

Universidad de las Ciencias Informáticas
Facultad 9



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

**ANÁLISIS Y DISEÑO DEL SISTEMA DE FACTURACIÓN Y COBRO
PARA LA EMPRESA DE GAS MANUFACTURADO.**

Autor:

Helimay Rivero Varona

Tutor:

Ing. Alfonso Chaveco Laurencio

Co-Tutor

Ing. Armando Ortiz Cabrera

Ciudad de La Habana

Junio de 2009



"La revolución se lleva en el corazón no en la boca para vivir de ella"

Ernesto "Che" Guevara.

Agradecimientos

Agradezco a mis dos soles porque gracias a ellas no solo llegué a la Universidad, ni me mantuve, sino que se hizo realidad uno de nuestros grandes sueños. Por siempre estar conmigo en todas las etapas por las que he atravesado, darme los mejores consejos que siempre vienen de las personas que uno más quiere, admira y respeta. Mom y abue para uds, el más grande AGRADECIMIENTO, uds fueron mi motor impulsor.

A nuestro Comandante, nuestra Revolución. A aquellas personas que confiaron en mí durante estos cinco años y a las que no, porque me dieron aliento a crecerme cuando más me hacía falta y confianza.

A las personas que aunque no estén presentes físicamente, en su momento me dieron no solo la mano cuando más lo necesitaba, Tío Ary para ti este trabajo, siempre te tendré en mi corazón, fuiste como un padre, no solo para mí sino para mi madre.

A aquellas personas que físicamente están y por su voluntad se marcharon.

A toda mi familia aquí en la capital que bastante que corrieron conmigo, tía Maga, Tata, Gladys, Nohemí en fin Santiago de las Vegas y Guanabo.

A mis amistades que estuvieron siempre cuando más las necesité, el Rolo, que por siempre que nos cambiaban de grupo siempre moría a la orilla conmigo. Por la paciencia que tuvo en enseñarme y a crecerme con sus consejos y con su "buen humor". Al Magdy que no fue mucho tiempo pero en el poco tiempo demostró ser un amigo de verdad que todos quisieran tener.

A mi tutor por haberme ayudado a levantar cuando me hizo falta y guiarme en gran medida para ir saliendo a flote.

A todos Uds. por estar aquí, sepan que siempre estarán en mi corazón como el más grato recuerdo de mi vida como estudiante. Gracias.

Dedicatoria

Dedico este trabajo a mis dos seres más queridos, mi linda mamá y mi linda abue que siempre han estado presentes en todo y que me han hecho crecer ante las diversas circunstancias por las que pasé durante estos años.

A las personas que me ayudaron a salir de algunos baches de la carrera, no solo con sus consejos serios sino de manera jocosa pero constructiva, siempre dándome el mejor aliento para seguir adelante.

A mis hermanos de escuela que siempre estuvieron allí cuando de verdad hacían falta.

Resumen

En la Empresa Cubana de Gas Manufacturado a diario fluye un elevado volumen de información relacionada con la Facturación y Cobro de los clientes de la empresa y la gestión de la misma se realiza de manera manual. Esto trae como consecuencia la pérdida, desactualización y descontrol en la información que se gestiona y no se encuentra de manera centralizada sino de manera distribuida por varias Casas Comerciales en algunos municipios de Ciudad de La Habana.

En el presente trabajo se propone realizar el Análisis y Diseño del sistema que se lleva a cabo en la Empresa Cubana de Gas Manufacturado, definiendo las insuficiencias y los problemas que la afectan hoy en día como lo son el tipo de tarifa que presenta, la manera en que la información está distribuida, además de modelar un Sistema para su automatización. A raíz de esta propuesta y con su seguimiento, se pretende lograr una mejor comprensión del problema para el Análisis y Diseño del mismo, asegurando la mejor solución, confiabilidad y cumplimiento de las necesidades del cliente convirtiéndolo en el mayor agrado para el mismo. Se pretende garantizar la uniformidad además de un control centralizado que garantice el buen funcionamiento y la seguridad de los Procesos así como que la tarifa sea la adecuada para efectuar el proceso de Facturación según el consumo de cada cliente.

PALABRAS CLAVES

Sistema de Gestión de la Información, SistMet, Gas Manufacturado, Gestión de la Información, Análisis, Diseño.

Índice

Introducción	8
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	13
Introducción	13
1.2 Conceptos asociados para el dominio del problema.	16
Definición de procesos dentro de la UEB Comercial.....	17
1.3 Algunos Sistemas de gestión.	18
1.4 Análisis de otras soluciones existentes en Cuba y en el mundo.	19
CONCLUSIONES.....	22
CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.....	23
2.1 Análisis de las Metodologías para el Desarrollo de Software.....	23
Lenguaje Unificado de Modelado (UML)	24
¿QUÉ ES UML?	24
2.2 Comparación entre las metodologías	26
2.3 Metodologías de Software	27
Metodología Extreme Programming (XP)	27
Metodología SCRUM.....	28
Metodología Proceso Unificado de Desarrollo (RUP)	29
Herramientas CASE (Computer-Aided Software Engineering).....	31
Visual Paradigm	33
Rational Rose Enterprise	34
Fundamentación de la metodología, lenguaje y herramientas de modelado seleccionadas para el desarrollo.	35
Análisis y Diseño	36
Rol de Analista	38
Rol de Diseñador.....	38
CONCLUSIONES.....	39
CAPÍTULO 3: Análisis y Diseño de la solución propuesta.	40

3.1 Descripción de los procesos del negocio.....	40
3.2 Información que se maneja.....	41
3.3 Modelo de Negocio.....	42
3.4 Diagrama de Casos de Uso del Negocio.....	46
3.5 Realización de los Casos de Uso del Negocio.....	47
3.6 Reglas del Negocio.....	57
3.7 Especificación de los requisitos del Software.....	59
3.7.1 Requisitos Funcionales.....	59
3.7.2 Requisitos No Funcionales.....	64
3.7.2.1 Usabilidad.....	64
3.7.2.2 Apariencia o Interfaz Externa.....	65
3.7.2.3 Portabilidad.....	65
3.7.2.4 Confiabilidad.....	65
3.7.2.5 Rendimiento.....	65
3.7.2.5 Soporte.....	66
3.7.2.6 Requerimiento de ayuda y documentación.....	66
3.7.2.7 Restricciones de diseño e implementación.....	66
3.7.2.8 Requerimientos de Hardware.....	66
3.7.2.9 Requerimientos de Software.....	67
3.7.2.10 Seguridad.....	68
3.8 Descripción del sistema propuesto.....	68
3.9 Definición de Casos de Uso del Sistema.....	69
3.9.1 Actores del Sistema.....	69
3.10 Diagrama de Casos de Uso del Sistema.....	70
3.10.1 Proceso de Facturación.....	71
3.10.2 Proceso de Cobro.....	80
3.11 Análisis del Sistema.....	87
3.11.1 Diagramas de clases del análisis y de interacción.....	87

3.12 Diagramas de clases del análisis	87
3.12.1 Modelo de Análisis	87
3.13 Diagrama de Interacción.	88
3.13.1 Diagrama de Secuencia.....	88
3.13.2 Diagrama de Colaboración.....	88
3.14 Diagramas de colaboración de algunos Casos de Uso.	89
3.15 Diagramas de Clases del Diseño.	91
3.16 Patrones	101
3.16.1 Patrones de Diseño.....	101
CONCLUSIONES	107

Introducción

Con el transcurso de los años y el desarrollo acelerado de la tecnología la vida del hombre ha cambiado, convirtiéndose en una poderosa herramienta para el desarrollo de nuevas técnicas de manufactura, comercio, comunicaciones y ha hecho que el flujo de la información sea más rápido y seguro, en determinadas actividades donde su uso se ha hecho indispensable.

La automatización de procesos complejos para lograr una mayor eficiencia, la confiabilidad de las bases de datos de conocimientos, la reducción de los errores humanos gracias a la estandarización que puede brindar un sistema informático, son varias de las razones por las que se puede decir que el mundo se encuentra inmerso en una nueva era: la era de la informatización. Gracias al acelerado desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) existen un sinnúmero de programas para facilitar el trabajo en cualquier área, la economía, el comercio, las finanzas, la producción, las telecomunicaciones son un ejemplo de estas.

Cuba como uno de los tantos países del mundo ha establecido dentro de sus objetivos estratégicos la informatización de la sociedad. En este sentido, el Ministerio de Industria Básica (MINBAS), específicamente la Empresa de Gas Manufacturado, ha logrado avances y actualmente cuenta con el Sistema Metrado (SistMet). La Empresa de Gas Manufacturado se encarga de la distribución, control y cobro del servicio de gas manufacturado a domicilio. Dentro de los servicios que brinda se encuentran la instalación del sistema de gas a particulares, reparación de sistemas de gas en caso de averías y cobro no solo por la reparación sino también por el servicio que presta dicha entidad además de la facturación del mismo. Dentro de los productos que brinda se encuentran los relojes contadores de gas, cocinas de gas.

Algunos procesos que se realizan en la empresa presentan ineficiencias que no solo afectan a la empresa sino a los clientes, dentro de las cuales tenemos:

- La facturación se realiza mediante una tarifa fija y no una tarifa escalonada, lo que significa que no se mide el consumo real del cliente, siempre va a realizarse la operación con un número constante, y no con una escala según el consumo mensual.

- El cobro del servicio solo se efectúa en la Unidad Eléctrica Básica de la Casa Comercial (de ahora en adelante UEB de la Casa Comercial), este proceso es un poco incómodo para los clientes a los cuales se le presta el servicio de gas manufacturado porque solo pueden pagar en dicha Casa Comercial y no en sucursales cercanas que también prestan el servicio.
- El cierre contable mensual, se retrasa, la información no está centralizada y tienen que hacerla llegar mediante personas encargadas a recoger los talonarios que se registran en cada uno de los municipios de Ciudad de La Habana donde se presta el servicio, provoca pérdida de la misma así como retraso a la hora de realizar uno de los servicios antes mencionados.
- El Sistema con que cuenta la empresa es poco adaptable a los cambios que requiere la entidad como por ejemplo realizar un nuevo tipo de factura y poder mantener la información centralizada para el mejoramiento del trabajo.
- SistMet, no cuenta con la implementación de una realización de una tarifa escalonada.
- El Sistema no permite incluir dentro del proceso de facturación el cobro de servicios post venta al cliente por Ejemplo: Variaciones en la instalación del servicio de gas.

Dado el problema anterior es necesaria la creación de un sistema automatizado partiendo de la modelación de los procesos en cuestión como la Facturación y el Cobro de la Empresa de manera tal que el estudio de los mismos permita un mejor uso del sistema.

Dichos procesos se han convertido en un servicio para la población donde se le realiza su factura y cobro a través de la Empresa de Gas Manufacturado.

La empresa cuenta con un sistema automatizado el cual presenta algunas dificultades como ser poco adaptable a los cambios que puedan ocurrir, la gestión de la información se realiza mayoritariamente de forma manual lo que provoca la pérdida de información y atrasos en los procesos de facturación y cobro de la empresa. Darle cumplimiento a algunas de las necesidades que se presentan en dicho sistema es complejo, por lo que se hizo necesario garantizar la gestión y control de los procesos de la empresa dentro de ellos los de Facturación y Cobro para la implementación de un nuevo sistema que pueda satisfacer las demandas de la entidad además de ser lo más configurable y adaptable posible.

En Cuba existen hasta el momento varios sistemas que realizan este tipo de procesos, tales como SistMet (Sistema Metrado) sistema existente en la empresa de Gas Manufacturado perteneciente al MINBAS (Ministerio de Industria Básica), VERSAT-SARASOLA, sistema contable financiero perteneciente al MINAZ (Ministerio de la Azúcar) y la realización de un sistema ERP (Planeación de Recursos de una Empresa). Este último es reconocido a nivel mundial por su integridad a todo tipo de Empresas, y adaptabilidad. (“epígrafe 1.3”)

Como resultado de la situación antes expuesta se plantea el siguiente **problema científico. Inexistencia de un modelo de Análisis y Diseño para el desarrollo de un software para los procesos de Facturación y Cobro establecidos en la Empresa de Gas Manufacturado.**

El **Objeto de Estudio** lo constituyeron **Los procesos de Facturación y Cobro de la Empresa de Gas Manufacturado.**

Dado el objeto de estudio se propone el siguiente **Campo de Acción** que es el **Análisis y Diseño de Sistemas Automatizados para la Facturación y Cobro de la Empresa de Gas Manufacturado.**

Con el **Objetivo General** de **realizar el Análisis y Diseño del Sistema de Facturación y Cobro de la Empresa de Gas Manufacturado.**

Con la siguiente **idea a defender. Después de realizar el análisis y diseño del Sistema a los procesos de Facturación y Cobro es posible que el mismo contribuya a la implementación de un sistema que pueda automatizar los procesos de Facturación y Cobro en la Empresa Cubana de Gas Manufacturado.**

Para darle cumplimiento a este objetivo se realizaron las siguientes **Tareas de Investigación**:

- Caracterizar formas y sistemas de gestión de Facturación y Cobro en Cuba y algunas partes del mundo.
- Caracterizar la aplicación que se utiliza actualmente para el control de los datos en la Empresa de Gas Manufacturado.
- Definir qué elementos pueden ser mejorados y qué nuevas funcionalidades se pueden tener en cuenta.
- Caracterizar las principales herramientas a utilizar en el análisis y diseño para el sistema propuesto para una implementación de la aplicación y el funcionamiento del nuevo sistema.
- Realizar un estudio del Arte de las Metodologías del desarrollo del software para escoger la metodología y las herramientas para la solución de la investigación.
- Modelar, especificar, caracterizar y documentar cómo son los procesos de Facturación y Cobro de la Empresa de Gas Manufacturado.

