementables). Se precisan los límites del sistema y la interacción con su entorno. (29)

Especificación de requisitos

La Especificación de Requisitos (ER) de un sistema de software describe su función, características, y las restricciones que gobiernan su desarrollo. Describe la información que entra y sale del sistema. La flexibilidad es un punto a tener en cuenta cuando se va a realizar una especificación.

Una especificación puede ser un documento escrito, un modelo gráfico, un modelo matemático formal, una colección de escenarios de uso, un prototipo o combinación de lo antes citado. Se sugiere que se realice una plantilla estándar, generalmente se genera el documento con el mismo nombre que contiene una descripción completa de las necesidades y funcionalidades del sistema que será desarrollado; se describe el alcance del sistema y la forma en cómo hará sus funciones. (29)

Validación de requisitos

La validación tiene como objetivo examinar las especificaciones para verificar que los requisitos del sistema han sido establecidos sin ambigüedad, inconsistencias, omisiones, que los errores detectados han sido corregidos y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso del proyecto y de producto. (29)

Existen diferentes técnicas para la validación de requisitos las cuales son:

Prototipos: Algunas propuestas se basan en obtener de la definición de requisitos prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea de la estructura de la interfaz del sistema con el usuario. Esta técnica tiene el problema de que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final. (30)

Matrices de Trazabilidad: Esta técnica consiste en marcar los objetivos del sistema y chequearlos contra los requisitos del mismo. Es necesario ir viendo los objetivos que cubre cada requisito para de esta forma detectar inconsistencias y objetivos no cubiertos. (30)

Revisión: esta técnica consiste en la lectura y corrección de la completa documentación o modelado de la definición de requisitos. Con ello solamente se puede validar la correcta interpretación de la información transmitida. (30)

Gestión de requisitos

Se encarga de darle seguimiento a los requisitos a lo largo de las etapas anteriores. El objetivo es realizar un conjunto de actividades que permiten identificar, seguir y controlar los cambios sufridos en cualquier momento del ciclo de vida del sistema. (31)

1.6.2 Técnicas para levantamiento de requisitos

La captura de requisitos es la actividad donde se extraen las necesidades del sistema. Durante

la investigación realizada se encontraron una serie de técnicas usadas para realizar esta actividad, de ellas se seleccionaron cual o cuales se aplicarían.

Entrevistas y Cuestionarios

Las entrevistas y cuestionarios se emplean para reunir información proveniente de personas o de grupos. Durante la entrevista, el analista conversa con el encuestado; el cuestionario consiste en una serie de preguntas relacionadas con varios aspectos de un sistema. Por lo común, los encuestados son usuarios de los sistemas existentes o usuarios en potencia del sistema propuesto. En algunos casos, son gerentes o empleados que proporcionan datos para el sistema propuesto o que serán afectados por él. El éxito de esta técnica, depende de la habilidad del entrevistador y de su preparación para la misma.

Sistemas existentes

Esta técnica consiste en analizar distintos sistemas ya desarrollados que están relacionados con el sistema a ser construido. Por un lado, se pueden analizar las interfaces de usuario, observando el tipo de información que se maneja y cómo es manejada, por otro lado también es útil analizar las distintas salidas que los sistemas producen (listados, consultas, etc.), porque siempre pueden surgir nuevas ideas sobre la base de estas.

Tormentas de ideas

Es una técnica grupal y consiste en la simple acumulación de ideas sin detenerse en el análisis del valor de las mismas, utilizada para el entendimiento común de los requisitos capturados y determinación de posibles cambios o errores en dicha captura. (32)

1.7 Patrones de diseño

Los patrones de diseño surgen ante la necesidad de la reutilización de diseños y de soluciones a problemas frecuentes encontrados por los ingenieros. Con la reutilización se consigue la reducción de tiempos y la disminución del esfuerzo de mantenimiento, esto trae consigo mayor eficiencia y consistencia en el diseño de la solución. Las características que presentan los patrones son las siguientes: son soluciones concretas, proponen soluciones a problemas concretos, no son teorías genéricas.

Se utilizan en situaciones frecuentes, ya que se basan en la experiencia acumulada al resolver

problemas reiterativos.

Favorecen la reutilización de código. Ayudan a construir software basado en la reutilización, a construir clases reutilizables. Los propios patrones se reutilizan cada vez que se vuelven a aplicar. (33)

Dentro de los patrones más conocidos se encuentran los Patrones de asignación de responsabilidad (GRASP).

Patrones GRASP:

Creador: con la solución de asignarle a una clase la responsabilidad de crear los objetos de la otra en los casos de: contener, agregar, registrar o utilizar. Brindando soporte de bajo acoplamiento, lo cual supone menos dependencias entre clases y posibilidades.

Bajo Acoplamiento: promueve la baja dependencia y una alta reutilización de las clases. Asigna responsabilidades manteniendo un bajo acoplamiento. Soporta el diseño de clases más independientes, que reducen el impacto de cambios que acrecientan la reutilización. Uno de los principales síntomas de un mal diseño y alto acoplamiento es una herencia muy profunda. Siempre hay que considerar las ventajas de la delegación respecto de la herencia.

Experto: la responsabilidad de realizar una labor es de la clase que tiene o puede tener los datos involucrados (atributos). Una clase, contiene toda la información necesaria para realizar la labor que tiene encomendada.

Controlador: asigna la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase.

Alta Cohesión: la alta cohesión al igual que el bajo acoplamiento logran un diseño fácil de reutilizar y adaptar, más legibilidad para los programadores y diseñadores en general; la extensibilidad y flexibilidad del diseño aumentan, ya que las clases usan solo lo que necesitan y en su mayoría son recursos propios, por otro lado, cada cual hace lo que le corresponde hacer según la parte del problema que está modelando. (34)

Patrón de arquitectura Modelo Vista Controlador:

El patrón de arquitectura conocido como Modelo-Vista-Controlador (MVC), separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario; es

decir separa en tres capas diferentes los datos de una aplicación, la interfaz de usuario, y la lógica de control:

Modelo: Esta capa administra el comportamiento y los datos del dominio de la aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador).

Vista: Esta capa maneja la visualización de la información, es decir que presenta el modelo en un formato adecuado para interactuar, que usualmente es la interfaz de usuario.

Controlador: Esta capa controla el flujo de datos entre la vista y el modelo; es decir que responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. (35)

1.8 Lenguajes de programación

Los lenguajes de programación estudiados para el diseño definido en la arquitectura son:

Python es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad. (36)

Características

- Lenguaje de alto nivel
- Implementado en C (Jython, IronPython, Pypy)
- Multiplataforma
- Multiparadigma: programación OO, estructurada o funcional (36)

Ventajas

- La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje, que ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.

- La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de 3 a 5 líneas de código menos que su equivalente en Java o C.
- La cantidad de plataformas en que se puede desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otros.
- Además, Python es gratuito, incluso para propósitos empresariales. (36)

XML

XML es un formato basado en texto, específicamente diseñado para almacenar y transmitir datos. Un documento XML se compone de elementos XML, cada uno de los cuales consta de una etiqueta de inicio, de una etiqueta de fin y de los datos comprendidos entre ambas etiquetas. XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades, de ahí que se le denomine metalenguaje. (37)

Ventajas

- Fácilmente procesable tanto por humanos como por software.
- Separa radicalmente la información o el contenido de su presentación o formato.
- Diseñado para ser utilizado en cualquier lenguaje o alfabeto.
- Su análisis sintáctico es fácil debido a las estrictas reglas que rigen la composición de un documento.
- Estructura Jerárquica.
- El número de marcas es ilimitado. (38)

1.9 Conclusiones

En este capítulo se analizaron varios Sistemas de Administración de Relaciones con el Cliente existentes en el mundo para seleccionar características comunes que puedan ser utilizadas en el sistema a desarrollar. Se tomara el OpenERP como sistema base para realizar la propuesta de solución, adicionándole características y funcionalidades que se necesitan para alcanzar el producto deseado. También se brindó información sobre del modelo de desarrollo, lenguajes de modelado y la herramienta CASE que se van a utilizar.

Capítulo 2: Descripción del sistema

2.1 Introducción

En este capítulo se describen los procesos de negocio, mediante el modelo conceptual se representan los conceptos y las relaciones de los procesos de Compra y Venta. Se definen los requisitos funcionales y los no funcionales, así como la descripción textual de los mismos. Serán plasmados los diferentes artefactos propuestos en el Modelo de desarrollo orientado a componentes.

2.2 Modelo de Negocio

El Modelado de Negocio se define como un proceso de representación de uno o más aspectos o elementos de una empresa, tales como su propósito, su estructura, su funcionalidad, su dinámica, su lógica de negocios y sus componentes. Es una actividad previa y complementaria al proceso de desarrollo de software que permite:

- Entender los problemas actuales en la organización o empresa, para identificar los aspectos a mejorar.
- Estudiar el impacto que pueden producir los cambios a nivel organizativo.
- Asegurar que los clientes, usuarios finales, desarrolladores y otros involucrados tengan una visión común de la organización.
- Obtener los requisitos del sistema de software.
- Entender como el sistema de software se ajusta a la organización. (39)

2.3 Mapa de procesos y descripción del negocio

Para una mayor comprensión del sistema que se desea desarrollar es necesario partir de la modelación del negocio, en este caso se modelará por procesos. El modelado de procesos del negocio consiste en describir la realidad de manera que esta pueda ser entendida y de ser necesario modificada con el fin de incorporarle mejoras. Un mapa de procesos muestra cómo se relacionan los procesos de negocio especificando las entradas y salidas de cada uno. Las entradas son los documentos que se necesitan para realizar el proceso, y las salidas la

constancia de lo que se hizo además de las condiciones en que quedó el proceso. A continuación se muestran los elementos que componen el mapa de procesos:

Tabla 2. Procesos de Negocio

Nº	Nombre del proceso	Breve descripción del proceso	del Proceso padre
1	Compra	Acción de adquirir u obtener algo a cambio de un precio establecido.	Compra/Venta
2	Venta	Es la acción y efecto de vender. Contrato en virtud del cual se transfiere a dominio ajeno una cosa propia por el precio pactado.	Compra/Venta

En la matriz de relaciones entre procesos de negocio de la Figura 1, se muestran las entradas y las salidas de cada uno de los procesos de la administración de las relaciones con el cliente y su relación con el proceso de logística, que es un proceso externo.

		Entradas (General)					
		Compra	Venta	Mercadotecnia	Servicios	Facturación	Inventario
S a l i d a s	Compra			Comportamiento de compras			Hoja de inventario
	Venta	Solicitud de Compra		Comportamiento de ventas	Contrato Perfil de cliente	Factura	Hoja de inventario
	Mercadotecnia	Cartera de proveedores	Oportunidades Perfil de cliente potencial Ficha comercial				
	Inventario	Catálogo de productos Hoja de inventario	Catálogo de productos Hoja de inventario				

Figura 1. Matriz de relaciones entre procesos de negocio

Proceso de Compra

Se realiza una solicitud de compra, se evalúa el presupuesto, se aprueba la solicitud y es entregada al proveedor, luego se realiza la contratación, estableciendo las obligaciones de pago, se realiza la compra y se crea la factura, luego el almacenero realiza la recepción, además archiva los documentos de entrega y notifica la recepción realizada.