Para la realización de la investigación fue necesario utilizar algunos **Métodos de Investigación** como:

Métodos Teóricos:

- **Analítico – sintético:** Para llegar a conclusiones a partir de la documentación consultada para entender fenómenos relacionados con los procesos de facturación y cobro, así como el flujo de la información.
- **Histórico – lógico:** Para conocer las tendencias actuales en cuanto a la gestión de la información de los procesos de Facturación y Cobro, surgimiento del gas manufacturado en Cuba y uso de tecnologías de desarrollo.

Métodos Empíricos:

- **Entrevista:** Entrevistas a trabajadores de la Empresa para obtener información actualizada sobre la realización de los procesos antes mencionados y el criterio especializado sobre la realización de los mismos.

Breve descripción de la estructuración del contenido

El documento está estructurado en cuatro capítulos, donde a continuación se expondrán algunas características de los mismos.

Capítulo 1: En el desarrollo de este capítulo se hace un estudio del estado del arte de las diferentes herramientas y tecnologías vinculadas a la solución propuesta, como por ejemplo situación de la Empresa en cuestión, procesos que se realizan en ella, flujo de la información. Se realiza además un análisis de las metodologías, tecnologías candidatas, y roles fundamentales del flujo de trabajo análisis y diseño, así como de algunos patrones posibles a necesitar a lo largo del trabajo.

Capítulo 2: En el capítulo se abordará el tema del análisis de la metodología de desarrollo, así como la justificación del uso de la misma, lenguaje de modelado, herramientas CASE y comparaciones entre algunas metodologías de desarrollo.

Capítulo 3: En el desarrollo de este acápite se aborda el estudio del negocio, donde se describen los procesos a automatizar, así como actores y trabajadores que actúan en los procesos de Facturación y Cobro en la Empresa de Gas Manufacturado. La descripción de los requerimientos para la correcta realización de los mismos, etc. Además de la información que se maneja en los procesos y las reglas del negocio para una mejor aceptación por el cliente. Se muestra el Análisis y el Diseño de la solución propuesta así como algunos de los diagramas realizados para un mejor entendimiento, los conceptos asociados a estos flujos de trabajo. La definición de cuáles son dichos diagramas y su comportamiento.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el año 1844, en virtud de la concesión dictada a favor de Don Jaime Rolf y Don Migue de la Silva se crea la Sociedad Anónima "Compañía Española del Alumbrado de Gas" para proveer de gas purificado al alumbrado público de la Ciudad de la Habana.

Al fundarse la fábrica de gas en Melones en 1880, la mayoría de las máquinas eran de vapor y en la medida que se fue desarrollando fueron sustituidas por máquinas q utilizaban la energía eléctrica. La construcción más antigua de la fábrica es la casa de las calderas que data de 1880, fecha en que se construyeron los gasómetros (Instrumento para medir el gas). (1)

El gas manufacturado se utiliza como combustible y como materia prima en la industria petroquímica. Como combustible se emplea por su gran poder calorífico, por ser su combustión fácilmente regulable y por ser limpia y producir escasa contaminación. Como materia prima es la más adecuada para la fabricación de amoníaco –producto base de toda la industria de abonos nitrogenados– y también del metanol –producto que se utiliza en la fabricación de plásticos y proteínas sintéticas–. A partir del gas manufacturado se obtienen materias primas de base en la industria petroquímica (etileno, butadieno y propileno). Y como fuente de calor para la producción de vapor y de electricidad.

¿Qué es el Gas Manufacturado?

Son los gases pertenecientes a la primera familia y sus componentes principales son el hidrógeno y el monóxido de carbono. Se obtiene de su fabricación a partir de naftas livianas o gas natural.

Su poder calorífico superior es de aproximadamente 4600 Kcal/m³ y su composición típica es:

Componente	Símbolo
Hidrógeno	H ₂
Oxido de Carbono	CO
K Anhídrido Carbónico	CO ₂
Metano	CH ₄
Otros	-

Tabla 1.1 Tabla de los elementos que conforman el Gas Manufacturado.

¿Qué tipos de gases combustibles hay?

Básicamente existen tres familias de gases combustibles:

Primera Familia: constituida por los gases manufacturados

Segunda Familia: formada por los gases naturales

Tercera Familia: formada por los comúnmente llamados GLP o Gases Licuados de Petróleo o Gas LP; son el Propano y el Butano. (2)

El gas natural es el combustible fósil más respetuoso con el medio ambiente, es el que tiene menos contenido de dióxido de carbono y el que produce menos emisiones a la atmósfera. Esta característica le permite contribuir a la disminución del efecto invernadero. A todo ello hay que añadir su versatilidad. Las nuevas aplicaciones de esta energía, como la cogeneración, la generación eléctrica con ciclos combinados y su uso como combustible para vehículos, están impulsando el consumo del gas natural, que está llamado a ser la energía del siglo XXI.

Gas Manufacturado en Cuba

La popular y muchas veces deficitaria "luz brillante" (kerosén), y los múltiples inventos caseros que para cocer los alimentos emplearon las familias capitalinas durante los años más difíciles del período especial,

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

van quedando como recuerdos del pasado. Hoy 243 033 núcleos usan el Gas Manufacturado y 347 105 el Gas Licuado.

Es el resultado de un programa inversionista que, desde su inicio a finales de 1998, no ha dejado de avanzar, aunque las dificultades financieras le hayan restado celeridad.

En diciembre de 1998, solo 120 mil núcleos, residentes en los municipios centrales de la provincia, contaban con gas manufacturado (el llamado gas de la calle), y el licuado llegaba a 153 000. Prácticamente todas las zonas periféricas del territorio, carecían de esos servicios.

Hoy las nuevas instalaciones del gas manufacturado se realizan en Mariana, Cerro y Plaza de la Revolución, y las correspondientes al licuado, en localidades de Arroyo Naranjo y San Miguel del Padrón

Entre fines de 1998 y septiembre del 2003, únicamente por el uso de ese combustible manufacturado, la provincia dejó de gastar 97,1 toneladas de kerosén y 14,8 de alcohol, a lo cual se le sumó el ahorro adicional de 19 500 dólares.

Al iniciarse el programa inversionista, ninguno de los clientes residenciales tenía metros contadores. El hecho de ubicar esos equipos de medición en las viviendas y el cobro en correspondencia con el consumo, estimularon el ahorro. Eso permitió, con casi la misma cantidad de gas, incorporar más de 100 mil nuevos núcleos familiares.

La instalación de las tuberías de acometida y del metro contador de cada cliente demanda una erogación promedio de 84 dólares, y las familias pagan por ese montaje 7 pesos con 30 centavos. Además, a partir de ahí los servicios de reparación de averías en las líneas y el cambio de metros contadores por desperfectos son gratuitos. En el caso del gas licuado los costos de producción resultan superiores, y también su precio de venta a las familias se halla subsidiado por el Estado. (3)

1.2 Conceptos asociados para el dominio del problema.

Por el servicio del Gas Manufacturado que se le presta a algunos hogares de Ciudad de La Habana existe una Unidad Eléctrica Básica Comercial (UEB Comercial), encargada de la comercialización del mismo, mediante un software y/o dispositivo para medir la factura de dichos servicios, los cuales se realizan a través de varios procesos como los que a continuación se explican. Estos y otros conceptos se abordan a continuación.

Sistema de Gestión:

Un sistema de gestión es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización.

UEB Comercial:

Empresa rectora del control, facturación y cobro de Gas Manufacturado en la Ciudad de la Habana.

Mi Psion:

Software encargado de descargar y cargar las lecturas de los dispositivos TPL.

TPL:

Dispositivo o aparato que lleva un lector cobrador, donde recoge y almacena las lecturas de cada metro contador de Gas de cada núcleo familiar. Terminal Procesador de Lectura.

Metro Contador:

Dispositivo que guarda el consumo de cada familia.

Casas comerciales municipales.

Empresa rectora a nivel de municipio, encargada de llevar el control y cobro de las facturaciones de las lecturas de cada núcleo familiar.

Definición de procesos dentro de la UEB Comercial.

Dentro de la UEB Comercial se realizan varias acciones para la realización de varios procesos, dichas acciones se definirán a continuación, en dependencia del proceso donde se encuentren jugando un papel para la realización del mismo. Dentro de éstas se tienen:

Facturación o Ventas de Gas Manufacturado: Se origina la acción cuando luego de haber realizado los contratos a los clientes, se le otorga su número de cliente y son procesados en el Departamento de Acceso al Sistema, para introducir sus datos y su venta se registra el no medrado de forma fija todos los meses y por la lectura mensual al cliente medrado.

Cobro del servicio: Visitar a los clientes en sus viviendas o entidades estatales para cobrar las facturaciones realizadas teniendo en cuenta las tarifas establecidas y lecturas de los metros para ingresar a la cuenta de la Empresa la venta recaudada.

Desconexión por falta de Pago: Se origina la acción cuando después de haber hecha efectiva la presentación de las facturas de cobro del cliente tanto privado como estatal no efectúa el pago del servicio en el tiempo reglamentado.

Re conexión: Se realiza luego de haber efectuado el pago y dentro del término de las 72 horas.

Ajustes a la facturación del cliente: Se origina cuando después de haberse reflejado la facturación del cliente no procede por haberse facturado indebidamente por varias causas, o cuando hay diferencias en la facturación, tanto al Mercado estatal como en el privado.

Servicios por Ventas: Son todos aquellos servicios que se prestan por solicitud del cliente de gas manufacturado que conllevan en gasto de materiales, horas de trabajo, movimiento de personal hacia y desde para garantizar la llegada del gas a la cocina del cliente etc.

1.3 Algunos Sistemas de gestión.

Resumen

Las Tecnologías de la Información y las Comunicaciones juegan un papel fundamental en la automatización de procesos industriales, y cobran cada día una mayor importancia, a medida que el uso de las mismas se incrementa. Los conceptos tales como la TIA (Automatización Totalmente Integrada), sistemas de control distribuidos, sistemas de planificación de recursos de la empresa ERP(Enterprise Resource Planning), sistemas MES(Manufacturing Execution System), entre otros, constituyen modelos y conceptos que la industria moderna no puede obviar para su eficiencia y competitividad.

Introducción a los ERP

¿Qué son los ERPs?

El ERP es una arquitectura de software que facilita el flujo de la información entre las funciones de manufactura, logística, finanzas y recursos humanos de una empresa.

Características principales de un ERP (4)

- Base de Datos centralizada.
 - Componentes del ERP interactúan entre sí consolidando todas las operaciones.
 - En un sistema ERP los datos se ingresan solo una vez. Datos deben ser consistentes, completos y comunes.
 - Las empresas deben modificar algunos de los procesos para alinearlos con los del sistema ERP.
 - Un sistema ERP incluye un conjunto de aplicaciones ERP o módulos.
 - Típicamente hay un software para cada unidad funcional.
 - Tendencia actual es ofrecer aplicaciones especializadas para determinadas industrias.
- **Integración:** El objetivo de un sistema ERP es integrar todos los procesos de la empresa, entendiéndola como una serie de áreas que se relacionan entre sí. Este enfoque permite una mayor eficiencia, reducción de tiempo y costes.

- **Modularidad:** Cada área funcional de la empresa se corresponde con un módulo del sistema de gestión. Estos módulos aunque independientes comparten información entre sí mediante una base de datos centralizada, lo que facilita la personalización y adaptabilidad por una lado, y por otro la facilidad de integración.
- **Adaptabilidad:** Gracias a la modularidad y capacidad de integración de las funcionalidades un sistema ERP es fácilmente adaptable a las necesidades de cada empresa, permitiendo una total configuración.

En la opinión personal se considera que este sistema de gestión empresarial sería de buen provecho para las empresas del país, solo se regiría por un solo sistema centralizado y como subsistemas los demás sistemas de las empresas existentes. Actualmente en la Universidad se está desarrollando un ERP que está en su fase de despliegue. La desventaja de este sistema es que los sistemas existentes a cualquier empresa, tienen que adaptarse a él, realizarle cambios al sistema de la empresa para poder implantar el ERP.

1.4 Análisis de otras soluciones existentes en Cuba y en el mundo.

Actualmente se cuenta con la presencia de una aplicación llamada **Sistema Metrado (SistMet)** en la Empresa de Gas Manufacturado que realiza la facturación y cobro de sus clientes, tanto de clientes metrados como no metrados. Además realiza reportes de los mismos y brinda los recibos de los mismos a la hora de realizar el cobro de la factura en cuestión de manera fija. La aplicación fue realizada en el lenguaje de programación Visual Basic, dicha aplicación es muy difícil de mantener y el código no cumple con los estándares de diseño y programación y con ningún paradigma que permita una alta calidad de la aplicación. Además de la poca comunicación con el autor de dicha aplicación. La aplicación presenta incoherencias en un sin número de actividades y el almacenamiento en la Base de Datos no es muy óptimo, además que no presenta ninguna organización, y no se encuentra normalizada, teniendo una redundancia de datos muy notable, haciendo los procesos de búsquedas muy tediosos y costosos en cuanto a los recursos de la computadora. Como resultado de lo antes expuesto y que todavía quedan muchas actividades se realizan manualmente, los procesos de facturación y cobro se demora varios días.

En Cuba existe un reconocido sistema llamado **VERSAT-Sarasola: Sistema cubano de Gestión Contable-Financiero**.

Sus creadores definen al **VERSAT-Sarasola** como “un paquete integrado para la gestión económica financiera que permite enviar información eficaz, de forma inmediata, desde lugares apartados, a la vez que ofrece mayor organización, control y disciplina en cada gestión”.

Se distingue por ser el primer sistema de contabilidad cubano certificado, según las nuevas normativas establecidas por los Ministerios de Finanzas y Precios y de la Informática y las Comunicaciones, para este tipo de Software.

¿Qué beneficios aporta?

Al evitarse la importación del sistema foráneo, el país se ahorró un millón 186 mil dólares que costaban las licencias, más la labor de consultoría, trabajo técnico, atenciones y otros gastos.

Entre otras ventajas, permitió la elaboración de un software similar o superior al que se pensaba adquirir, y además se le agregaron nuevas tecnologías. (5)

Desde hace ya varios años, se ha dado mayor importancia a las Tecnologías de Información y su alineación con las estrategias del negocio para mejorar sus procesos clave de negocio. Prueba de ello, es el incremento tan sustancial de adquisiciones de paquetes de software empresariales tales como el ERP (Enterprise Resource Planning), con el cual los directivos de las compañías esperan tener integradas todas las áreas o departamentos de la compañía que apoyan para la generación de sus productos y servicios.

En el mundo por el gran auge y la importancia que tiene el manejo de la información de clientes de sus gastos en diversos servicios, se han realizado varias aplicaciones no solo para pequeñas empresas sino para grandes transnacionales donde no solo el flujo de información es inmenso sino el manejo del flujo monetario. Dentro de estos tenemos:

Express Invoice 2.03

Express Invoice ofrece todo lo que se necesita para controlar los gastos, cobros, hacer un seguimiento de un inventario, clientes y proveedores, además de encargarse de todo el tema de la facturación y generar informes detallados.

Ideal tanto para pequeñas y medianas empresas como autónomas, Express Invoice es una aplicación muy profesional que permite mantener una base de datos de todos los clientes y proveedores, crear una factura y enviarla por correo electrónico o FAX desde la aplicación, programar incluso el envío de facturas y controlarlo desde la web.

Esto significa que Express Invoice integra un servidor web que, una vez activado, permite conectar a los datos desde cualquier navegador web. Otro dato interesante son los informes generados. En muy pocos clics se sabrá cuáles son las facturas impagadas, por ejemplo. También puede generar informes PDF de ventas, cobros, etc. (5)

ZFactura 1.1

Concebida para todas aquellas empresas de servicios que no necesitan de una gestión comercial completa, preferentemente autónoma y profesional independientes, ZFactura facilita la elaboración de facturas de forma eficaz. Entre los apartados que permite gestionar están los productos, clientes, proveedores, gastos, facturas de compra e ingresos. Gracias a su potente editor de informes, se puede configurar el aspecto de las facturas al gusto del cliente. Eso sí, tanta potencia se paga y es que el diseño puede ser tan exhaustivo como poco intuitivo. Más allá de la estética, entre sus funciones más útiles destaca la posibilidad de obtener informes sobre tus ingresos y gastos facilitándole el beneficio obtenido. También puede calcular informes básicos como listados de productos, clientes, etc.

Las distintas secciones interactúan bien unas con otras, por lo que en pocos clics se tiene la posibilidad de completar varios apartados. Quizá sería interesante que el programa avisara en el caso de que algún campo importante se quedara sin rellenar. Lo más reseñable del conjunto es que ZFactura ofrece lo que promete, que no es poco. (6)

Pagarés y Gestión de Cobros Pro 13.0

Pagarés y Gestión de Cobros es una sencilla herramienta con la que se puede emitir pagarés, tanto bancarios como de empresa, con su carta comercial correspondiente.

Se puede cruzar, sellar e incluso firmar, en un entorno de trabajo altamente configurable, y con un módulo específico que permite hacer un seguimiento de los cobros, anotar todas las gestiones realizadas, emitir cartas de reclamación, generar todo tipo de informes y gestionar avales bancarios. (7)

En la opinión personal es necesario destacar que ninguno de los sistemas expuestos anteriormente excepto el ERP, no cumplen con los requisitos que solicita un cliente como la Empresa de Gas Manufacturado, porque lo que se desea facturar y/o cobrar es el servicio de Gas Manufacturado a cada uno de los hogares que se encuentran en Ciudad de La Habana, y estos sistemas no realizan dicha facturación ni el cobro. El ERP si por poseer varios módulos que guardan relación con todo tipo de empresas, principalmente con el que se está desarrollando en la Universidad de las Ciencias Informáticas.

CONCLUSIONES

En este capítulo se abordaron los conceptos para el dominio del problema en cuestión, el uso de las Tecnologías de la Investigación y Comunicaciones, no solo en el mundo sino en el país. Se realizó un estudio del proceso de gas manufacturado el cual se ha convertido en un servicio para la población. Tareas por las cuales se fue desarrollando dicha investigación y la presentación de algunos sistemas automatizados existentes, no solo a nivel mundial sino nacional, de los cuales se esperan alguna que otra mejoría para el desarrollo de las empresas. Se abordaron los distintos de métodos empíricos para la mejor obtención de información con el personal que labora en la empresa.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

2.1 Análisis de las Metodologías para el Desarrollo de Software.

Una aplicación Web es un conjunto de páginas Web estáticas y dinámicas. Una *página Web estática* es aquella que no cambia cuando un usuario la solicita: el servidor Web envía la página al navegador Web solicitante sin modificarla. Por el contrario, el servidor modifica las *páginas Web dinámicas* antes de enviarlas al navegador solicitante. La naturaleza cambiante de este tipo de página es la que le da el nombre de dinámica. (9)

Para un mejor entendimiento entre diseñadores de aplicaciones, desarrolladores, y personas referentes a la carrera de Informática, surge la necesidad de hacer un lenguaje que se entendiera a nivel mundial: el Unified Model Language (UML).

El éxito de los proyectos de desarrollo de aplicaciones o sistemas se debe a que sirven como enlace entre quien tiene la idea y el desarrollador. El UML es una herramienta que cumple con esta función, ya que le ayuda a capturar la idea de un sistema para comunicarla posteriormente a quién esté involucrado en un proceso de desarrollo; esto se lleva a cabo mediante un conjunto de símbolos y diagramas. Cada diagrama tiene fines distintos dentro del proceso de desarrollo. (9)

La metodología en el desarrollo de un Software es el plano de apoyo y el conjunto de métodos que se deben utilizar para desarrollar un producto con calidad, al hacer la selección de la más adecuada se tendrán clientes complacidos y desarrolladores satisfechos. Servirá para que la confección y desarrollo del Software sea menos complicado, riguroso y fácil de controlar.

Con el lenguaje de modelado UML permitirá un mayor entendimiento a no solo los desarrolladores sino también a los que tengan al menos un conocimiento de este lenguaje y sus diagramas que a continuación se harán referencia bajo la metodología de desarrollo de Software que se haya escogido.

En un proyecto de desarrollo de Software la metodología define ¿Quién debe hacer? ¿Qué?, ¿Cuándo y Cómo debe hacerlo? No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable.

Lenguaje Unificado de Modelado (UML)

¿QUÉ ES UML?

UML es un conjunto de herramientas, que permite modelar (analizar y diseñar) sistemas orientados a objetos. (10)

UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y unas reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema.

Este lenguaje indica cómo crear y leer los modelos, pero no dice cómo crearlos. Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones: (11)

- **Visualizar:**

UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.

- **Especificar:**

UML permite especificar cuáles son las características de un sistema antes de su construcción.

- **Construir:**

A partir de los modelos especificados se pueden construir los sistemas diseñados.

- **Documentar:**

Los propios elementos gráficos sirven como documentación del sistema desarrollado que puedes servir para su futura revisión.

Un modelo UML está compuesto por tres clases de bloques de construcción:

- **Elementos:** Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.)
- **Relaciones:** relacionan los elementos entre sí.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

- **Diagramas:** Son colecciones de elementos con sus relaciones.

Diagramas UML

Un diagrama es la representación gráfica de un conjunto de elementos con sus relaciones. En concreto, un diagrama ofrece una vista del sistema a modelar. Para poder representar correctamente un sistema, UML ofrece una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas. UML incluye los siguientes diagramas:

- Diagrama de casos de uso.
- Diagrama de clases.
- Diagrama de objetos.
- Diagrama de secuencia.
- Diagrama de colaboración.
- Diagrama de estados.
- Diagrama de actividades.
- Diagrama de componentes.
- Diagrama de despliegue.

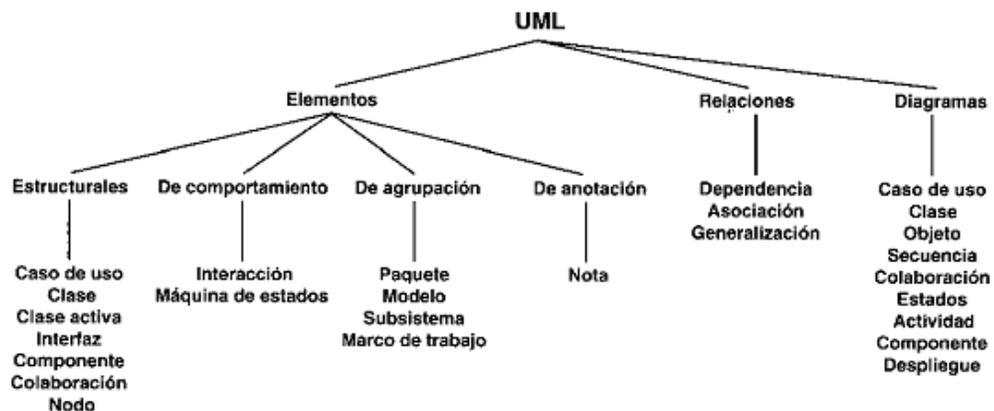


Figura 1 El vocabulario de UML (Jacobson, Booch, Rumbaugh).

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

Los diagramas más interesantes (y los más usados) son los de casos de uso, clases y secuencia.

El **diagrama de casos de usos** representa gráficamente los casos de uso que tiene un sistema. Se define un caso de uso como cada interacción supuesta con el sistema a desarrollar, donde se representan los requisitos funcionales. Es decir, se está diciendo lo que tiene que hacer un sistema y cómo.

El **diagrama de clases** muestra un conjunto de clases, interfaces y sus relaciones. Éste es el diagrama más común a la hora de describir el diseño de los sistemas orientados a objetos.

En el **diagrama de secuencia** se muestra la interacción de los objetos que componen un sistema de forma temporal.

El resto de diagramas muestran distintos aspectos del sistema a modelar. Para modelar el comportamiento dinámico del sistema están los de **interacción, colaboración, estados y actividades**. **Los diagramas de componentes y despliegue** están enfocados a la implementación del sistema.

2.2 Comparación entre las metodologías

Existen dos tipos de Metodologías las Ágiles y las Pesadas o Robustas, de las cuales se hará referencia dentro de breve. Así como cuáles son algunas de ellas y algunas características.

Metodologías Ágiles	Metodologías Pesadas
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el Proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Tabla 2.1 Comparación entre las metodologías (13)

Las metodologías ágiles (como por ejemplo XP, SCRUM, DSDM, Crystal, etc.) forman parte del movimiento de desarrollo ágil de software, que se basan en la adaptabilidad de cualquier cambio como medio para aumentar las posibilidades de éxito de un proyecto.

2.3 Metodologías de Software

Metodología Extreme Programming (XP)

La Programación Extrema (PX), mejor conocida por su nombre en inglés Extreme Programming (XP), es una de las llamadas Metodologías Ágiles de desarrollo de software más exitosas de los tiempos recientes. Cada día se genera incontable información sobre el tema, generalmente en inglés. (14)

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

Algunas características de XP

1. Pruebas unitarias

La producción de código está dirigida por las pruebas unitarias. Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. Los clientes escriben las pruebas funcionales para cada historia de usuario que deba validarse. En este contexto de desarrollo evolutivo y de énfasis en pruebas constantes, la automatización para apoyar esta actividad es crucial. (15)

2. Refactorización (Refactoring)

Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. La refactorización mejora la estructura interna del código sin alterar su comportamiento externo. (15)

3. Programación en pares

Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Mientras uno teclea el otro va revisando el código.