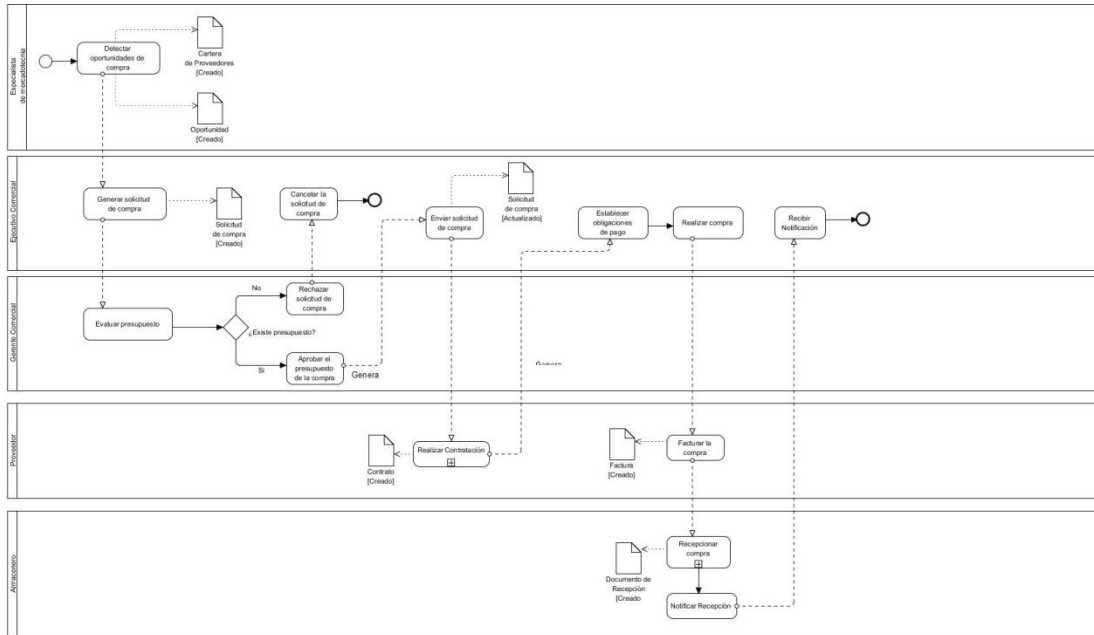


Figura 2. Diagrama de proceso de negocio compra

Proceso de Venta

Para realizar una venta es necesario detectar una oportunidad de venta, el cliente realiza un pedido, el pedido es analizado y luego aprobado, se realiza la contratación, una vez establecidas las obligaciones de pago se realiza la venta junto a la creación de una factura, por último se actualiza el inventario y se notifican las actualizaciones.

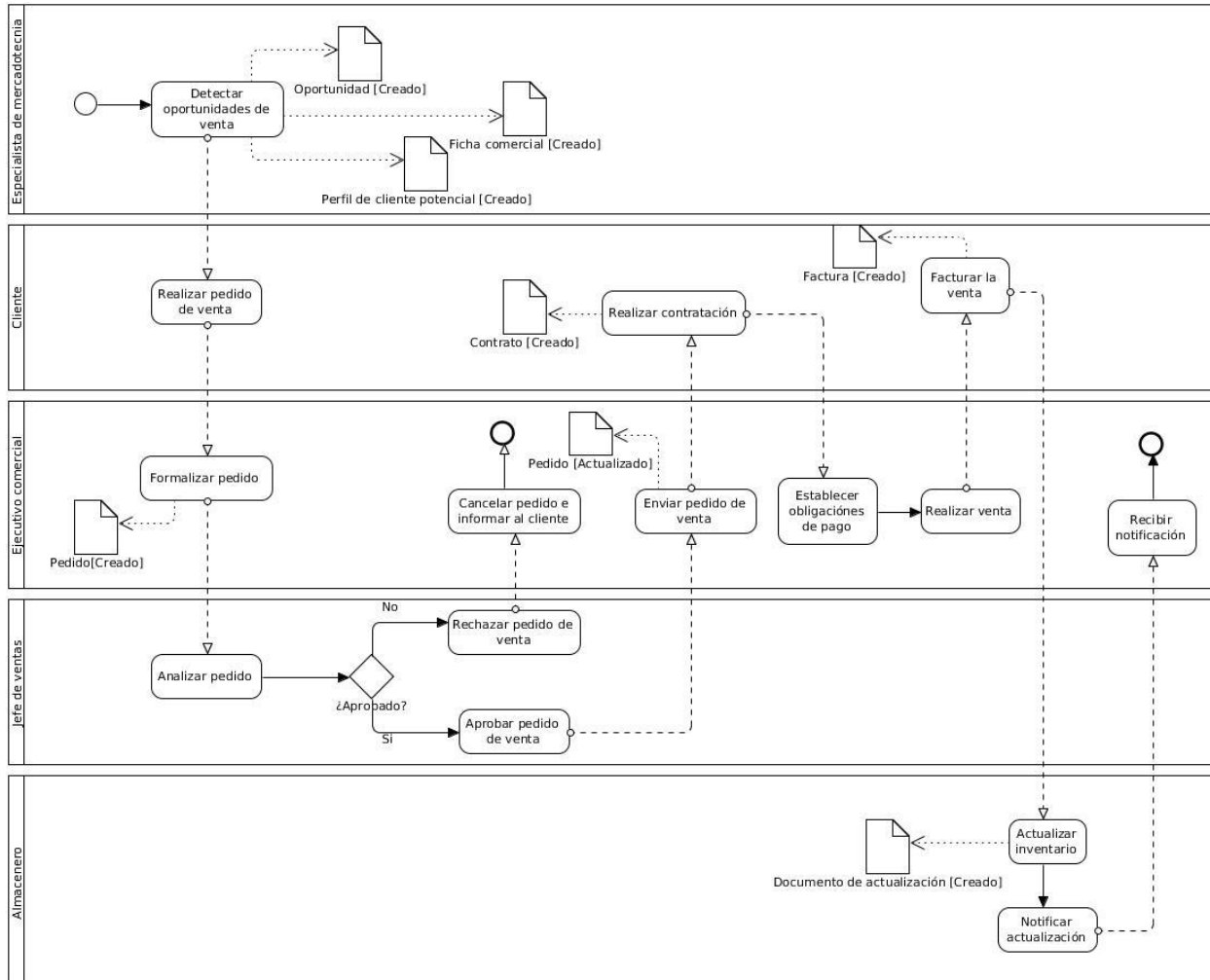


Figura 3. Diagrama de proceso de negocio venta

Las descripciones de flujo de los procesos de negocio se encuentran en los anexos número 1 y 2.

2.4 Representación del Modelo Conceptual

A continuación se muestra el modelo conceptual donde se representan los principales conceptos y sus relaciones pertenecientes a los procesos de Compra y Venta; teniendo como objetivo principal establecer un lenguaje común entre los participantes en el desarrollo del software.

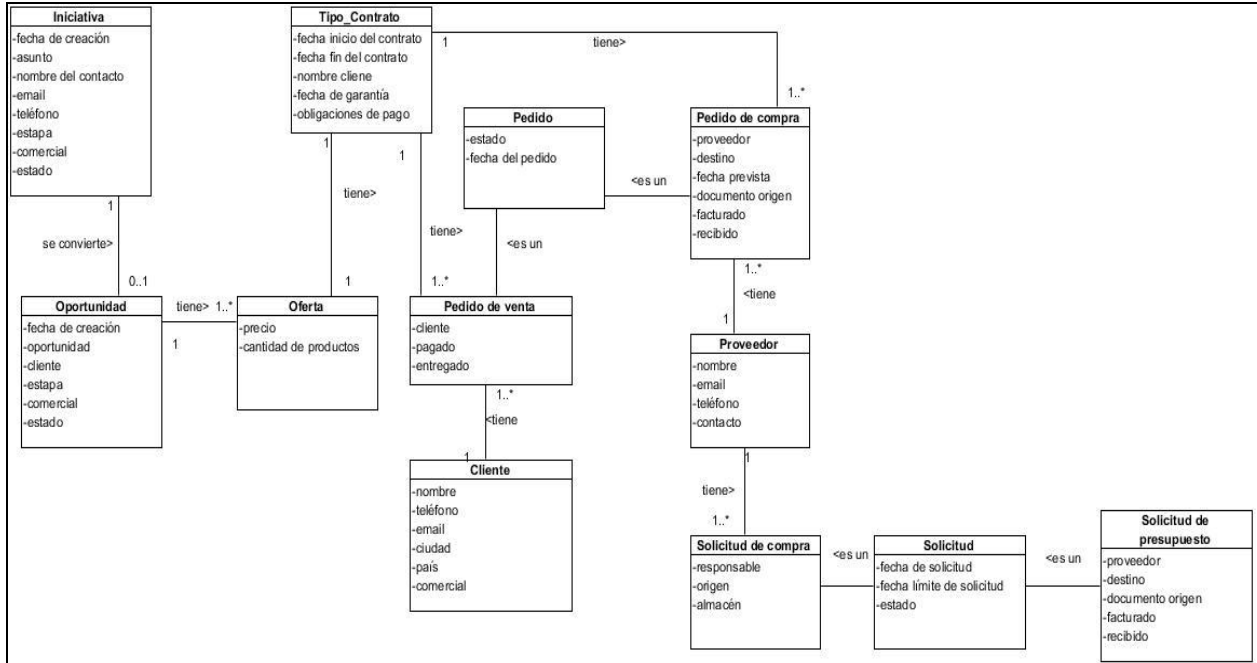


Figura 4. Modelo conceptual de los procesos de Compra y Venta

2.5 Especificación de requisitos

2.5.1 Técnicas utilizadas para la recopilación de requisitos

Entrevistas y cuestionarios: con esta técnica se obtuvo una descripción global de cada uno de los procesos, permitiendo identificar las actividades específicas que se realizan en cada uno de ellos, además de realizar una modelación de los mismos. La plantilla que fue utilizada para la descripción de los procesos fue definida por la dirección del proyecto.

Tormenta de idea: en esta técnica participaron todos los analistas designados al desarrollo del sistema; donde se realizaron debates, escucharon el criterio de cada uno de los participantes para lograr el entendimiento de cada uno de los requisitos capturados, permitiendo determinar los posibles cambios o errores en dicha captura.

Sistemas existentes: esta técnica fue utilizada durante el análisis de varios sistemas existente, ya sean libres o propietarios, tal es el caso de OpenERP y SAP, por otro lado también es útil analizar las distintas salidas de estos sistemas, con el objetivo de que surjan nuevas ideas sobre la base de estas.

2.5.2 Requisitos funcionales

A continuación se muestra el listado de los requisitos funcionales de los procesos de Compra y Venta del sistema de Administración de Relaciones con el Cliente:

Agrupación de requisitos Gestionar solicitud de compra

RF 1.1 Adicionar solicitud de compra.

RF 1.2 Eliminar solicitud de compra.

RF 1.3 Buscar solicitud de compra.

RF 1.4 Modificar solicitud de compra.

RF 1.5 Listar solicitud de compra.

RF 1.6 Consultar solicitud de compra.

RF 1.7 Imprimir solicitud de compra.

Agrupación de requisitos Gestionar solicitud de presupuesto

RF 2.1 Adicionar solicitud de presupuesto.

RF 2.2 Eliminar solicitud de presupuesto.

RF 2.3 Buscar solicitud de presupuesto.

RF 2.4 Modificar solicitud de presupuesto.

RF 2.5 Listar solicitud de presupuesto.

RF 2.6 Consultar solicitud de presupuesto.

RF 2.7 Imprimir solicitud de presupuesto.

Agrupación de requisitos Gestionar pedido de compra

RF 3.1 Adicionar pedido de compra.