Toda la producción de código debe realizarse con trabajo en parejas de programadores. Según Cockburn y Williams en un estudio realizado para identificar los costos y beneficios de la programación en parejas. (15)

Metodología SCRUM

SCRUM es un Proceso Ágil que sirve para administrar y controlar el desarrollo de Software. El desarrollo se realiza en forma iterativa e incremental. Cada ciclo o iteración termina con una pieza de Software ejecutable que incorpora nueva funcionalidad. Las iteraciones en general tienen una duración entre 2 y 4 semanas.

Está diseñado especialmente para adaptarse a los cambios en los requerimientos, éstos y las prioridades se revisan y ajustan durante el proyecto en intervalos muy cortos y regulares. De esta forma se puede adaptar en tiempo real el producto que se está construyendo a las necesidades del cliente.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

Una de las características más importantes es que es muy fácil de explicar y de entender, lo que ayuda mucho a su implantación. (16)

En Scrum, el equipo se focaliza en construir Software de calidad, definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en remover cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo. Se busca que los equipos sean lo más efectivos y productivos posible. (José H. Canós, 2003)

La transparencia se refiere a que aquellos aspectos del proceso que afectan al resultado, deben ser visibles y conocidos por aquellos que controlan el proceso.

Los roles del SCRUM (17)

- **El dueño del producto:** Determina las prioridades.
- **El Scrum Manager:** Gestiona y facilita la ejecución del proceso.
- **El Team:** Construye el producto.
- **El ScrumMaster:** Asesora y observa.

Metodología Proceso Unificado de Desarrollo (RUP)

El RUP es un proceso bastante estandarizado pero además es un proceso bastante flexible, al cual se le pueden hacer determinadas variaciones en dependencia del tipo de aplicación en las que se vaya a trabajar y es ahí donde ha estado la ganancia de por qué se ha convertido en un gran estándar, no solamente para la producción de softwares informáticos sino también que ha sido adaptado a otros tipos de procesos industriales en los cuales ha dado muy buenos resultados su utilización.

La metodología en cuestión es una serie de ciclos que constituyen la vida de un sistema. El ciclo de vida de RUP se caracteriza por ser: (9)

❖ **Dirigido por Casos de Uso:**

Los casos de uso reflejan lo que los usuarios desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí, guían el proceso de desarrollo, pues los

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

modelos que se obtienen como resultado de los diferentes flujos de trabajo representan la realización de los casos de uso.

❖ Centrado en la Arquitectura

La arquitectura muestra la visión común del sistema completo, en la que el equipo de proyecto y los usuarios deben estar de acuerdo. Describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. Las iteraciones que se realizan en RUP comienzan por los casos de uso relevantes desde el punto de vista de la arquitectura.

❖ Iterativo e Incremental

Cada fase de RUP se desarrolla en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de Elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración.

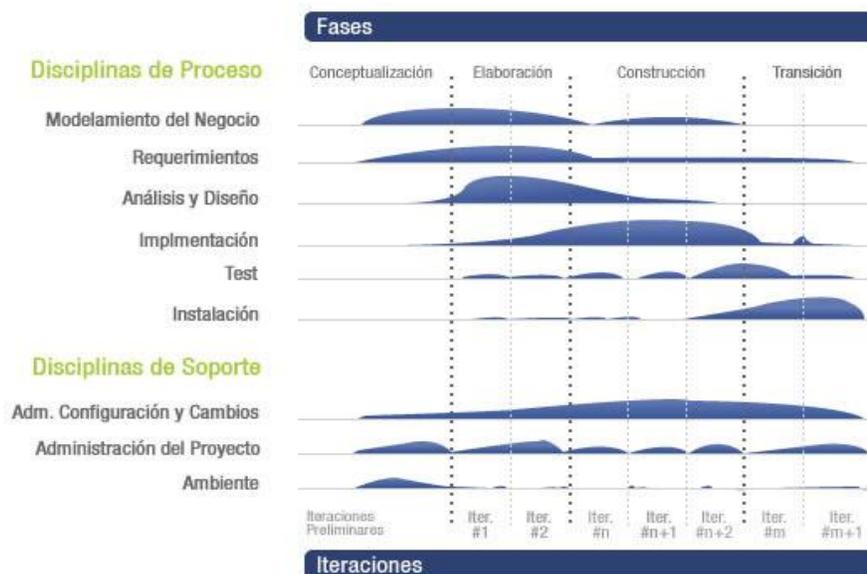


Figura 2. Fases e iteraciones de la Metodología RUP

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

RUP describe la manera de implementar efectivamente "mejores prácticas" comercialmente probadas para el desarrollo de software. RUP ofrece a cada miembro del equipo las guías, plantillas y herramientas necesarias para aprovechar al máximo las siguientes mejores prácticas:

1. Desarrolle el software iterativamente.
2. Administre Requerimientos.
3. Utilice arquitecturas basadas en componentes.
4. Modele el software visualmente.
5. Verifique la calidad del software.
6. Controle los cambios al software.

Herramientas CASE (Computer-Aided Software Engineering)

Hoy en día, muchas empresas se han extendido a la adquisición de herramientas CASE (Ingeniería Asistida por Computadora), con el fin de automatizar los aspectos clave de todo el proceso de desarrollo de un sistema, desde el principio hasta el final e incrementar su posición en el mercado competitivo, pero obteniendo algunas veces elevados costos en la adquisición de la herramienta y costos de entrenamiento de personal así como la falta de adaptación de la herramienta a la arquitectura de la información y a las metodologías de desarrollo utilizadas por la organización. Algunas herramientas CASE no ofrecen o evalúan soluciones potenciales para los problemas relacionados con sistemas o virtualmente no llevan a cabo ningún análisis de los requerimientos de la aplicación.

Sin embargo, CASE proporciona un conjunto de herramientas semiautomatizadas y automatizadas que están desarrollando una cultura de ingeniería nueva para muchas empresas. Uno de los objetivos más importantes del CASE (a largo plazo) es conseguir la generación automática de programas desde una especificación a nivel de diseño.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

Un sistema CASE es una herramienta informática que permite a los desarrolladores de software concentrarse en las tareas que les son propias, aumentando su productividad y mejorando la calidad de su trabajo. El objetivo máximo que persiguen es la generación automática de programas de ordenador a partir de las especificaciones de diseño. (18)

Las herramientas CASE de modelado con UML permiten aplicar una metodología de análisis y diseño orientados a objetos y hacer una abstracción del código fuente, en un nivel donde la arquitectura y el diseño se tornan más obvios y fáciles de entender y modificar. Cuanto más grande es un proyecto, es más aconsejable utilizar este tipo de herramientas. La herramienta CASE utilizada por el proyecto, debe ser capaz de ayudar al equipo en la mayor parte de las etapas, y posibilitar que las salidas o resultados obtenidos en cada etapa, complementen o sean la entrada de la etapa siguiente. Mediante el uso de las herramientas CASE, los analistas de sistemas pueden capturar los requisitos del sistema con un modelo de casos de uso. De igual forma, los diseñadores y arquitectos pueden producir el modelo de diseño y los desarrolladores pueden transformar los modelos en una aplicación funcional.

- Las herramientas CASE se han desarrollado como una de las soluciones para afrontar los problemas de una **calidad de software** pobre y una **documentación inadecuada**.
- CASE es la automatización de métodos existentes de Ingeniería del Software con el objetivo de mejorar la calidad del producto junto con la eficiencia de los desarrolladores de software.
- La idea básica CASE es soportar cada etapa del ciclo de vida con un conjunto de herramientas que ahorren esfuerzos
- Algunas de las herramientas CASE se concentran en soportar las primeras etapas del ciclo de vida. Para ello, dan una asistencia automatizada para el **dibujo automático de diagramas**, **toolbox para dibujar en pantalla**, y **chequeo de correcciones**. (19)

Dentro de algunas Herramientas CASE tenemos al Visual Paradigm, Rational Rose, Enterprise Architect.

Visual Paradigm

Visual Paradigm para UML es una herramienta UML profesional muy potente que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño Orientados a Objetos, construcción, pruebas y despliegue. Diseñado para varios tipos de usuarios, incluyendo Ingenieros de Software, Analistas de Sistemas, Analistas de Negocio y Arquitectos de Sistema. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Esta herramienta soporta hasta la fecha UML 2.1 completo y Notación para el Modelado de Procesos de Negocio (de ahora en adelante BPMN) permite realizar ingeniería tanto directa como inversa.

- BPMN define diagramas de procesos de negocios basados en la técnica de diagramas de flujo, adaptados para graficar las operaciones de los procesos de la organización
- Se compone de un conjunto de elementos gráficos que facilitan un diagrama entendible tanto por audiencias de negocios como técnicas.

A partir de un modelo relacional en SQL Server, MySql, etc. Es capaz de desplegar todas las clases asociadas a las tablas (siguiendo el patrón de diseño Una Clase-Una Tabla). Para gestionar la persistencia y el mapeo de estas clases con la base de datos utiliza Hibernate para Java y NHibernate en el caso de un proyecto .Net.

Es colaborativa, soporta múltiples usuarios trabajando sobre el mismo proyecto; genera la documentación del proyecto automáticamente en varios formatos como Web o .pdf, y permite control de versiones.

Es posible generar código desde Visual Paradigm para plataformas como .NET, Java y PHP, así como obtener diagramas a partir del código, esto es de gran utilidad pues ahorra tiempo a los desarrolladores y reduce las posibilidades de cometer errores. Brinda la posibilidad de obtener una Base de Datos relacional y el código necesario para acceder a ésta a partir del diagrama de un Diagrama Entidad Relación, además se conecta fácilmente a varios servidores de base de datos. Se integra con varios ambientes de desarrollo integrados (IDE) lo cual permite pasar del código al modelado y viceversa. Establece interoperabilidad con otras aplicaciones como el Visio y el Rational Rose y documentar todo el trabajo y especificaciones de CU sin necesidad de utilizar herramientas externas, por ejemplo editores de texto, utilizando plantillas que se

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

encuentran o que pueden ser creadas por los usuarios. Disponible en múltiples lenguajes y plataformas: Microsoft Windows (98, 2000, XP, o Vista) Linux, Mac OS X, Solaris o Java.

Visual Paradigm para UML es un producto que facilita a las organizaciones mediante los diagramas de diseño visual, integrar y desplegar sus aplicaciones empresariales de misión crítica y sus bases de datos. La herramienta le ayuda a su equipo de desarrollo de software a realizar un modelo de excelencia, construcción, despliegue de software en el proceso de desarrollo, y acelerar al máximo tanto al equipo como a las contribuciones individuales.

Visual Paradigm para UML soporta un conjunto de lenguajes, tanto en la generación de código y de ingeniería inversa en Java, C + +, CORBA IDL, PHP, XML Schema, Ada y Python. Además, apoya la generación de código C #, VB. NET, Objeto Definition Language (ODL), Flash ActionScript, Delphi, Perl, Objective-C, y Ruby.

Rational Rose Enterprise

Esta herramienta CASE creada por los ingenieros (Booch, Rumbaugh y Jacobson) es de fácil utilización. Permite una modelación absoluta de los procesos del negocio y del sistema, además resulta de gran utilidad para los desarrolladores de proyectos, pues ella cubre todo el ciclo de vida del proyecto desde su fase inicial hasta su culminación, esta herramienta posibilita establecer una trazabilidad entre los modelos (análisis y diseño) y el código ejecutable, también permite el desarrollo de software en equipo basado en metodología como RUP, cada rol tiene su propia vista de arquitectura (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue), pero utilizan un lenguaje común para comprender y comunicar la estructura y funcionalidad del sistema en construcción. Cada analista, desarrollador o diseñador puede usar Rational Rose para definir y comunicar el negocio, el diseño y la arquitectura de la aplicación que se está desarrollando. Es una completa solución para mostrar de forma gráfica el análisis de los procesos del negocio y los requerimientos del sistema.

Incluye Unified Modeling Language (UML [™]) de apoyo y es uno de los más completos productos de la familia Rational Rose. Así como apoya el Análisis, ANSI C ++, Rosa J y Visual C++, patrones, Enterprise JavaBeans [™] 2.0. Permite la ingeniería inversa de algunas de las más comunes de Java 1.5 construye

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

ofertas de código y análisis de la calidad de las capacidades de generación de código, con configurables modelo a las capacidades de sincronización de código, así como la gestión más granular y el uso de modelos por separado con la función de controlar los componentes del modelo. Se integra con IBM Rational otras herramientas de desarrollo de ciclo de vida, así como cualquier SCC compatible con sistema de control de versiones, como IBM Rational ClearCase ®. Incluye un sitio modelado Add-In, que proporciona visualización, modelado y herramientas para el desarrollo de aplicaciones Web. Proporciona modelado UML para el diseño de bases de datos, con capacidad para representar a la integración de los datos y requisitos de las aplicaciones a través de diseños lógicos y físicos. Crea las definiciones de tipo de documento XML (DTD) para el uso en su aplicación y el único Sistema Operativo compatible es Windows.

Fundamentación de la metodología, lenguaje y herramientas de modelado seleccionadas para el desarrollo.

Elegir una metodología no es una cuestión simple. Depende principalmente de dos factores, el tipo de proyectos y la cultura que exista en la empresa o en la universidad.

El primer paso es conocer a fondo las metodologías evaluadas o buscar a alguien que las conozca, y en una situación ideal haber trabajado con varias de ellas. No hay metodología que funcione de manera universal, de hecho, cada vez más es necesario ajustarse para cada organización y tipo de proyecto. Realizar este ajuste es algo que necesita de una experiencia y un conocimiento previo.

Luego de la realización de un estudio de las metodologías de desarrollo más utilizadas en la actualidad se llegó a la conclusión que la metodología idónea a seguir es RUP para el desarrollo del sistema que se desea implementar, es la más conocida de las metodologías por su auge a nivel mundial, donde hay personas capacitadas en la misma. Un proceso iterativo e incremental flexible a la hora de dividir el trabajo en varias fases teniendo bien definidas las tareas a realizar en cada una de éstas, además de proponer para cada flujo de trabajo artefactos para organizar el Proceso de Desarrollo de Software.

Se ha decidido utilizar el lenguaje de modelado UML debido a que está consolidado como el lenguaje estándar en el análisis y diseño de sistemas informáticos. Sus características visuales facilitan a los

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

integrantes de un equipo multidisciplinario, ya sean analistas, diseñadores, programadores o cualquier otro rol, una fácil comunicación e intercomunicación que tributa a la común comprensión de lo que se quiere hacer. A través de él se logran robustas aplicaciones, bien diseñadas y de fácil mantenimiento como resultados de ser desarrolladas también bajo el paradigma orientado a objetos y la metodología de desarrollo RUP. Así como en el desarrollo han participado investigadores de reconocido prestigio. Ha sido apoyado por prácticamente todas las empresas importantes de informática. Se ha aceptado como un estándar por el Grupo de Gestión o Administración de Objetos (de ahora en adelante OMG).

Prácticamente todas las herramientas CASE y de desarrollo lo han adoptado como lenguaje de modelado.

El patrón que se va a utilizar es el Modelo-Vista-Controlador (MVC) por el estudio del rol de arquitecto en el proyecto así como el estudio del mismo dentro del proyecto y la experiencia dentro del equipo de desarrollo.

También como herramienta de modelado se seleccionó el Visual Paradigm por su intuitivo modelado, además de ser una herramienta multiplataforma y contar con amplias funcionalidades que satisfacen las necesidades de los desarrolladores. Así como que está hecho para BPMN y a pesar de ser software propietario la UCI paga la licencia. Con el Visual Paradigm se agiliza el modelado de software, aumentando la productividad y viabiliza el trabajo en equipo. Así como su compatibilidad con diversos lenguajes de programación, contiene más plugins y así como el soporte de subdiagramas para todos los modelos UML, provee pleno soporte para ingeniería directa e inversa, soporta aplicaciones web y realiza el diseño centrado en Casos de Uso y enfocado al Negocio que genera un Software de mayor calidad. Está hecho para BPMN, contiene muchos más estereotipos y enfocado a procesos. Es más rico en modelación.

Análisis y Diseño

El objetivo principal de esta disciplina es transformar los requerimientos a una especificación que describa cómo implementar el sistema. El análisis fundamentalmente consiste en obtener una visión que se preocupa de ver que hace el sistema de software a desarrollar, por tal motivo este se interesa en los requerimientos funcionales. Por otro lado, el diseño es un refinamiento que toma en cuenta los requerimientos no funcionales, por lo cual se centra en cómo el sistema cumple sus objetivos.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

Los objetivos específicos del análisis y diseño son:

1. Transformar los requerimientos al diseño del futuro sistema.
2. Desarrollar una arquitectura para el sistema.
3. Adaptar el diseño para que sea consistente con el entorno de implementación.

Al principio de la fase de elaboración hay que definir una arquitectura candidata: crear un esquema inicial de la arquitectura del sistema, identificar clases de análisis y actualizar las realizaciones de los Casos de Uso con las interacciones de las clases de análisis. Durante la fase de elaboración se va refinando esta arquitectura hasta llegar a su forma definitiva. En cada iteración hay que analizar el comportamiento para diseñar componentes. (20)

Propósitos generales del análisis y el diseño:

- Transformar los requerimientos en un diseño de lo que será el sistema.
- Evolucionar una arquitectura robusta para el sistema.
- Adaptar el diseño al ambiente de implementación, diseñándolo para el desarrollo.

Tabla 2.3 Propósitos particulares del análisis y diseño. (Jacobson, y otros).

Propósitos del análisis	Propósitos de diseño
Comprensión perfecta de los requisitos, refinarlos y estructurarlos.	Comprensión exacta de aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario.
Análisis profundo de los requisitos funcionales utilizando el lenguaje de los desarrolladores.	Brindar una entrada apropiada para actividades de implementación ya sean requisitos o subsistemas

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

	individuales interfaces y clases.
--	-----------------------------------

Rol de Analista

El analista dirige y coordina la etapa de elicitación de requerimientos y modelación de casos de uso de manera que va perfilando las funcionalidades del sistema y delimitando el mismo. Por ejemplo él delimita quiénes son los actores y cuáles son los casos de uso que propician la interacción con el sistema. (Rational Software Corporation, 2003)

La persona que trabaje como el rol de analista debe ser experta identificando y entendiendo problemas y oportunidades. Ha de contar con habilidades para ofrecer soluciones a las necesidades del problema. Debe ser muy comunicativa. El conocimiento del negocio y el dominio de tecnologías han de ser otras habilidades importantes en este rol, que vienen a perder un poco de importancia si es un profesional que entiende y absorbe las nuevas informaciones rápidamente. Algo que no debe de faltar es que debe ser capaz de colaborar eficazmente con los otros integrantes del equipo.

Rol de Diseñador

El diseñador es el responsable de diseñar partes del sistema, teniendo en cuenta las restricciones de los requerimientos, la arquitectura y el proceso de desarrollo para el proyecto. Él identifica y define las responsabilidades, funcionamientos, atributos, y relaciones de los elementos del diseño. Asegura que el diseño es consistente con la arquitectura del software, y la detalla a un punto donde la aplicación puede proceder. (Rational Software Corporation, 2003) La persona que desempeñe el rol de diseñador debe contar con sólidos conocimientos en cuanto a los requisitos del sistema, la arquitectura del mismo, las técnicas de diseño de software, incluyendo el análisis orientado objeto y el lenguaje unificado de modelado, las tecnologías en las que el sistema será implementado y las líneas bases del proyecto.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

CONCLUSIONES

En este capítulo se describieron algunas herramientas existentes, se hizo un análisis de las mismas y lo que ayudó a la mejor realización de la solución del problema. Se expusieron algunas metodologías de desarrollo de software y de herramientas CASE. Así como la definición del análisis y el diseño y roles de analista y diseñador para un mejor entendimiento a personas ajenas a la carrera.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Introducción

En el presente capítulo se describe la solución propuesta. Principalmente se basará en el modelo de negocio a través de los casos de uso del negocio así como los conceptos que se manejan en el negocio, ya sean actores o trabajadores pertenecientes al entorno. Esto posibilita el establecimiento de un vocabulario común para los involucrados, que va en pos de un mejor entendimiento del proceso en cuestión. Se describirán los requisitos funcionales y no funcionales que ha de satisfacer el sistema. Se representará la modelación en términos de casos de usos del mismo. Además, se detallará cada caso de uso mediante sus especificaciones y se muestran algunos diagramas de las clases del análisis y el diseño del sistema en cuestión.

3.1 Descripción de los procesos del negocio.

En la Empresa de Gas Manufacturado se realizan los servicios de Facturación y Cobro para algunos núcleos familiares de Ciudad de La Habana donde se benefician con dichos servicios los clientes que habitan en dichos hogares.

El proceso de Facturación es el proceso mediante el cual se calcula cuánto tiene que pagar un cliente de gas manufacturado de acuerdo al consumo del mismo en un período de tiempo determinado que es alrededor de un mes. Existe para ello una forma de pago establecida que depende del tipo de cliente así como de las circunstancias actuales con que cuenta la vivienda, ejemplo para los clientes con metro contador la factura depende del consumo del gas en el período de tiempo que se esté midiendo, y para los clientes sin metro contador, existe una tarifa fija y la factura para esos clientes se calcula de acuerdo a la cantidad de personas registradas en ese hogar.

Mientras que el proceso de Cobro es el proceso mediante el cual se le realiza el cobro a los distintos tipos de clientes ya sean clientes metrados o clientes no metrados según el consumo de gas manufacturado que es en metros cúbicos. Donde el responsable de realizar dicho cobro es el lector cobrador o la cajera de la Casa Comercial. El encargado de procesar todo el efectivo que se ha recaudado mediante la información de cobro procesada es el controlador económico de la Casa Comercial. El lector cobrador va

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

a cada vivienda a realizar el cobro del consumo mensual del gas manufacturado o de lo contrario el cliente va a la Casa Comercial y realiza el pago con la cajera de dicha Casa. Este proceso concluye con el cierre del cobro realizado por el Analista del Sistema y/o la tecleadora del Sistema cada mes.

3.2 Información que se maneja.

El Sistema de Facturación y Cobro debe interactuar con un sistema implantado actualmente en la Empresa de Gas Manufacturado que tiene estrecha relación con los procesos del negocio. Se debe mencionar el SistMet, cuya base de datos almacena la información de las facturas, lecturas y la información de los contratos. El nuevo sistema se nutrirá de la información que se almacena en la base de datos de SistMet.

Otras fuentes de información utilizadas por procesos de negocio son documentos en formato Microsoft Excel, por lo que el sistema propuesto debe de trabajar con este estándar tanto para lectura como para escritura. Se exportará a formato pdf y Excel utilizando librerías existentes de PHP.

Los datos del sistema se manejan a través del protocolo HTTPS, el cifrado de datos se gestionará a nivel del servidor web utilizando SSL.

El servidor web y base de datos del sistema se encontrarán en la casa matriz comercial de la empresa de Gas Manufacturado, teniendo en cuenta que las fuentes de datos son las casas comerciales y estas se encuentran en diferentes municipios de la capital cubana se puede suponer que la distribución es bastante amplia.

3.3 Modelo de Negocio.

Actores del Negocio.

Actor del Negocio:

- Un actor del negocio es cualquier individuo, grupo, organización o máquina que interactúa con el negocio.
- Un actor siempre permanece fuera de las fronteras del negocio que se está investigando.
- El actor debe expresar un rol, no un usuario o persona específica.
- Un mismo usuario puede actuar como diferentes actores (Roles)
- El actor siempre se involucra al menos con un CU.

Actor	Descripción
Cliente	Persona a la cual se le contrata el servicio de Gas Manufacturado.
Lector Cobrador	Persona jurídica responsabilizada con la visita al cliente para realizar el cobro o lectura del metro contador, así como incidencias sobre la calidad del servicio de gas.
Tramitadora de Documento de la Casa Comercial	Persona designada para realizar diferentes tipos de acciones concretas dentro de la actividad comercial.
Jefe del Departamento de Sistema	Es la persona encargada de dirigir y controlar todo el proceso que se realiza en el Departamento de Sistema de la UEB Comercial. Se encarga de informar a las casas comerciales que el cierre de mes está listo, y de esta forma ordena que a partir de ese momento deba comenzar todo el proceso de lectura.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Departamento. de Ventas y Facturación	El Departamento de Ventas y Facturación de la UEB Comercial es el encargado de trabajar con los efectivos y toda la parte económica al efectuar el cobro a los clientes.
Jefe del Departamento. de Venta y Facturación	Es la persona encargada de dirigir y controlar todo el proceso que se realiza en el Departamento de Ventas y Facturación, así como los recursos humanos en que laboran.
Cajera	Este actor está relacionado con el cobro del dinero de cada cliente que paga el servicio en la casa comercial directamente, o sea, es el cliente que no paga en el terreno y va a efectuar el pago a la unidad.
Jefe Comercial	Es la persona encargada de toda la gestión y la comunicación de la información que se realiza con la UEB Comercial.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Trabajadores del Negocio.