RF 3.2 Eliminar pedido de compra.

RF 3.3 Buscar pedido de compra.

RF 3.4 Modificar pedido de compra.

RF 3.5 Listar pedido de compra.

RF 3.6 Consultar pedido de compra.

RF 3.7 Imprimir pedido de compra.

Agrupación de requisitos Gestionar iniciativa de venta

RF 4.1 Adicionar iniciativa de venta.

RF 4.2 Eliminar iniciativa de venta.

RF 4.3 Buscar iniciativa de venta.

RF 4.4 Modificar iniciativa de venta.

RF 4.5 Listar iniciativa de venta.

RF 4.6 Consultar iniciativa de venta.

RF 4.7 Imprimir iniciativa de venta.

Agrupación de requisitos Gestionar oferta

RF 5.1 Adicionar oferta.

RF 5.2 Eliminar oferta.

RF 5.3 Buscar oferta.

RF 5.4 Modificar oferta.

RF 5.5 Listar oferta.

RF 5.6 Consultar oferta.

RF 5.7 Imprimir oferta.

Agrupación de requisitos Gestionar pedido de venta

RF 6.1 Adicionar pedido de venta.

RF 6.2 Eliminar pedido de venta.

RF 6.3 Buscar pedido de venta.

RF 6.4 Modificar pedido de venta.

RF 6.5 Listar pedido de venta.

RF 6.6 Consultar pedido de venta.

RF 6.7 Imprimir pedido de venta.

Agrupación de requisitos Gestionar cliente

RF 7.1 Adicionar cliente.

RF 7.2 Eliminar cliente.

RF 7.3 Buscar cliente.

RF 7.4 Modificar cliente.

RF 7.5 Listar cliente.

RF 7.6 Consultar cliente.

RF 7.7 Imprimir cliente.

RF 8. Requisito funcional Convertir iniciativa en oportunidad

Agrupación de requisitos Nomenclador tipo de contrato

RF 9.1 Adicionar contrato.

RF 9.2 Eliminar contrato.

RF 9.3 Modificar contrato.

Agrupación de requisitos Nomenclador tipo de oferta

RF 10.1 Adicionar oferta.

RF 10.2 Eliminar oferta.

RF 10.3 Modificar oferta.

Agrupación de requisitos Nomenclador tipo cliente

RF 11.1 Adicionar cliente.

RF 11.2 Eliminar cliente.

RF 11.3 Modificar cliente.

Agrupación de requisitos Gestionar contrato

RF 12.1 Adicionar contrato.

RF 12.2 Eliminar contrato.

RF 12.3 Buscar contrato.

RF 12.4 Modificar contrato.

RF 12.5 Listar contrato.

RF 12.6 Consultar contrato.

RF 12.7 Imprimir contrato.

Reportes

RF 13. Análisis de compras.

RF 14. Análisis de recepciones.

RF 15. Análisis de ventas.

En el listado de los requisitos se definieron las funcionalidades a reutilizar del sistema OpenERP. También se definieron otras agrupaciones que son las que realizan un aporte en el presente trabajo, en este caso se encuentran la agrupación de requisitos Gestionar oferta, que es una actividad que realizan los gerentes comerciales con los clientes antes de establecer un contrato de venta y la agrupación de requisitos Gestionar contrato que tanto para las ventas como para las compras debe ser elaborado, ya que es la entrada fundamental para realizar seguimiento tanto de las ventas como de las compras.

2.5.3 Requisitos no funcionales

Los requisitos no funcionales que se muestran a continuación fueron definidos por los arquitectos del proyecto CRM.

Funcionalidad

Idoneidad

- El sistema permitirá generar reportes estándares en formatos PDF, XSL, DOC.
- El sistema permitirá el manejo de archivos e imágenes.

Interoperabilidad

- El sistema será capaz de interactuar con Cedrux mediante WSDL.

Seguridad

- El sistema manejará la seguridad de acceso y administración de usuarios mediante el otorgamiento de privilegios y roles, asignación de perfiles.
- Todo uso de las funcionalidades del sistema requiere la autenticación de los usuarios.
- El sistema concederá acceso a cada usuario autenticado solo a las funciones que le estén permitidas, de acuerdo a la configuración del sistema.
- El sistema manejará mecanismos de encriptación para las contraseñas de los usuarios.

Confiabilidad

Madurez

- Ninguna información que se haya ingresado en el sistema y se haya asociado a alguna operación será eliminada físicamente de la base de datos, independientemente de que para el sistema este elemento ya no exista.

- El sistema implementará el uso de campos obligatorios y validaciones para garantizar la integridad de la información que se introduce por el usuario.

Tolerancia ante fallos

- Ante el fallo de una funcionalidad del sistema, el resto de las funcionalidades que no dependen de esta deberán seguir funcionando.
- Ante el fallo de una funcionalidad del sistema, el resto de las funcionalidades que dependen de esta deberán notificar su imposibilidad de realización debido a la causa original.
- El sistema permite detectar fallos internos y notificar al usuario de la ocurrencia de estos.

Recuperabilidad ante fallos

- El sistema tendrá un servidor de datos de respaldo que se actualizará diario mediante el sistema Bacula.

Usabilidad

Comprensibilidad

- Todos los mensajes de error del sistema deberán incluir una descripción textual del error.
- El sistema será una aplicación web centralizada.
- Se generará un manual de ayuda que le será entregado al cliente de conjunto con el producto.

Cognoscibilidad

- El sistema no utilizará textos de más de 30 caracteres para las etiquetas de la interfaz de usuario y estas se mantendrán siempre que se muestre el mismo campo en cualquier interfaz.
- El idioma de todas las interfaces de la aplicación será el español.
- El sistema será consistente en el uso de abreviaturas, usará la misma abreviación siempre para la misma palabra y nunca en un elemento de selección o menú.

Operabilidad

- Los flujos de navegación para la gestión de cualquier concepto del negocio no excederán las 3 interfaces.
- Los campos de texto tendrán un tamaño estándar de acuerdo con el espacio con que se cuente en el área de la página y en la medida que se llene esa área primaria se agregará la barra de desplazamiento vertical.
- Las etiquetas contendrán los términos significativos del concepto que deseen transmitir.
- El sistema expondrá el menú general desde cualquiera de sus páginas.
- El sistema implementará un mecanismo para el control de la concurrencia a una misma tupla de datos, de manera que se garantice consistencia en la información que visualicen en cada momento los usuarios.

Atracción

- En el sistema no existirán más de 3 interfaces para lograr una funcionalidad completa.
- El sistema diferenciará los mensajes de información de los mensajes de error y de advertencia valiéndose de distintos íconos para cada tipo.
- Se colocará un máximo de 10 campos en los formularios del sistema.
- El sistema presentará los términos capitalizados, es decir, la primera palabra tendrá su primera letra en mayúsculas.
- La tipografía y colores serán estándares en toda la aplicación.

Eficiencia

Rendimiento

- El sistema no excederá los 3 segundos de respuesta al efectuar acciones de cargar un registro (esta cifra no incluye los retardos por concepto de tráfico de red).
- El sistema no excede los 2 segundos para efectuar acciones de salva de información (esta cifra no incluye los retardos por concepto de tráfico de red).

Utilización de recursos

- El sistema interactuará con impresoras para imprimir los diferentes documentos que genere la aplicación como respuesta a las funcionalidades del sistema.

Mantenibilidad

Diagnosticabilidad

- El sistema deberá poseer un mecanismo de almacenamiento, detección y tratamiento de errores.
- El sistema será modular, tomando como criterio para la creación de los módulos los macro procesos de negocio identificados.
- El sistema desagregará las funcionalidades por niveles de reutilización, modificación y/o mantenimiento.

Flexibilidad

- El sistema permitirá agregar nuevas funcionalidades o modificar alguna existente sin romper la estructura y consistencia de los componentes.

Estabilidad

- El sistema estará disponible durante 8 horas, los 7 días de la semana, los 365 días del año.
- El sistema será modular, tomando como criterio para la creación de los módulos los macro procesos de negocio identificados, favoreciendo así la incorporación, modificación o eliminación de funcionalidades.

Portabilidad

Adaptabilidad

- El sistema permitirá la manipulación de dispositivos externos como las impresoras.

Instalabilidad

- El sistema podrá ser instalado en el ambiente especificado en los requisitos tecnológicos para servidores.

Coexistencia

- El sistema interactuará con herramientas ofimáticas y visualizador de ficheros PDF para la presentación de los reportes que genere y los ficheros que importe.

Hardware

Servidor Aplicación

Procesador: 3.00 GHZ

RAM: 1 GB

Disco duro: 160 GB

UPS: 1

Lector de CD: 1

Tarjeta de red: 1

Servidor de Bases de Dato

Procesador: 3.00 GHZ

RAM: 1 GB

Disco duro: 160 GB

UPS: 1

Lector de CD: 1

Tarjeta de red: 1

Servidor Clientes Ligeros

Procesador: 3.00 GHZ

RAM: 1 GB

Disco duro: 160 GB

UPS: 1

Lector de CD: 1

Tarjeta de red: 1

PC_Cliente

Procesador: 1.40 GHZ

RAM: 256 MB (Recomendado 512 MB)

Tarjeta de red: 1

Cliente Ligero

Procesador: 2.0 GHZ

RAM: 256 MB (Recomendado 512 MB)

Tarjeta de Red: 1

2.6 Especificaciones de requisitos

Las especificaciones de requisitos son una versión completa de cómo se va a comportar el sistema que se desea desarrollar. A continuación se muestra como ejemplo la agrupación Gestionar solicitud de compra:

Agrupación de requisitos Gestionar solicitud de compra

Tabla 3. Especificación de requisito Adicionar solicitud de compra

Precondiciones	El proveedor ha sido validado.
Flujo de eventos	
Flujo básico	
1	Se introducen los datos de solicitud de compra: Responsable Fecha de solicitud Fecha límite de solicitud

CAPÍTULO 2: DESCRIPCIÓN DEL SISTEMA

	Origen Almacén	
2	El sistema valida (ver validación 1) los datos introducidos.	
3	Si los datos son correctos el sistema los registra.	
4	El sistema confirma el registro de los datos.	
5	Concluye el requisito.	
Pos-condiciones		
1	Se registró en el sistema una nueva solicitud de compra.	
Flujos alternativos		
Flujo alternativo 4.a Información errónea		
	El sistema señala los datos erróneos y permite corregirlos.	
	El usuario corrige los datos.	
	Volver al paso 2 del flujo básico.	
Pos-condiciones		
1	NA	
Flujo alternativo 4.b Información incompleta		
1	El sistema señala los datos vacíos y permite corregirlos.	
2	El usuario corrige los datos.	
3	Volver al paso 2 del flujo básico.	
Pos-condiciones		
1	NA	
Flujo alternativo *.a El usuario cancela la acción		
1	Concluye el requisito.	
Pos-condiciones		
1	No se registran los datos.	
Validaciones		
1	Se validan los datos según lo establecido en el Modelo conceptual 1.0	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	N/A
Conceptos	Solicitud de compra	Visibles en la interfaz: Responsable Fecha de solicitud Fecha límite de solicitud Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 4. Especificación de requisito Buscar solicitud de compra

Precondiciones	Se ha registrado al menos una solicitud de compra en el sistema
-----------------------	---