Trabajador del Negocio:

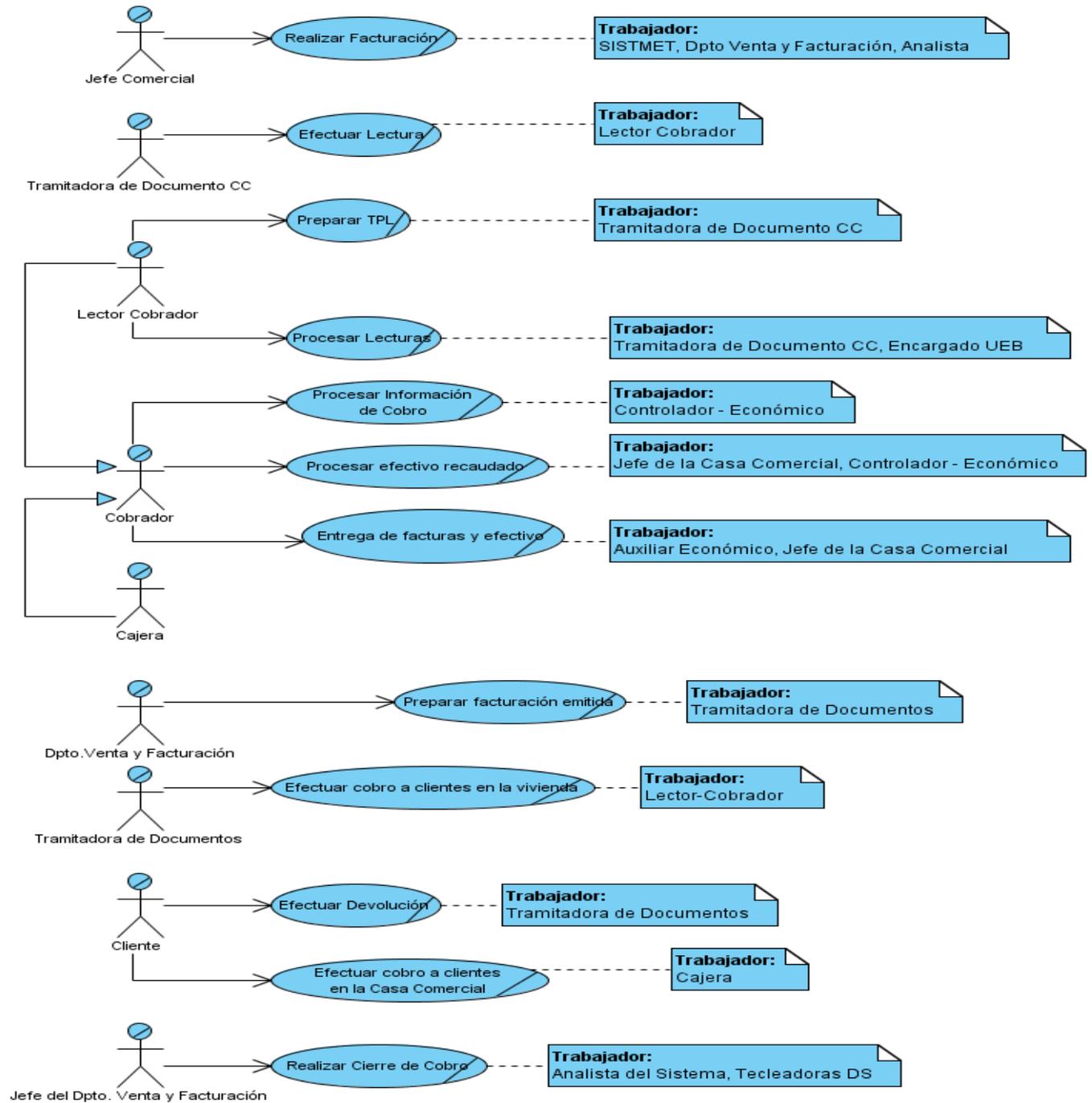
- Representa a personas, o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso.
- Los trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema que se quiere construir.

Trabajador	Descripción
Tramitadora de Documentos CC	La tramitadora es la encargada de cargar y descargar la información de los TPL (Dispositivo Electrónico) del lector cobrador en el sistema que está instalado en cada Casa Comercial. Es la encargada de corregir cualquier error que se cometa en el proceso de lectura. Es la responsable de que toda la información referente a las lecturas por TPL lleguen al departamento de facturación lo más depuradas posible, con el mínimo margen de error, y con calidad. Además recibe la información enviada por la empresa y emite los listados de cobro que serán entregados a los cobradores.
Lector Cobrador	Es el responsable de efectuar el proceso de lectura, reflejando correctamente las claves de incidencias y las lecturas de consumo del metro contador de Gas de cada cliente/contrato. Es el encargado de visitar al cliente y llevarle su factura para efectuar el cobro.
Analista de sistema	El Analista es el encargado de recibir la información procedente de las casas comerciales en el período de tiempo estipulado, revisarla y realizar el procesamiento de la misma.
SISTMET	Aplicación Informática que se utiliza para procesar,

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	rectificar y consolidar los datos, para crear los talonarios y enviarlos a la tramitadora.
Departamento. Venta y Facturación	Es el departamento donde se efectúan todas las acciones económicas de la parte de venta.
Auxiliar Económico	Es el encargado de la gestión económica de la casa comercial. Es el encargado de recibir la información que trae el lector cobrador después de haber cobrado en las viviendas y de ahí en adelante efectuar los cuadros de la información.
Controlador Económico	Es el encargado de recibir la información que trae el cobrador después de haber cobrado en las viviendas y de ahí en adelante efectuar los cuadros de la información.

3.4 Diagrama de Casos de Uso del Negocio



CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

3.5 Realización de los Casos de Uso del Negocio.

Los Casos de Uso del Negocio consisten en la descripción de las secuencias de actividades que, en conjunto, producen algo para el actor del negocio. El proceso (workflow) consiste en un flujo básico de una o más alternativas de flujos. La estructura del flujo se describe gráficamente con la ayuda de un diagrama de actividades.

3.5.1 Realizar Facturación

Caso de uso del negocio	Realizar Facturación
Actores	Jefe Comercial
Resumen	El caso de uso inicia cuando se recibe en el Departamento de Sistema la información de las lecturas de la Casa Comercial, se procede entonces a calcular el consumo para cada cliente, en dependencia de la clave de lectura de cada uno, se factura cada cliente y se imprimen los talonarios, así como el listado de recibo.
Acción del actor	Respuesta del proceso de negocio
1. El Jefe Comercial de cada Casa Comercial envía un fichero “.dbf” que contiene las lecturas y el consumo de cada uno de sus clientes.	2. El analista del Departamento de Facturación descarga el fichero enviado.
	3. El analista del Departamento de Facturación Importa el fichero a la Base de Datos.
	4. El analista del Departamento. de Facturación selecciona la Casa Comercial.
	5. El analista selecciona el criterio de facturación por el que va a facturar. Los criterios de facturación que existen son: a) Todos los libros b) Todos los libros y lectores c) Un libro y un lector d) Todos los lectores.
	6. La aplicación SISMET localiza la información en la Base de Datos (para esto se consulta la tabla maestro de la Base de Datos actual).
	7. El Analista del Departamento de Facturación selecciona la opción de “Facturar”.
	8. La aplicación SISMET extrae el consumo.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	9. La aplicación SISMET verifica el origen (Ver flujo alterno 9a.).
	10. La aplicación SISMET; si no pertenece a la Casa Comercial de Peñas Altas verifica si el cliente es medrado o no (Ver flujo alterno 10a.).
	11. La aplicación SISMET consulta las claves (ver flujo alterno 11a.)
	12. Si es clave "0" se realiza la factura de la siguiente manera: Calculando: $\text{Importe} = \text{consumo} * 0.11$
	13. La aplicación SISMET imprime las Chequeras.
	14. La aplicación SISMET imprime los Recibos de Cobro.
	15. El analista del Departamento de Sistema entrega la información a al Departamento. de Venta y Facturación.
	16. El Departamento de Venta y Facturación revisa la información garantizando la corrección de posibles errores (Ver flujos alternos).
	17. El Departamento de Venta y Facturación en caso que no se haya detectado ningún error agrupa las chequeras por Casa Comercial.
	18. El Departamento de Venta y Facturación envía esta información a las Casas Comerciales.
Flujos Alternos	
9a.	Si el origen de la Casa Comercial es Peñas Altas <ol style="list-style-type: none"> 1. Se factura al cliente de la forma siguiente: $\text{Importe} = \text{consumo} * 0.11$ 2. Ir a la actividad 13.
10a.	Si es cliente es no Medrado: <ol style="list-style-type: none"> 1. Se factura al cliente de la forma siguiente: $\text{Importe} = \text{consumo} * \text{habitantes} * 0.0484$ 2. Ir a la actividad 13
11a.	Si es cualquier clave menos la "0" <ol style="list-style-type: none"> 1. Se factura al cliente de la forma siguiente $\text{Importe} = \text{promedio} * 0.11$ 2. Ir actividad 13
16a.	Si se detecta algún error. <ol style="list-style-type: none"> 1. Se realiza una "Retoma" Tipos de errores que pueden detectarse: <ol style="list-style-type: none"> a) Errores de impresión

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

<ul style="list-style-type: none"> b) Cuando sale la lectura del mes anterior (por lo que sale consumo 0) c) Cuando el importe es alto (ejemplo: que pase de los \$300) d) Cuando una chequera sale repetida. <p>2. Ir actividad 15</p>	
Mejoras propuestas.	[No identificadas]

3.5.2 Preparar TPL

Caso de uso del negocio	Preparar TPL
Actores	Lector Cobrador
Resumen	El caso de uso se inicia cuando el Lector Cobrador se presenta ante la Tramitadora para que esta le entregue el TPL con la información de los clientes a los que les tiene que efectuar lectura. La Tramitadora carga el TPL con la información necesaria, y le entrega el TPL al Lector Cobrador.
Acción del actor	Respuesta del proceso de negocio
1. El Lector Cobrador se presenta y solicita a la Tramitadora de Documentos de la Casa Comercial que prepare el TPL.	2. La Tramitadora de Documentos de la Casa Comercial introduce el nombre del fichero a preparar.
	3. La tramitadora de Documentos de la Casa Comercial consulta la "Listado de Ruta".
	4. La Tramitadora selecciona la forma en que desea cargar el TPL que puede ser por "cobrador y libro" o por "ruta y folio". (Ver flujos alternos 4a.)
	5. Introduce número de ruta y folio.
	6. La Tramitadora prepara el fichero creando el "Modelo de Tramitadora", "Modelo de Incidencia" y un nuevo "fichero" obteniendo toda esta información consultando la Base de Datos de SISTMET.
	7. La tramitadora selecciona la opción enviar la información al TPL. (Ver flujos alternos 7a.)
8. El Lector Cobrador recibe el TPL cargado con toda la información necesaria para efectuar las lecturas.	

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Flujos Alternos
4ª. Por cobrador y libro
1. Introduce identificador del cobrador y el libro.
7a. La Tramitadora puede visualizar un resumen de la información
1. La Tramitadora si lo desea puede imprimir un documento con toda la información enviada al TPL.

3.5.3 Efectuar Lectura

Caso de uso del negocio	Efectuar Lectura
Actores	Tramitadora de la Casa Comercial
Resumen	El caso de uso inicia cuando la Tramitadora de Documentos de la Casa Comercial le entrega al Lector Cobrador el TPL con los datos de los clientes. El Lector Cobrador procede a visitar a cada cliente para efectuar la lectura del metro contador, anota la lectura y su clave correspondiente, en caso de algún error se actualiza el modelo de incidencias reflejando la descripción del mismo. Una vez que termine de realizar las lecturas de todos los clientes entrega la información del TPL y Modelo de Incidencias a la Tramitadora de la Casa Comercial.
Acción del actor	Respuesta del proceso de negocio
1. La Tramitadora de la Casa Comercial entrega el TPL cargado con la información de todos los clientes al Lector Cobrador para que efectúe las lecturas.	
	2. El Lector Cobrador llega a la casa del cliente y observa la lectura del Metro Contador. (Ver Flujo Alterno 2ª)
	3. Teclea en el TPL la lectura observada en el Metro Contador.
	4. Presiona el botón ENTER del TPL, almacena la lectura en el fichero del sistema. (Ver Flujo Alterno 4ª)
	5. Teclea en el TPL la clave correspondiente a la observación de la lectura.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	6. Presiona el botón ENTER del TPL, almacena la clave en el fichero del sistema correspondiente a la lectura observada. (Ver Flujo Alterno 6ª)
	7. Entrega el TPL a la Tramitadora de la Casa Comercial con la información de la lectura de los clientes.
8. La aplicación SISMET extrae el consumo.	
Flujos Alternos	
2ª. En caso de que no se pueda observar el metro contador, el Lector Cobrador registra en el TPL la clave 9 (Difícil Acceso) correspondiente a la situación presentada. Ir a actividad 5.	
4ª En caso de que el Lector Cobrador se haya equivocado al introducir la Lectura del Metro Contador, actualiza el Modelo de Incidencia, especificando la lectura correcta y la descripción. Ir a actividad 5.	
6ª En caso de que el Lector Cobrador se haya equivocado al introducir la clave correspondiente a la lectura observada, actualiza el Modelo de Incidencia, especificando la clave correcta y la descripción. Ir a actividad 7.	
Mejoras propuestas.	Automatizar el proceso de lectura del consumo, poniendo en cada vivienda un dispositivo de lectura automático que se encargue de contabilizar el consumo de M3 de Gas y que pueda ser leído desde el sistema a través de redes inalámbricas.