Flujo de eventos		
Flujo básico		
1	Se insertan los criterios de búsqueda: Responsable Origen	
2	El sistema muestra un listado de las solicitudes de compra que cumplen los criterios de búsqueda especificados. Se muestran la fecha límite de solicitud, responsable y fecha de solicitud.	
3	Concluye el requisito.	
Pos-condiciones		
1	NA	
Flujos alternativos		
Flujo alternativo 2.a No existen datos que cumplan con los criterios especificados		
1	El sistema notifica que no existen datos que cumplan con los criterios especificados.	
Pos-condiciones		
1	NA	
Validaciones		
1	NA	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	N/A
Conceptos	Solicitud de compra	Visibles en la interfaz: Fecha límite de solicitud Responsable Fecha de solicitud Utilizados internamente: Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 5. Especificación de requisito Listar solicitud de compra

Precondiciones	Se ha registrado al menos un solicitud de compra en el sistema
Flujo de eventos	
Flujo básico	
1	El sistema muestra un listado de las solicitudes de compra. Se muestran fecha límite de solicitud, responsable y fecha de solicitud.
2	Concluye el requisito.
Pos-condiciones	
1	NA

Flujos alternativos		
Flujo alternativo		
1	NA	
Pos-condiciones		
1	NA	
Validaciones		
1	NA	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	N/A
Conceptos	Solicitud de compra	Visibles en la interfaz: Fecha límite de solicitud Responsable Fecha de solicitud Utilizados internamente: Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 6. Especificación de requisito Modificar solicitud de compra

Precondiciones	Se ha registrado al menos una solicitud de compra en el sistema
Flujo de eventos	
Flujo básico	
1	Se selecciona la solicitud de compra a modificar.
2	El sistema muestra y permite editar los datos de la solicitud de compra.
3	Se introducen los datos de la solicitud de compra: Responsable Fecha de solicitud Fecha límite de solicitud Estado
4	El sistema valida (ver validación 1) los datos introducidos.
5	Si los datos son correctos el sistema los registra.
6	El sistema confirma el registro de los datos.
7	Concluye el requisito.
Pos-condiciones	
1	Se modificaron los datos de la solicitud de compra.
Flujos alternativos	
Flujo alternativo 5.a Información errónea	
1	El sistema señala los datos erróneos y permite corregirlos.
2	El usuario corrige los datos.

3	Volver al paso 4 del flujo básico.	
Pos-condiciones		
1	NA	
Flujo alternativo 5.b Información incompleta		
1	El sistema señala los datos vacíos y permite corregirlos.	
2	El usuario corrige los datos.	
3	Volver al paso 4 del flujo básico.	
Pos-condiciones		
1	NA	
Flujo alternativo *.a El usuario cancela la acción		
1	Concluye el requisito.	
Pos-condiciones		
1	No se registran las modificaciones realizadas.	
Validaciones		
1	Se validan los datos según lo establecido en el Modelo conceptual 1.0	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	Paso 1: Buscar solicitud de compra, en la agrupación Gestionar solicitud de compra.
Conceptos	Solicitud de compra	Visibles en la interfaz: Responsable Fecha de solicitud Fecha límite de solicitud Almacén Origen
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 7. Especificación de requisito Eliminar solicitud de compra

Precondiciones	Se ha registrado al menos una solicitud de compra en el sistema
Flujo de eventos	
Flujo básico	
1	Se selecciona la solicitud de compra a eliminar.
2	El sistema verifica (ver validación 1) que se pueda eliminar.
3	Se solicita confirmación para eliminar la solicitud de compra.
4	Si el usuario confirma se elimina la solicitud de compra.
5	El sistema confirma la eliminación.
6	Concluye el requisito.
Pos-condiciones	
1	Se eliminó la solicitud de compra.
Flujos alternativos	

Flujo alternativo 3.a La solicitud de compra ha sido procesada.		
1	El sistema notifica por qué no puede eliminarse la solicitud de compra.	
Pos-condiciones		
1	NA	
Flujo alternativo *.a El usuario cancela la acción		
1	Concluye el requisito.	
Pos-condiciones		
1	No se elimina la solicitud de compra.	
Validaciones		
1	Se validan los datos según lo establecido en el Modelo conceptual 1.0	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	Paso1: Buscar solicitud de compra, en la agrupación Gestionar solicitud de compra.
Conceptos	Solicitud de compra	Visibles en la interfaz: Fecha límite de solicitud Responsable Fecha de solicitud Utilizados internamente: Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 8. Especificación de requisito Consultar solicitud de compra

Precondiciones	Se ha registrado al menos una solicitud de compra en el sistema
Flujo de eventos	
Flujo básico	
1	El sistema permite seleccionar una solicitud de compra.
2	El sistema muestra: Responsable Fecha de solicitud Fecha límite de solicitud Origen Almacén
3	Concluye el requisito.
Pos-condiciones	
1	NA
Flujos alternativos	
Flujo alternativo	
1	NA
Pos-condiciones	

1	NA	
Validaciones		
1	NA	
Relaciones	Requisitos Incluidos	N/A
	Extensiones	Paso 1: Buscar solicitud de compra, en la agrupación Gestionar solicitud de compra.
Conceptos	Solicitud de compra	Visibles en la interfaz: Responsable Fecha de solicitud Fecha límite de solicitud Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Tabla 9. Especificación de requisito Imprimir solicitud de compra

Precondiciones	Se ha registrado al menos una solicitud de compra en el sistema
Flujo de eventos	
Flujo básico	
1	El sistema permite seleccionar los elementos que se desean imprimir.
2	El sistema imprime los datos (ver formatos de salida).
3	Concluye el requisito.
Pos-condiciones	
1	NA
Flujos alternativos	
Flujo alternativo *.a El usuario cancela la acción	
1	Concluye el requisito.
Pos-condiciones	
1	No se imprime la información.
Flujo alternativo 2.a El sistema permite realizar una vista previa	
1	NA
Pos-condiciones	
1	NA
Flujo alternativo * No se pueden imprimir los datos por errores en el sistema, la red o la impresora.	
1	El sistema notifica al usuario que los datos no pueden ser impresos.
Pos-condiciones	
1	NA
Validaciones	
1	NA

Relaciones	Requisitos Incluidos	N/A
	Extensiones	Paso 1: Buscar solicitud de compra, en la agrupación Gestionar solicitud de compra.
Conceptos	Solicitud de compra	Visibles en la interfaz: Responsable Fecha de solicitud Fecha límite de solicitud Origen Almacén
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Las descripciones de los demás requisitos funcionales se encuentran en los Anexos (desde el anexo número 3 hasta el 51).

2.6 Prototipos de interfaz de usuario

Los prototipos de interfaz de usuario ayudan a identificar, comunicar y probar un producto antes de crearlo. La realización de los mismos, logra un entendimiento común entre el cliente y los desarrolladores, solucionando así la mayoría de los cambios posteriores en los proyectos de desarrollo de software. A continuación se muestra como ejemplo las interfaces que tiene una solicitud de compra.

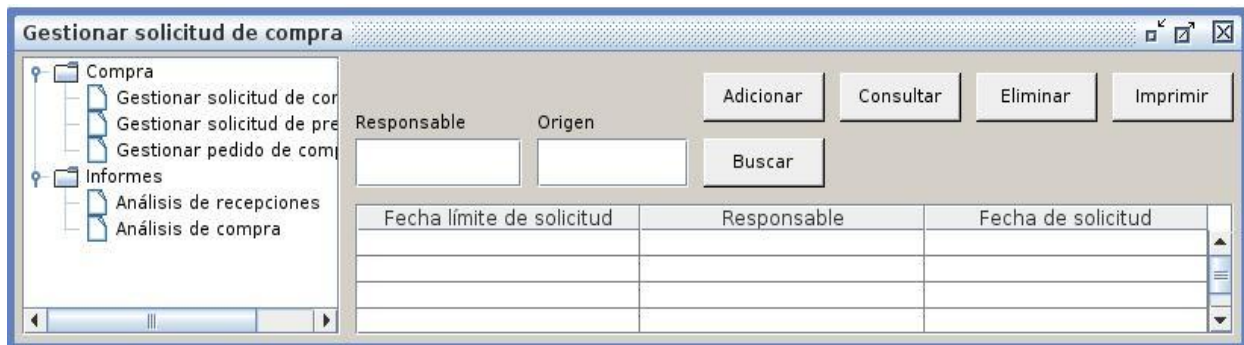


Figura 5. Prototipo de interfaz de usuario para gestionar una solicitud de compra

Adicionar solicitud de compra

Origen	<input type="text"/>	Fecha de solicitud	<input type="text"/> ▼
Responsable	<input type="text"/>	Fecha límite de solicitud	<input type="text"/> ▼
Almacén	<input type="text"/>		

Aceptar Aplicar Cancelar

Figura 6. Prototipo de interfaz de usuario para adicionar una solicitud de compra

Modificar solicitud de compra

Origen	<input type="text" value="Asas"/>	Fecha de solicitud	<input type="text" value="26/02/2012"/> ▼
Responsable	<input type="text" value="Administrador"/>	Fecha límite de solicitud	<input type="text" value="10/02/2013"/> ▼
Almacén	<input type="text" value="Almacén2"/>		

Aceptar Cancelar

Figura 7. Prototipo de interfaz de usuario para modificar una solicitud de compra

Consultar solicitud de compra

Origen	<input type="text" value="Asas"/>	Fecha de solicitud	<input type="text" value="26/02/2012"/> ▼
Responsable	<input type="text" value="Administrador"/>	Fecha límite de solicitud	<input type="text" value="10/02/2013"/> ▼
Almacén	<input type="text" value="Almacén2"/>		

Modificar Cancelar

Figura 8. Prototipo de interfaz de usuario para consultar una solicitud de compra

Para más información acerca de los prototipos de interfaz de usuario ver en los Anexos (desde el anexo número 52 hasta el 79).

2.7 Validación de requisitos

Las técnicas utilizadas para la validación de los requisitos de los procesos Compra y Venta fueron los siguientes:

Prototipos: esta técnica dio la posibilidad de crear una idea de la estructura de interfaz de usuario sin contar con la total funcionalidad del sistema, estas interfaces se crearon a partir de la especificación de los requisitos y se utilizó como herramienta de modelado el Visual Paradigm.

Revisión: después de identificar los requisitos funcionales se reunieron el analista principal de proyecto y el analista del proceso, para realizar un análisis de las plantillas de especificación de requisitos, donde se analizaron cada uno de los requisitos y en el caso de existir incoherencias, falta de algún atributo o poca claridad se le dio solución al problema, logrando una mejor descripción en manos del programador.

2.8 Validación de los resultados por métricas

Los requisitos definidos para el proceso de Compra y Venta fueron comprobados para determinar su especificidad (ausencia de ambigüedad) a través de la métrica de la calidad de especificación de los requisitos de software.

Para desarrollar esta métrica es preciso conocer el valor total de requisitos (R_t), el mismo se halla mediante la siguiente fórmula:

$$R_t = R_f + R_{nf}$$

$$134 = 69 + 65$$

Donde:

R_t : Total de requisitos

R_f : Cantidad de requisitos funcionales

R_{nf} : Cantidad de requisitos no funcionales.

Para determinar la especificidad (ausencia de ambigüedad) de los requisitos. Se explica una métrica basada en la consistencia de la interpretación de los revisores para cada requisito:

Para ello se calcula **Q1** para determinar la especificidad de los requisitos:

$$Q1 = R_{ui} / R_t$$

$$Q1 = 125/134$$

$$Q1 = 0.93$$

Donde:

R_{ui}: Número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas.

Q1: Ausencia de ambigüedad

Cuanto más cerca de 1 esté el valor de **Q1**, menor será la ambigüedad de la especificación, se arrojó como resultado que existieron un grupo de pequeñas ambigüedades en la primera revisión realizada.