3.5.4 Preparar la facturación emitida.

Caso de uso del negocio	Preparar la facturación emitida.
Actores	Departamento de venta y facturación.
Resumen	El caso de uso inicia cuando el Departamento de Venta y Facturación envía los Talonarios y los Recibos de Cobro a cada una de las Casas Comerciales, en las cuales las Tramitadoras de Documentos separan la información por lector Cobrador y efectúan la entrega a cada uno de ellos para que comiencen el cobro en el terreno.
Acción del actor	Respuesta del proceso de negocio
1. El Departamento de Venta y	2. En la Casa Comercial la Tramitadora de

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

<p>Facturación envía a las Casas Comerciales los documentos para empezar el proceso de cobro. Estos son: Recibos de Cobro por Libros y Cobradores y los Talonarios con las deudas de los clientes.</p>	<p>Documentos recibe lo que envía la Casa Comercial. Aquí son revisados y archivados con la posible deuda de los clientes organizados por rutas.</p>
	<p>3. Ya con toda la información en orden entonces la Tramitadora procede a agrupar la información por cada cobrador para lo cual se emite el Listado de Cobro.</p>
	<p>4. El Listado de Cobro es revisado y puntuado por ella misma y a su vez se adjuntan los Talonarios de Cobro.</p>

3.5.5 Efectuar cobro a clientes en la vivienda.

Caso de uso del negocio	Efectuar cobro a clientes en la vivienda.
Actores	Tramitadora de Documentos
Resumen	El caso de uso inicia cuando el Comercial de la Casa Comercial le entrega el Listado de Cobro y los Talonarios al Lector Cobrador. El mismo efectúa el cobro en el terreno y al terminar entrega los Talonarios y el dinero al Controlador Económico.
Acción del actor	Respuesta del proceso de negocio
1. Entrega de información al Lector Cobrador.	2. El Lector Cobrador recibe el trabajo del día.
	3. Procede a visitar a los clientes asignados para efectuarles el cobro. La cual es guiada por los Talonarios de Cobro que reflejan la dirección del usuario así como su importe a pagar.
	4. Si el cliente se encuentra en la casa y desea pagar, se le cobra el servicio, dejándole un comprobante como constancia de haber pagado el Gas Manufacturado. En el cobro del servicio se actualizaran los talonarios los cuales perderán dos de sus tres secciones. (Ver

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	Flujo alternativo 4a, 4b.)
	5. Ya con las cuentas cobradas el Lector Cobrador procede a llevar las facturas y el efectivo a la Casa Comercial.
Flujos Alternos	
4a. Si el cliente no se encuentra en la casa se procede de la siguiente manera: el Lector Cobrador le deja el comprobante de aviso de que tiene que pagar el servicio en la Casa Comercial en diez días hábiles. Ir a la actividad 5.	
4b. Si el cliente esta y no desea pagar el servicio se procede de la misma manera que la situación anterior.	

3.5.6 Efectuar cobro a clientes en la Casa Comercial.

Caso de uso del negocio	Efectuar cobro a clientes en la casa Comercial.
Actores	Cliente
Resumen	El caso de uso se inicia cuando el cliente procede a efectuar el pago del servicio a la Casa Comercial en donde la Cajera procede a cobrarle el servicio.
Acción del actor	Respuesta del proceso de negocio
1. El cliente procede a efectuar el pago en la Casa Comercial.	2. En la Casa Comercial la Tramitadora de Documentos que tiene que ver con la caja de la casa efectúa el cobro en la misma. En este proceso se actualizarán una serie de documentos entre los cuales se encuentran: Listado de Cobro, las Facturas de Pago de los Clientes, y el Modelo Recibo de Cobro.
	3. Ya con el servicio pagado la Tramitadora efectúa el cuadro de los clientes a los que se les aplicó el cobro del gas.
	4. Con toda la información del día los dos trabajadores tanto la Tramitadora como el Lector Cobrador le entregan la información al Controlador Económico de la Casa Comercial.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

3.5.7 Procesar Información de Cobro.

Caso de uso del negocio	Procesar Información de Cobro
Actores	Cobrador
Resumen	El caso de uso se inicia cuando los Cobradores y Tramitadoras entregan el dinero y los talonarios al Controlador Económico quien procede a la confección de varios modelos (ver descripción y modelo de objetos).
Acción del actor	Respuesta del proceso de negocio
1. El Lector Cobrador entrega el dinero y los talonarios al Controlador Económico.	2. El Controlador Económico recibe el dinero y las Facturas de Cobro Mercado de Consumo debidamente acuñadas y fechadas.
	3. El Controlador Económico encargado de la custodia del efectivo cuenta el dinero que entregan los Cobradores y Tramitador de Documentos, confeccionando el Modelo Liquidación de Efectivo en original y copia que tiene que estar firmado por ambos, guarda el original en custodia y la copia la entrega al Cobrador y Tramitador.
	4. El Controlador Económico encargado de la recepción de las cuentas cobradas comprueba que las suma de estos y el importe reflejado por el Cobrador o Tramitador en el cargo al cobrador y el Modelo Liquidación de Efectivo sean las mismas y confecciona el modelo IDC (Informe diario de caja). Colocan una copia de dicho modelo en la copia del cargo al cobrador A16 que se queda en la oficina.
	5. El Controlador Económico tiene que comprobar que las cuentas cobradas y por cobrar coincidan con las cargadas en el cargo al cobrador debe hacerse total y no parcial.
	6. El Controlador Económico confecciona el Modelo Informe Diario del Trabajo realizado por cobradores reflejando el trabajo diario de los cobradores.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

3.5.8 Procesar efectivo recaudado.

Caso de uso del negocio	Procesar efectivo recaudado
Actores	Cobrador
Resumen	El caso de uso inicia cuando el Cobrador o la Cajera entrega el efectivo y los talonarios al Controlador Económico, quien realiza una serie de comprobaciones al efectivo y los talonarios recibidos y procede a generar reportes de control de la actividad económica en la Casa Comercial, así como el depósito del efectivo en el banco.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El Lector Cobrador le entrega el dinero efectivo y los talonarios al Controlador Económico.	2. El Controlador Económico contado el efectivo recibido, desglosa por valores, suma los importes y confecciona slip para banco.
	3. El Controlador Económico lleva el efectivo al banco y recibe el comprobante como constancia.
	4. El Controlador Económico confecciona el Modelo Resumen Diario de Cobros, reflejando todos los cobros del día, se suma y comprueba que el importe final sea igual al importe depositado en el banco y también confecciona el Movimiento Diario de Efectivo en Caja, reflejando al final del día toda la entrada y salida del efectivo recibido y el que queda guardado en caja al final de la Jornada Laboral.
	5. Los Controladores Económicos adjuntan el Resumen Diario de Cobro en efectivo y Movimiento Diario de Efectivo en Caja y Recibos de Cobro con sus respectivos Informe Diario de Caja y Devolución del Efectivo si hubiera, lo firman y entregan al Jefe de la Casa Comercial para su revisión y firma y envían a la Dirección Comercial.
	6. Los Auxiliares Económicos y Controladores revisan si los Recibos de Cobro están debidamente confeccionados cumpliendo todos los requisitos necesarios.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

3.5.9 Realizar cierre del cobro.

Caso de uso del negocio	Realizar cierre del cobro
Actores	Jefe del Departamento. de Ventas y Facturación
Resumen	El caso de uso se inicia cuando el Jefe del Departamento de Ventas y Facturación le envía los Talonarios al Analista de Sistemas del Departamento de Sistemas.
Casos de uso asociados	
Acción del actor	Respuesta del proceso de negocio
1. El Jefe del Departamento de Ventas y Facturación le envía los Talonarios al Analista de Sistemas del Departamento de Sistemas.	2. El Analista de Sistemas del Departamento de Sistemas recibe los Talonarios y procede a su organización y entrega a la Tecleadora del Departamento de Sistemas.
	3. La Tecleadora del Departamento de Sistemas recibe los Talonarios y teclea el recibo de estos.
	4. El Analista de Sistemas del Departamento de Sistemas procede a crear el Reporte de Cuadre de Talonarios y el Reporte de Cuadre de H67.
	5. En el Departamento de Ventas y Facturación proceden a la creación de los siguientes modelos: <ul style="list-style-type: none">• Resumen H67 por caja.• Reporte del lector-cobrador.• Resumen de caja.• Resumen de deuda.• Certificación de deuda.• Resumen de cobro.

3.6 Reglas del Negocio

Las reglas del negocio son la declaración de políticas y condiciones que han de satisfacerse. Se utilizan cuando hay muchas o complejas condiciones guiando las operaciones del negocio.

1. Tienen que estar facturados todos los clientes para realizar el cierre de mes.
2. Deben estar tecleados y actualizados el estado de cobro y deuda del cliente antes de realizar el cierre de mes.
3. Debe estar hecho el cierre contable de cobro para efectuar el cierre de mes.
4. Todos los clientes deben estar en estado pendiente de lectura, o sea, listos para ser leídos cuando se indique por parte del jefe del departamento de sistema que comience el proceso de lectura.
5. Debe estar instalado Mi Psion (aplicación que permite enlazar el TPL con la computadora) para poder interactuar con el Terminal Portátil de Lectura (TPL)
6. Antes de ser introducida la lectura del metro contador por el Lector Cobrador debe ser introducida la clave correspondiente.
7. La información de las lecturas que se emite desde las casas comerciales hacia el departamento de sistema del la UEB Comercial no puede tener errores.
8. Para poder emitir los cobros por libros tienen que haberse emitido los recibos de cobro.
9. Para visitar a un cliente, antes deben poseerse los talonarios.
10. Para hacer la liquidación del efectivo tiene que haberse recibido el dinero.
11. Después de una clave 1 debe ponerse una clave 5.
12. No deben existir varias claves 7 en una misma ruta.
13. Todas las lecturas tienen que tener clave.
14. Cuando el consumo es mayor de 100 m se factura por clave 2 (bajo consumo a investigar).
15. Siempre que sea clave 5, la Tramitadora introduce número de metro.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

16. Para procesar las Lecturas que lleve un Lector cobrador, este debe tener por lo menos un 96% de lecturas efectivas. (Clave 6 y 9 no efectivas)
17. La Tramitadora de Documentos confecciona los modelos Recibo de cobro y en ellos debe señalar el tipo de cuenta que se está cobrando con él número de la clave que corresponde.

Claves	Conceptos
1	Cobro año actual
2	Instalación de cocina
3	Reconexión
4	Diferencia de facturación
5	Otros
6	Cuentapropista
7	Gastos de servicio
8	Consumo sin contrato
9	Año anterior
0	Venta de cocina

Se distribuirá el modelo Recibo de cobro de la siguiente manera, el original para el cliente, 1ra copia para contabilidad de la Empresa, 2da copia Casa Comercial y 3ra copia al Departamento de Ventas de la UEB COMERCIAL.

18. Cuando hay extravío de chequeras se confecciona Modelo Recibo de Cobro con las claves 1 o 9.
19. Cuando las facturas extraviadas son varias debe prepararse una relación detallada con la explicación de lo ocurrido y la firma del jefe de la Casa Comercial para justificar el importe total de los cobros por recibos (modelo Recibo de cobro).
20. Los saltos que se detecten en la numeración de los Recibo de cobro en el archivo del Departamento. de Ventas serán investigados por el inspector y/o controlador económico de este

Departamento. de la UEB COMERCIAL.

21. El expediente de cobro que confecciona la Casa Comercial está integrado por los documentos siguientes:

- Total de recibos de las cuentas cobradas.
- Total de modelos Recibo de Cobro cobrado.
- Original y 1ra copia modelo Devoluciones de Efectivo (Informe diario de Caja). Resumen diario de cobro en efectivo.
- Liquidación de Cargo al Cobrador.

3.7 Especificación de los requisitos del Software.

3.7.1 Requisitos Funcionales.

La gestión de requisitos se identifica actualmente como una muy buena práctica que contribuye, como ninguna otra, al éxito de los proyectos de software, al posibilitar un entendimiento común entre el cliente y el grupo de desarrolladores de los requisitos del cliente que deben concebirse en el producto final, la comprensión de los problemas que se necesitan solucionar y las posibles vías de resolverlos. Es en este paso donde se establece la especificación preliminar de los Requisitos Funcionales y No Funcionales; así como los Casos de Uso de los procesos que se desean automatizar: Constituye el documento legal primario para el diseño y construcción de la aplicación futura. Un Requisito Funcional se define como una característica requerida por el cliente para solucionar un problema o conseguir un objetivo. Es lo que el cliente le solicita al sistema, porque es lo que necesita.

Según Roger S Pressman para que un esfuerzo de desarrollo de software tenga éxito, es esencial comprender perfectamente los requisitos del software. Independientemente de lo bien diseñado o codificado que esté un programa, si se ha analizado y especificado pobremente, decepcionará al usuario y desprestigiará al que lo ha desarrollado.

Además, según Frederick P. Brooks, Jr:

“... la parte más difícil en la construcción en sistemas de software es decidir precisamente qué construir. Ninguna otra parte del trabajo conceptual es tan ardua como establecer los

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

requerimientos técnicos detallados, incluyendo todas las interfaces con humanos, máquinas y otros sistemas de software. Ninguna otra parte del trabajo puede perjudicar tanto el resultado final si se realiza de forma errónea. Ninguna otra parte es tan difícil de rectificar posteriormente”

Estas dos ideas abordan de manera muy clara la importancia de realizar una identificación correcta de las necesidades que el sistema debe satisfacer, es por ello que las actividades de identificación de requerimientos deben realizarse con la mayor organización y seriedad posible.

Algunos de los requerimientos son:

RF1. Introducir Nombre Fichero TPL.

El sistema debe permitir introducir el nombre del fichero que se va a crear con los datos de los clientes a facturar, para cargar el TPL.

RF 2. Seleccionar Ruta o Casa Comercial a Facturar.

El sistema debe permitir seleccionar la ruta que se desea facturar según el criterio de factura o por la Casa Comercial.

RF 3. Buscar Datos de Clientes a Realizar Lectura.