Tabla 10. Revisores de requisitos

Revisor	Rol
Ing. Dailin Galafet Céspedes	Analista principal
Ing. Olga Yarisbel Rojas Grass	Analista

2.9 Administración de requisitos

Matriz de trazabilidad: se utilizó para realizar el seguimiento de todos los requisitos y así poder detectar inconsistencias u objetivos no cubiertos. A continuación se muestra la matriz de trazabilidad donde se puede observar la relación que existe entre los requisitos y los procesos.

CAPÍTULO 2: DESCRIPCIÓN DEL SISTEMA

-->	Venta	Compra
Imprimir iniciativa de venta	X	
Análisis de ventas	X	
Análisis de recepciones		X
Análisis de compra		X
Imprimir contrato	X	
Consultar contrato	X	
Listar contrato	X	
Modificar contrato	X	
Buscar contrato	X	
Eliminar contrato	X	
Adicionar contrato	X	
Convertir iniciativa en oportunidad	X	
Imprimir cliente	X	
Consultar cliente	X	
Listar cliente	X	
Modificar cliente	X	
Buscar cliente	X	
Eliminar cliente	X	
Adicionar cliente	X	
Imprimir pedido de venta	X	
Consultar pedido de venta	X	
Listar pedido de venta	X	
Modificar pedido de venta	X	
Buscar pedido de venta	X	
Eliminar pedido de venta	X	
Adicionar pedido de venta	X	

Figura 9. Matriz de trazabilidad procesos-requisitos

Imprimir oferta	X	
Consultar oferta	X	
Listar oferta	X	
Modificar oferta	X	
Buscar oferta	X	
Eliminar oferta	X	
Adicionar oferta	X	
Consultar iniciativa de venta	X	
Listar iniciativa de venta	X	
Modificar iniciativa de venta	X	
Buscar iniciativa de venta	X	
Eliminar iniciativa de venta	X	
Adicionar iniciativa de venta	X	
Imprimir pedido de compra		X
Consultar pedido de compra		X
Listar pedido de compra		X
Modificar pedido de compra		X
Buscar pedido de compra		X
Eliminar pedido de compra		X
Adicionar pedido de compra		X
Imprimir solicitud de presupuesto		X
Consultar solicitud de presupuesto		X
Listar solicitud de presupuesto		X
Modificar solicitud de presupuesto		X
Buscar solicitud de presupuesto		X
Eliminar solicitud de presupuesto		X
Adicionar solicitud de presupuesto		X
Imprimir solicitud de compra		X
Consultar solicitud de compra		X
Listar solicitud de compra		X
Modificar solicitud de compra		X
Buscar solicitud de compra		X
Eliminar solicitud de compra		X
Adicionar solicitud compra		X

Figura 10. Matriz de trazabilidad procesos-requisitos

2.10 Conclusiones

En este capítulo se elaboraron los artefactos relacionados con el negocio, se identificaron los requisitos funcionales aplicando técnicas como las entrevistas, tormenta de ideas y sistemas existentes que facilitaron la obtención de los mismos. Se realizaron los prototipos de interfaz de usuario por cada una de las agrupaciones de requisitos, brindándole al cliente una visión de cómo va a estar estructurado el futuro sistema. A partir de los prototipos y de la revisión técnica formal se validaron los requisitos, con el objetivo evitar errores en próximas etapas de desarrollo del sistema.

Capítulo 3: Diseño de la solución

3.1 Introducción

En el presente capítulo se realizará el diseño del sistema a partir del análisis y las especificaciones de los requisitos que se obtuvieron en el capítulo anterior, estos servirán de guía para la implementación de los procesos Compra y Venta. Se transforman los requisitos al diseño del futuro sistema utilizando una arquitectura definida, logrando adaptar el diseño para que sea consistente con el entorno de implementación.

3.2 Diagrama de componentes

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes, mostrando además las dependencias entre estos. Este tipo de diagrama contiene componentes, interfaces y sus relaciones, puede contener también paquetes que se utilizan para agrupar elementos del modelo. Teniendo en cuenta la arquitectura definida por la dirección del proyecto y el modelo de desarrollo a utilizar se definen como estructura básica a los componentes. Al seleccionarse el sistema OpenERP y este no estar orientado a servicios, es necesaria la utilización del Protocolo para el Acceso Simple de Objetos (SOAP) para consumirlo a través de la librería SOAPpy y para que este pueda brindarlos lo hace a través del Lenguaje de Descripción de Servicios Web (WSDL). A continuación se presenta el diagrama de componentes propuesto para el desarrollo de la solución:

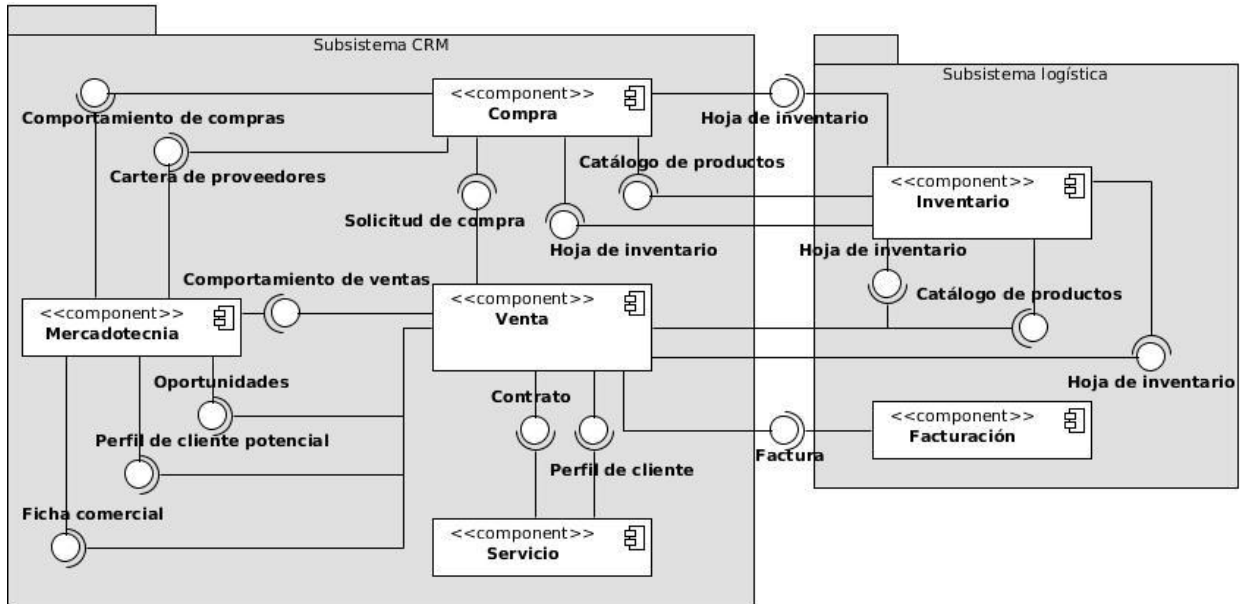


Figura 11. Diagrama de componentes

A continuación, una explicación más detallada de la funcionalidad de cada uno de estos componentes a partir de los servicios que brindan y reciben:

Compra: se encarga de registrar las solicitudes de compra, las solicitudes de presupuesto y los pedidos de compra.

Venta: se encarga de registrar las iniciativas, oportunidades y pedidos de venta.

Mercadotecnia: este componente se encarga de realizar un estudio del mercado, también usará actividades que le ayudará a gestionar las relaciones con las empresas.

Servicio: en este componente se les brindan prestaciones a los clientes como, reparaciones, quejas y devoluciones para lograr una mayor satisfacción de los mismos.

3.3 Patrones de diseño

El empleo de los patrones GRASP tiene como objetivo la descripción de los principios fundamentales del diseño de objetos para la asignación general de responsabilidades. Los utilizados fueron:

Experto: en el diagrama de clases se observa el uso del patrón experto porque las clases poseen responsabilidades específicas a cumplir, de acuerdo con la información que manejan.

Alta cohesión: las clases usan solo lo que necesitan y en su mayoría son recursos propios, por otro lado, permitió que la complejidad de las clases fuese baja.

Bajo acoplamiento: este patrón permite que el diseño de clases fuese más independiente, por si ocurriera algún cambio en alguna de ellas no tuviese una gran repercusión en las demás clases. Aumentó la reutilización y disminuyó la dependencia entre las clases.

Controlador: asigna la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase.

3.4 Patrón de arquitectura Modelo Vista Controlador

El patrón Modelo – Vista – Controlador (MVC) se aplicó en los diagramas de clases, para separar en tres capas diferentes los datos, la interfaz de usuario y la lógica del negocio. La siguiente figura muestra cómo se ve la aplicación de este patrón en un fragmento del diagrama de clases del diseño:

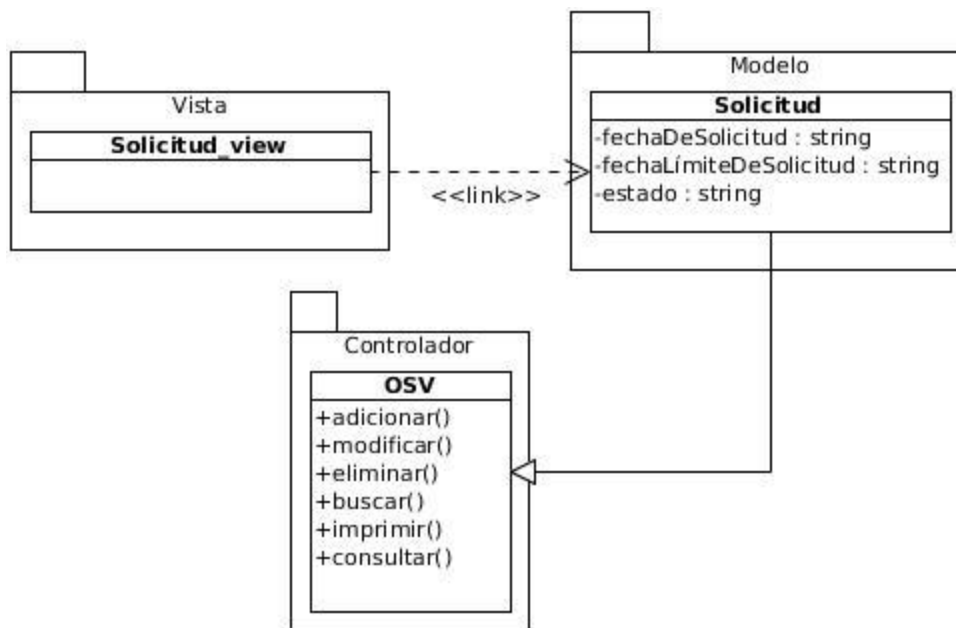


Figura 12. Representación general del patrón MVC en el diagrama de clases

3.5 Diagrama de clases del diseño

En el diseño se modela el sistema y se encuentra su forma, incluyendo la arquitectura, para que soporte todos los requisitos y otras restricciones relacionadas con el entorno de la implementación, tiene impacto en el sistema a desarrollar. Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es utilizado como entrada fundamental de las actividades de implementación.

La siguiente imagen muestra el diagrama de clases del diseño:

CAPÍTULO 3: DISEÑO DE LA SOLUCIÓN

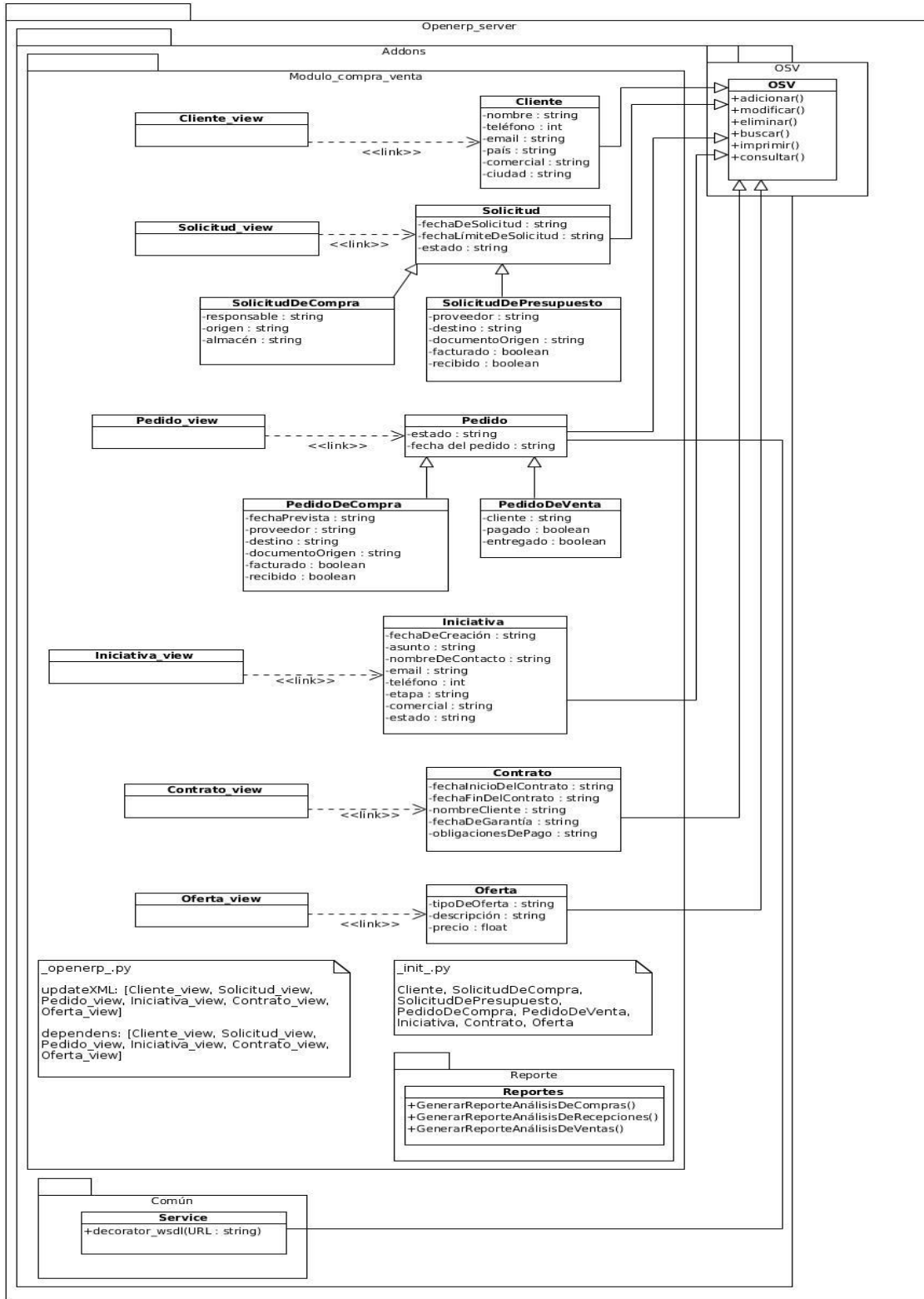


Figura 13. Diagrama de clases del diseño

3.6 Modelado de datos

El modelo de datos describe el comportamiento de los elementos físicos que persisten para dar soporte a la información que almacenará el sistema, con sus atributos correspondientes. Con el objetivo de garantizar la persistencia de los datos se modeló y normalizó el modelo de entidad relación del sistema para la gestión de los procesos Compra y Venta. De estos se muestra seguidamente una porción, que demuestra de forma general como se garantiza la persistencia de datos.

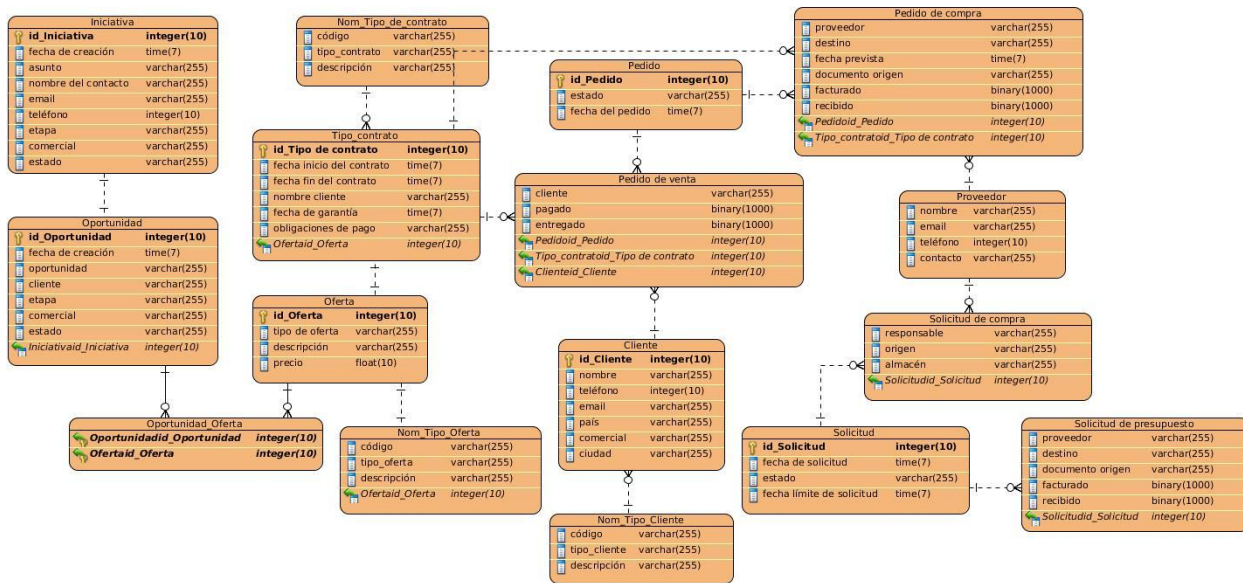


Figura 14. Modelo de entidad-relación

3.7 Descripción de las tablas

A continuación se realiza la descripción de las clases existentes en el modelo de entidad-relación.

Tabla 11 Descripción de iniciativa

Nombre: Iniciativa		
Descripción: La tabla va a recoger los datos de la iniciativa.		
Atributo	Tipo	Descripción
Fecha de creación	Time	Fecha que fue creada la iniciativa

Asunto	Varchar	Descripción de la iniciativa
Nombre del contacto	Varchar	Nombre de la persona a contactar para la iniciativa
Email	Varchar	Email del contacto
Teléfono	Numérico	Teléfono del contacto
Etapas	Varchar	Etapas que se encuentran la iniciativa
Comercial	Varchar	Cargo del que atiende la iniciativa
Estado	Varchar	Estado de la iniciativa

Tabla 12 Descripción de oportunidad

Nombre: Oportunidad		
Descripción: La tabla va a recoger los datos de una oportunidad.		
Atributo	Tipo	Descripción
Fecha de creación	Time	Fecha que fue creada la oportunidad
Oportunidad	Varchar	Descripción de la oportunidad
Cliente	Varchar	Nombre de la persona a contactar para la oportunidad
Etapas	Varchar	Etapas en que se encuentran la oportunidad
Comercial	Numérico	Cargo del que atiende la oportunidad
Estado	Varchar	Estado en que esta la oportunidad

Tabla 13 Descripción de contrato

Nombre: Contrato		
Descripción: La tabla va a recoger los datos de un contrato.		
Atributo	Tipo	Descripción
Fecha de inicio del contrato	Time	Fecha que fue creada el contrato
Fecha fin del contrato	Time	Fecha en que finaliza el contrato
Nombre del cliente	Varchar	Nombre del cliente
Fecha de garantía	Time	Fecha de garantía
Obligaciones de pago	Varchar	Obligaciones de pago

Tabla 14 Descripción de oferta

Nombre: Oferta		
Descripción: La tabla va a recoger los datos de una oferta.		
Atributo	Tipo	Descripción
Tipo de oferta	Varchar	Si la oferta va a ser un servicio o un producto
Descripción	Varchar	Descripción de la oferta
Precio	Float	Precio de la oferta

Tabla 15 Descripción de pedido de venta

Nombre: Pedido de venta		
Descripción: La tabla va a recoger los datos de un pedido de venta.		
Atributo	Tipo	Descripción
Estado	Varchar	Estado del pedido
Fecha del pedido	Time	Fecha que se realizó el pedido

CAPÍTULO 3: DISEÑO DE LA SOLUCIÓN

Cliente	Varchar	Nombre del cliente
Pagado	Binary	Si fue pagado el pedido
Entregado	Binary	Si fue entregado el pedido

Tabla 16 Descripción de pedido de compra

Nombre: Pedido de compra		
Descripción: La tabla va a recoger los datos de un pedido de compra.		
Atributo	Tipo	Descripción
Estado	Varchar	Estado del pedido
Fecha del pedido	Time	Fecha en que se realizó el pedido
Proveedor	Varchar	Nombre del proveedor
Destino	Varchar	Destino del pedido
Fecha prevista	Time	Fecha prevista para realizar el pedido
Documento origen	Varchar	Referencia al documento que ha generado la solicitud de compra
Facturado	Binary	Si se realizó la factura
Recibido	Binary	Si se recibió el pedido

Tabla 17 Descripción de cliente

Nombre: Cliente		
Descripción: La tabla va a recoger los datos de un cliente.		
Atributo	Tipo	Descripción
Tipo de oferta	Varchar	Si la oferta va a ser un servicio o un producto

Nombre	Varchar	Nombre del cliente
Teléfono	Integer	Teléfono del cliente
Email	Varchar	Email del cliente
País	Varchar	País del cliente
Comercial	Varchar	Quien atiende al cliente
Ciudad	Varchar	Ciudad del cliente

Tabla 18 Descripción de solicitud de compra

Nombre: Solicitud de compra		
Descripción: La tabla va a recoger los datos de una solicitud de compra.		
Atributo	Tipo	Descripción
Fecha de solicitud	Time	Fecha en que se realizó la solicitud
Estado	Varchar	Estado de la solicitud
Fecha límite de solicitud	Time	Fecha límite de la solicitud
Responsable	Varchar	Quien atiende la solicitud
Origen	Varchar	Quien realizó la solicitud
Almacén	Varchar	Almacén que realizó la solicitud

Tabla 19 Descripción de solicitud de presupuesto

Nombre: Solicitud de presupuesto		
Descripción: La tabla va a recoger los datos de una solicitud de presupuesto.		
Atributo	Tipo	Descripción
Fecha de solicitud	Time	Fecha en que se realizó la

		solicitud
Estado	Varchar	Estado de la solicitud
Fecha límite de solicitud	Time	Fecha límite de la solicitud
Proveedor	Varchar	Nombre del proveedor al que se realizó la solicitud
Destino	Varchar	Destino de la solicitud
Documento origen	Varchar	Referencia al documento que ha generado esta solicitud
Facturado	Binary	Si se facturo la solicitud
Recibido	Binary	Si se recibió la solicitud

3.8 Métricas para validar el diseño

Las métricas de clases en sentido general, permiten averiguar en qué medida están bien definidas las clases y sus relaciones. Las métricas empleadas son:

Tamaño Operacional de Clase (TOC): Numero de métodos. Responsabilidad, aumenta para TOC+, Complejidad de implementación, aumenta TOC+, Reutilización, disminuye TOC+.

Relaciones entre Clases (bajo) (RC): Numero de relaciones de uso de una clase. Acoplamiento, aumenta RC+. Complejidad del mantenimiento, aumenta RC+. Reutilización, disminuye RC+. Cantidad de Pruebas, aumenta RC+.

Para aplicar la métrica TOC se tuvo en cuenta la cantidad de procedimientos que tenía cada clase, a partir del promedio de los procedimientos y mediante un criterio se obtuvo la categoría (baja, media, alta) para la Responsabilidad, Complejidad y Reutilización.

Luego de efectuar los cálculos para un:

Total de clases	10
Promedio de procedimientos	6.1

Tabla 20 Cantidad de procedimientos por clases

No	Nombre	Procedimientos	Responsabilidad	Complejidad	Reutilización
1	Cliente	6	Baja	Baja	Alta
2	Solicitud	6	Baja	Baja	Alta
3	SolicitudDeCompra	6	Baja	Baja	Alta
4	SolicitudDePresupuesto	6	Baja	Baja	Alta
5	Pedido	6	Baja	Baja	Alta
6	PedidoDeCompra	6	Baja	Baja	Alta
7	PedidoDeVenta	6	Baja	Baja	Alta
8	Iniciativa	7	Media	Media	Media
9	Contrato	6	Baja	Baja	Alta
10	Oferta	6	Baja	Baja	Alta

Se obtuvieron las siguientes gráficas para la Responsabilidad, Complejidad y Reutilización, utilizando las siguientes tablas para obtener las categorías:

Tabla 21 Criterio para calcular la Responsabilidad

Responsabilidad	Categoría	Criterio
	Baja	\leq Prom.
	Media	Entre Prom. y 2^* Prom.
	Alta	$> 2^*$ Prom.

Tabla 22 Criterio para calcular la Complejidad de implementación

Complejidad implementación	Categoría	Criterio
	Baja	\leq Prom.
	Media	Entre Prom. y 2^* Prom.
	Alta	$> 2^*$ Prom.

Tabla 23 Criterio para calcular la Reutilización

Reutilización	Categoría	Criterio
	Baja	\leq Prom.

	Media	Entre Prom. y 2* Prom.
	Alta	<= Prom.

Tabla 24 Responsabilidad de las clases

Responsabilidad	Cantidad de clases
Baja	9
Media	1
Alta	0

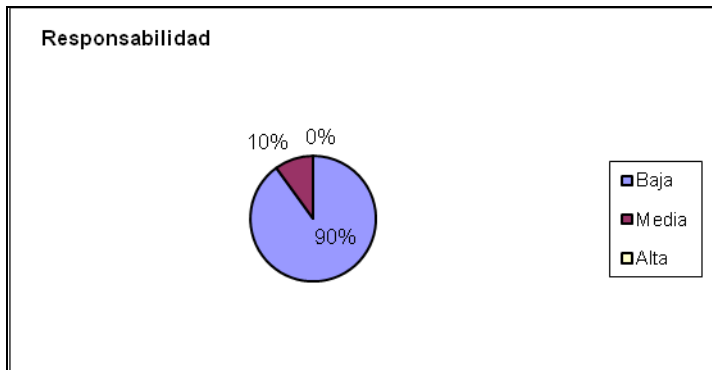


Figura 15 Representación de la responsabilidad

Tabla 25 Complejidad de las clases

Complejidad	Cantidad de clases
Baja	9
Media	1
Alta	0

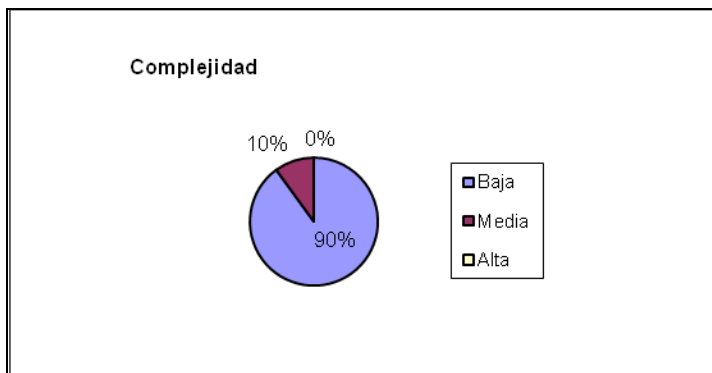


Figura 16 Representación de la Complejidad

La responsabilidad y la complejidad de las clases son relativamente bajas.

Tabla 26 Reutilización de las clases

Reutilización	Cantidad de clases
Baja	0
Media	1
Alta	9

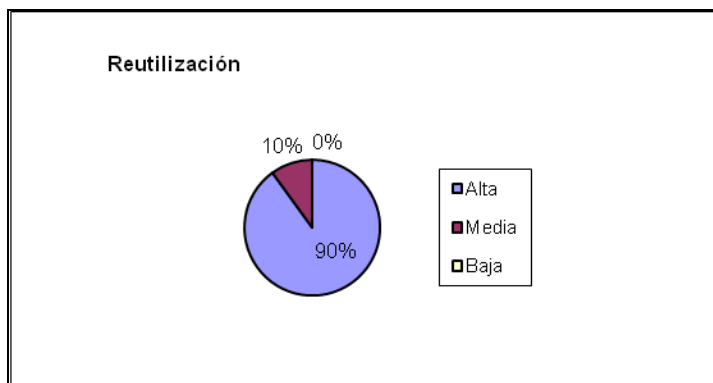


Figura 17 Representación de la reutilización

Cuando la responsabilidad (Figura 15) de las clases entre media y baja es mayor que el 80% se puede decir que cumple con el patrón de alta cohesión, además de que presentan una reutilización alta.

Para aplicar la métrica RC se tuvo en cuenta la cantidad de relaciones que tenían las clases, a partir del promedio de las relaciones y mediante un criterio se obtuvo la categoría (baja, media, alta y ninguna en caso del Acoplamiento) y (baja, media, alta) para la Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas.

Luego de efectuar los cálculos para un:

Total de clases	10
Promedio de asociaciones de uso	2.3

Tabla 27 Relaciones de uso por clases

No	Nombre	Procedimientos
1	Cliente	2
2	Solicitud	2
3	SolicitudDeCompra	2

4	SolicitudDePresupuesto	1
5	Pedido	2
6	PedidoDeCompra	3
7	PedidoDeVenta	3
8	Iniciativa	1
9	Contrato	4
10	Oferta	3

Tabla 28 Acoplamiento de las clases

Acoplamiento	Cantidad de clases
Ninguno	0
Bajo	2
Medio	4
Alto	4

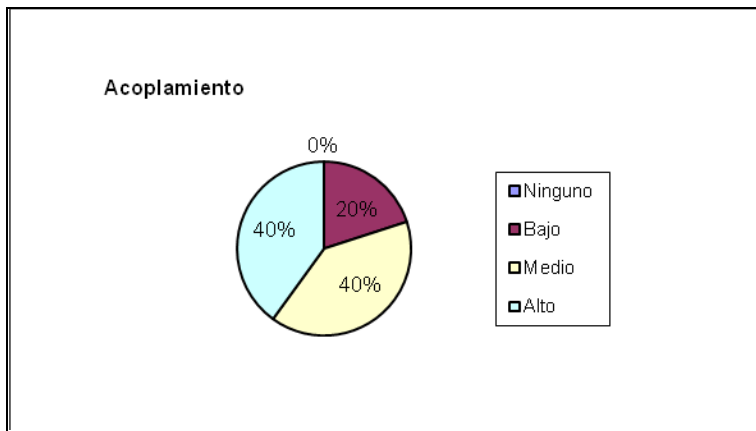


Figura 18 Representación del acoplamiento

Tabla 29 Reutilización de las clases

Reutilización	Cantidad de clases
Bajo	0
Medio	4
Alto	6

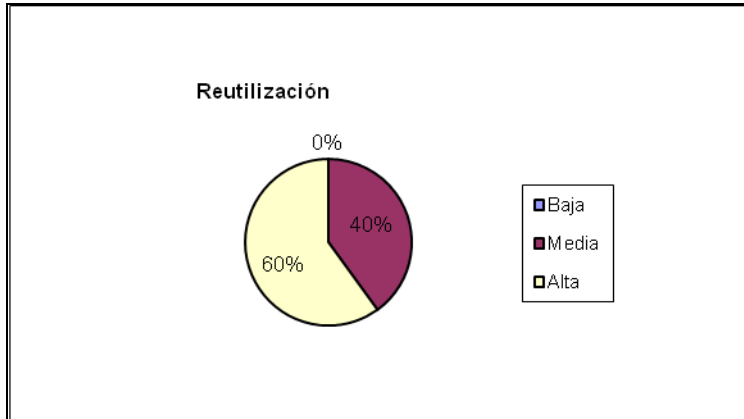


Figura 19 Representación de la reutilización

Tabla 30 Complejidad de mantenimiento de las clases

Complejidad de Mantenimiento	Cantidad de clases
Bajo	6
Medio	4
Alto	0

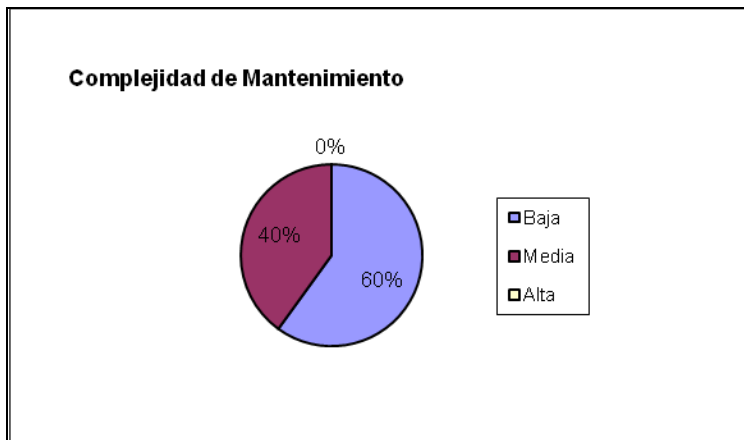


Figura 20 Representación de la complejidad de mantenimiento

Tabla 31 Cantidad de pruebas de las clases

Cantidad de pruebas	Cantidad de clases
Bajo	6
Medio	4
Alto	0

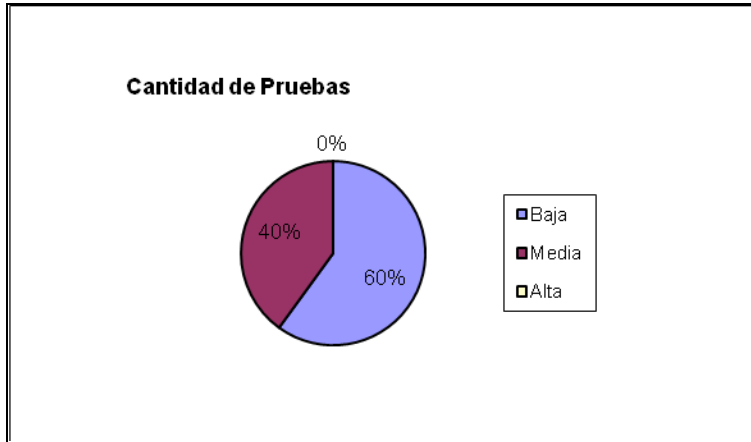


Figura 21 Representación de la cantidad de pruebas

El acoplamiento existente entre las clases es bajo, no existen clases sin acoplamiento, el 20% es bajo, el 40% es medio, y el 40% tiene alto acoplamiento. El nivel de reutilización de las clases es bastante bueno, el 60% de las clases pueden ser reutilizadas.

3.9 Conclusiones

Mediante el diseño realizado en este capítulo donde se generaron los diagramas de clases del diseño, modelo de datos y diagrama de componentes correspondientes a esta etapa, queda hecha la propuesta del sistema facilitando la realización posterior de la implementación. A su vez se aplicaron las métricas para la validación del mismo, arrojando el TOC que las clases contienen baja responsabilidad, baja complejidad y alta reutilización, mientras que el RC demostró la existencia de un acoplamiento bajo y una alta reutilización de las clases.

Conclusiones generales

Después de realizada la investigación y de cumplirse los objetivos propuestos para su desarrollo se arribaron a las siguientes conclusiones:

- Después de analizar los procesos de Compra y Venta y los diferentes CRM estudiados, se propuso utilizar como base la suite de OpenERP, modificando algunas de sus funcionalidades y añadiéndole las necesarias para ajustarlo al modelo económico cubano.
- Mediante la descripción de los procesos de negocios Compra y Venta y los 134 requisitos funcionales definidos se realizó la propuesta del sistema.
- Los requisitos fueron validados mediante las técnicas de revisiones y prototipos de interfaz de usuario arrojando que tenían la calidad requerida para su aprobación.
- Se generaron los artefactos de la etapa del diseño teniéndose en cuenta los patrones a utilizar y la arquitectura definida.
- Se aplicaron las métricas para la validación del diseño, arrojando el TOC que las clases contienen baja responsabilidad, baja complejidad y alta reutilización, mientras que el RC demostró la existencia de un acoplamiento bajo y una alta reutilización de las clases.

Recomendaciones

Realizar la implementación de los procesos Compra y Venta para el Sistema de Administración de Relaciones con el Cliente a partir de los artefactos elaborados en el presente trabajo.

Bibliografía

1. **Rendón, Carlos Augusto.** *La extrategia de ventas.* 2007.
2. **Thompson, Iván.** *Concepto de Venta.* 2012.
3. **Benaque, José Luis.** *Otros conceptos y herramientas de contabilidad y finanzas.* 2007.
4. **Zaragoza, Jesús.** deConceptos. [En línea] 2012. [Citado el: 20 de 1 de 2012.]
5. Web and Macros. [En línea] 2009. [Citado el: 20 de 11 de 2011.]
<http://www.webandmacros.com/crm.htm>.
6. GestioPolis.com. [En línea] 2008. [Citado el: 20 de 11 de 2011.]
<http://www.gestiopolis.com/recursos2/documentos/fulldocs/ger/erpjoel.htm>.
7. Microsoft Pymes y autónomos. [En línea] 10 de 1 de 2012.
<http://www.microsoft.com/business/es-es/Content/Paginas/article.aspx?cbcid=475>.
8. Clientes-Potenciales. [En línea] 2011. [Citado el: 11 de 1 de 2012.] <http://clientes-potenciales.blogspot.com/>.
9. Microsoft Dynamics. [En línea] 2011. [Citado el: 21 de 11 de 2011.]
<http://www.microsoft.com/latam/dynamics/crm/introduccion/default.aspx>.
10. Microsoft Dynamics. [En línea] 2011. [Citado el: 21 de 11 de 2011.]
<http://www.microsoft.com/latam/dynamics/crm/datasheet/default.aspx>.
11. SugarCrmhispano.com. [En línea] 2004. [Citado el: 21 de 11 de 2011.]
http://www.sugarcrmhispano.com/crm/vp-tid:7-pid:2-que_es_sugarcrm.html.
12. SIESA. [En línea] 2011. [Citado el: 21 de 11 de 2011.] <http://www.siesa.com/oferta-de-valor/94.html>.
13. excale Solutions. [En línea] 2011. [Citado el: 25 de 11 de 2011.]
<http://www.excalesolutions.com/servicios/desarrollo-implementacion-formacion-openerp>.
14. Cybercia. [En línea] 2011. [Citado el: 10 de febrero de 2012.]
http://www.cybercia.com/index.php?option=com_content&view=article&id=52&Itemid=71#1.2.
15. Tecnologías información . [En línea] 2009. [Citado el: 25 de 11 de 2011.]
<http://www.tecnologias-informacion.com/crm.html>.
16. OaxRom. [En línea] 2010. [Citado el: 29 de 11 de 2011.]
http://www.oaxrom.com/index.php?option=com_content&view=article&id=29&Itemid=56.

17. INFORMATICA HOY. *INFORMATICA HOY*. [En línea] 2007. [Citado el: 15 de 5 de 2012.] <http://www.informatica-hoy.com.ar/software-crm/SAP-CRM.php>.
18. **Jacobson, Ivar, Boch, Grady, James**. *El Proceso Unificado de Desarrollo de Software*. 2000. pág. 13.
19. **Francisco Ruiz**. Facultad de ciencias económicas. [En línea] 2007. [Citado el: 2 de 12 de 2011.] http://www.economicasunp.edu.ar/02-EGrado/materias/trelew/analisis_sistemas%20II/info/Tecnologia%20para%20la%20Gestion%20de%20Procesos%20de%20Negocio.pdf.
20. **Rumbaugh, James, Ivar**. *El Lenguaje Modificado de Modelado*. 2000. pág. 19.
21. **García, Joaquín**. IngenieroSoftware. [En línea] 2005. [Citado el: 3 de 12 de 2011.] <http://www.ingenierossoftware.com/analisisydiseno/uml.php>.
22. **Ferré Grau, Xavier y Sánchez Segura, María Isabel**. Elquintero. [En línea] [Citado el: 3 de 12 de 2011.] <http://www.elquintero.net/Manuales/UML/umlTotal.pdf>.
23. **Loureiro, Tania Teresa**. *Análisi y diseño de la solución informática para el subsistema de Caja, del sistema de gestión empresarial Cedrux*. 2009.
24. Freed Download Manager. [En línea] 2010. [Citado el: 3 de 12 de 2011.] [http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_14720_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/).
25. **Cecilia**. *OSRMT: Open Source Requirements Management Tool*. 2008.
26. **Gonzales, Carlos**. UsabilidadWeb. [En línea] 2009. [Citado el: 2012 de 6 de 23.] <http://www.usabilidadweb.com.ar/postgre.php>.
27. **Johany**. *Ingeniería De Requerimientos Ingeniería De Software*. 2007.
28. **Díez, A**. “*IRqA y el desarrollo de proyectos: Experiencias Prácticas*”. 2006.
29. **Pressman, R**. *Ingeniería del Soft ware. Un enfoque práctico*. La Habana : Félix Varela, 2005.
30. **Hernández, Dairelis Pérez González y Grenia**. *Guía para realizar Ingeniería de Requisitos a la línea de productos informáticos que utilizan tecnología multimedia*. 2010.
31. **Sommerville, Lan**. “*Ingeniería del Soft ware*”. s.l. : Pearson, 2005.
32. **Chaves, Michael Arias**. *La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software*. 2006.
33. **Lago, Ramiro**. *Patrones de diseño*. 2007.

34. **Visconti, Marcello.** *Fundamentos de Ingeniería de.* 2008.
35. **autores, Colectivo de.** *Arquitectura y Patrones de diseño.* 2009.
36. **Alvarez, Miguel Angel.** *Lenguaje de programación de propósito general, orientado a objetos, que también puede utilizarse para el desarrollo web.* 2009.
37. **Caballero, Julio.** *Trabajo de Investigación de XML.* 2011.
38. **Montalvo, Marlene Melián.** *XML el nuevo lenguaje universal.* 2010.
39. **Montilva, Jonás A.** *Modelado de Negocio.* 2007.
40. **Chile, Gobierno de.** *Administrador de esquemas y metadatos.* [En línea] 2010. [Citado el: 17 de 3 de 2012.] <http://www.aem.gob.cl/glosario.html>.
41. **Herrera, Cristhian.** *Gestión de las Relaciones con los Clientes.* Quito (Ecuador) : s.n., 2005.
42. **Kotler, Philip.** *Dirección de Mercadotecnia.* s.l. : Pearson Educación, 2003. Octava Edición.
43. *MarketingPower.com.* [En línea] American Marketing Association. [Citado el: 17 de 10 de 2011.] <http://www.marketingpower.com/>.
44. **Stanton, Etzel y Walker.** *Fundamentos de marketing.* México : s.n., 2004.
45. *Web and Macros.* [En línea] 2009. [Citado el: 19 de 10 de 2011.] <http://www.webandmacros.com/crm.htm>.
46. **CRM Y GIS: ESTADO DEL ARTE Y EVOLUCIÓN. Palacios, Joaquín M^a Fernández González y Ricardo Colomo.** Sevilla : s.n., 2006.
47. **Rosero Arrieta, Christian Andrés y Mendoza Mera, Jaime Alfredo.** *Integración de Sugar CRM con ASTERISK.* Guayaquil, Ecuado : s.n., 2009.
48. **Bustamante G, Wayner X., y otros.** *Análisis Comparativo entre MICROSOFT CRM Y SUGAR CRM.* Loja – Ecuador : s.n., 2007.
49. *Microsoft Dynamics CRM.* [En línea] 2011. [Citado el: 11 de 2 de 2012.] <http://crm.dynamics.com/es-es/marketing>.
50. *SAI.* [En línea] [Citado el: 3 de 11 de 2011.] <http://www.sai.es/index.php/crm/goldmine.html>.
51. *GoldMine.* [En línea] 2008. [Citado el: 3 de 11 de 2011.] <http://www.goldmine.com.ar/beneficios.htm>.
52. *vtigerCRM Spain.* [En línea] 2011. [Citado el: 3 de 11 de 2011.] <http://crmevolutivo.com/>.

53. Microsoft.com. [En línea] [Citado el: 3 de 11 de 2011.]
<http://www.microsoft.com/spain/businesssolutions/dynamics/productos/crm/default.msp>.
54. Cybercia. [En línea] enero de 2011. [Citado el: 3 de 11 de 2011.]
http://www.cybercia.com/index.php?option=com_content&view=article&id=52&Itemid=71.
55. **Pressman, Roger.** *Ingeniería del Software: Un enfoque práctico.* s.l. : McGraw-Hill , 2005. ISBN: 9701054733.
56. BuenasTareas. [En línea] 2011. [Citado el: 3 de 11 de 2011.]
<http://www.buenastareas.com/ensayos/Ingenieria-De-Requisitos/3483123.html>.
57. **Pressman, Roger.** *Ingeniería del Software. Un enfoque práctico. 5ta. Edición.* s.l. : McGraw-Hill, 2002. ISBN 84-481-3214-9.
58. **Jacobson, Ivan, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software.* s.l. : Addison Wesley, 2000.
59. **Olmos, Lic. Yoansy Pérez.** *Técnicas y herramientas de la ingeniería de requisitos adecuadas para simuladores reales.* La Habana : s.n., 2009.
60. **E M.J Escalona, N. Koch.** *Ingeniería de Requisitos en Aplicaciones para la Web:Un estudio comparativo.* 2002.
61. **Angel.** <http://geektheplanet.net/5462/patrones-gof.shtml>. [En línea] 2011. [Citado el: 10 de 5 de 1012].