El sistema debe permitir realizar búsquedas en la Base de Datos donde se encuentra la información de los clientes que pertenecen a la ruta que se va a facturar.

RF 4. Cargar Datos de Clientes a los cuales se les va a realizar la factura con el fichero nombrado en el RF1.

El sistema debe permitir que se carguen los datos de los clientes que se buscaron en el R2 que se van a facturar y se guardan en el fichero que se crea en el RF1.

RF 5. Cargar fichero al TPL.

El sistema debe permitir que los datos de los clientes a facturar sean cargados hacia el TPL y el mismo sea preparado para entregárselo a cada uno de los lectores cobradores según los siguientes criterios:

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

- número de la Ruta y el Folio.

- identificador del cobrador y el libro.

RF 7. Crear Modelo de Tramitadora.

El sistema debe crear el Modelo de Tramitadora, con los datos de los clientes que se van a Facturar.

RF 8. Mostrar datos a cargar al TPL

El sistema debe mostrar un listado con los datos de los clientes que se van a cargar en el TPL.

RF 9. Imprimir datos a cargar en el TPL

El sistema debe imprimir los datos de los clientes que se van a cargar en el TPL.

RF 10. Cargar datos a Facturar.

El sistema debe cargar los datos a facturar desde un fichero con extensión .dbf a través del TPL con los datos de los clientes que es enviado por las Tramitadoras de las Casas Comerciales hacia la UEB Comercial.

RF 13. Seleccionar criterios para facturar

El sistema debe mostrar diferentes tipos de criterios a facturar

- Todos los libros.
- Todos los libros y lectores.
- Un libro y un lector.
- Todos los lectores.

RF 14. Cargar la Tarifa Escalonada

El sistema debe cargar los datos de un fichero con distintos rangos de consumo con los cuales se establece la tarifa escalonada por la que se va a facturar.

RF 15. Cargar consumo de cliente

El sistema debe cargar de la Base de Datos de la UEB Comercial el consumo mensual de cada cliente para realizarle la factura según el consumo registrado por el lector cobrador

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

RF 17. Mostrar clientes facturados

El sistema debe mostrar un reporte con todos los datos de los clientes facturados.

RF 18. Corregir errores facturación

El sistema debe permitir que se corrijan los errores que se produjeron en el proceso de facturación.

RF 19. Guardar datos de Facturación

El sistema debe guardar los datos de la Facturación en la Base de Datos de la UEB Comercial, después que estos hayan sido corregidos.

RF 21. Imprimir cobros

El sistema debe permitir imprimir los cobros.

RF 22. Descargar datos del TPL

El sistema debe permitir que se descarguen los datos de las lecturas de los clientes del TPL.

RF 23. Comprobar datos del TPL

El sistema debe comparar los datos que se descargaron del TPL con los datos que se encuentran en el Modelo de Tramitadora y buscar incongruencias.

RF 24. Comprobar el % de lecturas

El sistema debe detectar cuando el % de lecturas realizadas por el Lector Cobrador no supera el % de lecturas esperado.

RF 26. Corregir errores en los datos del TPL

El sistema debe permitir que se corrijan los errores encontrados en los datos descargados del TPL.

RF 27. Calcular el consumo del cliente en el mes

El sistema debe calcular el consumo del mes actual de cada cliente.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

RF 28. Guardar datos de lecturas y consumo de cada cliente.

El sistema debe guardar los datos corregidos de las lecturas que se descargaron del TPL y el consumo de cada cliente en la Base de Datos que se encuentra en cada una de las Casas Comerciales.

RF 29. Mostrar informes de lecturas corregidas.

El sistema debe mostrar un reporte con los datos de las lecturas corregidas listas para enviarlas a la UEB Comercial.

RF 30. Exportar datos de lectura a fichero.

El sistema debe de exportar los datos de las lecturas corregidas hacia un fichero.

RF 31. Emitir listado a entregar al lector-cobrador

El sistema debe permitir que la tramitadora genere un reporte con la ruta, el mes y el importe de cada cliente para que cada cobrador pueda cobrarle a cada cliente.

RF 32. Registrar cobro de clientes en las casas comerciales.

El sistema debe permitirle al usuario con el rol de cajera registrar que el cliente efectúe el cobro en efectivo en la casa comercial.

RF 33. Generar recibo de cobro al cobrar servicio de gas.

El sistema debe permitirle al usuario con el rol de cajera imprimir un comprobante de cobro después de haber efectuado el mismo.

RF 34. Generar IDC (Informe Diario de Caja)

El sistema debe permitirle al usuario con el rol de auxiliar económico generar reporte IDC.

RF 35. Generar Modelo liquidación en efectivo por cobrador.

El sistema debe permitirle al usuario con el rol de cajera generar reporte de liquidación de efectivo, desglosado por cobrador.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

RF 36. Generar Informe Diario del Trabajo realizado por cobradores

El sistema debe permitirle al usuario con el rol de controlador económico generar Informe Diario del Trabajo realizado por cobradores.

RF 37. Comprobar las cuentas cobradas y por cobrar con las presentes en el modelo Cargo al Cobrador

El sistema debe permitir al usuario con el rol de controlador económico comprobar que las cuentas cobradas y por cobrar coincidan con las que están reflejadas en el modelo Cargo al Cobrador.

RF 38. Generar Resumen Diario de cobros en efectivo

El sistema debe permitir al usuario con el rol de controlador económico generar reporte Resumen Diario de cobros en efectivo.

3.7.2 Requisitos No Funcionales

Un Requisito no Funcional, por su parte, es aquel que, no por ser No Funcional deja de tener funcionalidad; puede comprometer el funcionamiento del producto desarrollado y permite que el resultado, es decir, el software sea atractivo, usable, rápido; y tiene como premisa que, una vez conocido lo que el sistema debe hacer, se determine cómo ha de comportarse y qué cualidades debe tener este.

3.7.2.1 Usabilidad.

La aplicación debe ser concebida para ser utilizada por personas que tengan conocimientos básicos sobre informática y en el trabajo con los procesos de Facturación y Cobro.

El sistema debe estar disponible las veinticuatro horas del día, sin ninguna interrupción.

El sistema debe ser accesible desde todos los puntos donde exista una máquina conectada a la red.

El sistema debe diferenciar las interfaces gráficas y opciones para los usuarios que accedan al sistema con diferentes roles.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

3.7.2.2 Apariencia o Interfaz Externa.

Diseño sencillo y fácil de usar, permitiendo que no sea necesario mucho entrenamiento para utilizar el sistema.

3.7.2.3 Portabilidad.

La herramienta debe ser capaz de correr en cualquier sistema operativo. Esto se debe a que será una aplicación Web, implementada con PHP que es un lenguaje multiplataforma.

3.7.2.4 Confiabilidad.

Ante fallas técnicas del sistema, corregirlas en un período menor a un mes.

Si este sistema presentara algún desperfecto, este será corregido en un período menor a un mes para que no afecte a sus usuarios. La manera de garantizar la mayor aceptación por parte del cliente es el soporte con que pueda contar el software dándole seguimiento a través de un procedimiento de soporte del software por los desarrolladores.

El sistema debe tener la capacidad de identificar con certeza a los diversos usuarios o entidades que interactúan con él.

El sistema debe tener la capacidad de darle seguridad al usuario, de que las informaciones solo serán vistas por quien esté capacitado para esto.

3.7.2.5 Rendimiento.

El sistema debe responder en un tiempo relativamente rápido a las peticiones del usuario (menos de 1 segundos).

Teniendo en cuenta que el sistema no es de tiempo real y no necesita de respuestas de milisegundos se plantea este como un tiempo máximo estimado el cual debe ser mucho menor, pero si se encuentra en estos límites no debe afectar el funcionamiento del sistema.

3.7.2.5 Soporte.

El software podrá operar en cualquier sistema operativo debido a que se desarrollará en PHP y herramientas de software libre, con la excepción del sistema gestor de base de datos SQL Server 2000 que solo corre sobre el Sistema Operativo Windows.

3.7.2.6 Requerimiento de ayuda y documentación.

Proporcionar documentación de ayuda al usuario según el rol que va a desempeñar en el sistema. Como manuales de usuario, manuales de instalación, entre otros, donde en el manual de usuario se encuentre explícitamente cada una de las funciones de cada rol.

3.7.2.7 Restricciones de diseño e implementación.

- ✚ El sistema debe usar framework Symfony 1.011
- ✚ El sistema se desarrollará en lenguaje PHP 5.0 Orientado a Objeto.

3.7.2.8 Requerimientos de Hardware.

Mínimo

Servidor Web: Hardware de la estación de trabajo del servidor Web, donde se ejecutará el sistema

- ✚ Procesador Pentium III o superior 1x2 cache. 2.8 GHz.
- ✚ Memoria RAM 256 MBytes.
- ✚ Almacenamiento en discos de 5 GBytes.

Servidor de Base de Datos: Hardware de la estación de trabajo servidor BD.

- ✚ Procesador Pentium III o superior 1x2 cache. 2.8 GHz.
- ✚ Memoria RAM 512 MBytes.
- ✚ Almacenamiento en discos de 40 GBytes.

Cliente Web: Hardware de la estación de trabajo del cliente.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

- ✚ Procesador Pentium III o superior o Clientes Ligeros.
- ✚ Memoria RAM 128 MBytes.

Recomendable

Servidor Web: Hardware de la estación de trabajo del servidor Web, donde se ejecutará el sistema

- ✚ Procesador Pentium III o superior 2x2 cache. ≥ 3.00 GHz.
- ✚ Memoria RAM ≥ 512 MBytes.
- ✚ Almacenamiento en discos de 5 GBytes.

Servidor de Base de Datos: Hardware de la estación de trabajo servidor BD.

- ✚ Procesador Pentium D 2x2 cache. 3.00 GHz.
- ✚ Memoria RAM ≥ 1 GBytes.
- ✚ Almacenamiento en discos de 40 a 120 GBytes.

Cliente Web: Hardware de la estación de trabajo del cliente.

- ✚ Procesador Pentium III o superior o Clientes Ligeros.
- ✚ Memoria RAM ≥ 256 MB.

3.7.2.9 Requerimientos de Software.

PC Cliente: Software instalado en la estación de trabajo del cliente.

- ✚ Sistema Operativo Windows 2000 o Superior. El Navegador Web compatible con HTML 2.0 y CSS, podrá ser Netscape 3 (o superior), Internet Explorer 4.2 (o superior) y compatibles.

PC Servidor Web: Software instalado en el Servidor Web.

- ✚ Servidor de Web Apache 2.2.1

PC Servidor de BD: Software instalado en el Servidor de Base de datos.

- ✚ Servidor de Bases de Datos SQL-Server 2000.

3.7.2.10 Seguridad.

✚ **Confidencialidad:** la información manejada por el sistema debe estar protegida de acceso no autorizado y divulgación.

Crear cuentas de usuarios que contengan roles y permisos sobre el sistema.

✚ **Integridad:** Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos, que esté libre de manipulaciones no autorizadas, accidentales o deliberadas, de manera tal que el sistema funcione perfectamente y brinde el mejor servicio.

✚ **Disponibilidad:** A los usuarios se les garantizará el acceso en cualquier horario de trabajo así como en tiempo extra, garantizando la búsqueda de información, lo que posibilitará a los mismos obtener los datos deseados en un momento dado. Porcentaje de éxito en obtener lo que se requiere.

3.8 Descripción del sistema propuesto

El análisis de la producción de las casas comerciales se realiza con frecuencia mensual para el estudio de las lecturas, facturación y cobro del gas manufacturado que se suministra a las familias de la capital cubana y que el consumo de dicho gas se mide a través de metros contadores. Entre los métodos de evaluación se utiliza la comparación con los valores históricos, lo que requiere inevitablemente el almacenamiento de los datos para el uso en años posteriores. El Sistema es el encargado de procesar no solo la lectura de los metros contadores sino de generar algunos reportes para la realización del cobro de las familias y generar los talonarios para cada una de las familias que cuentan con el servicio de Gas Manufacturado para gestionar no solo el cobro de cada uno de los clientes ya sea en la Casa Comercial o en el hogar de cada cliente sino también gestionar las facturas que se obtienen de cada reporte emitido en dicho sistema. Con el propósito de satisfacer los requerimientos planteados anteriormente se ha analizado la creación de un sistema que permita facturar por una tarifa escalonada dinámica según los metros cúbicos consumidos por cada cliente, facilidades de pago a los clientes en diferentes casas comerciales, consulta y reporte de información del estado de facturación y cobro actualizada en cualquier momento.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

El sistema cuenta con cinco módulos:

Módulo de Seguridad: El módulo consiste en que en el sistema debe brindarle al cliente la mayor seguridad posible, y en dependencia del cargo que ocupe en la empresa son los permisos que puede tener en el sistema. Ya sea el Analista del Sistema, la cajera u otro tipo de persona que tenga relación con el mismo.

Módulo de Facturación: El módulo consiste en lograr que el nuevo sistema emita las facturas de una manera más organizada, segura y cómoda posible para los usuarios del sistema en cuestión.

Módulo de Cobro: El módulo consiste en que dada la factura emitida por el sistema de manera escalonada.

Módulo de Administración: El módulo consiste en gestionar usuarios a en el sistema.

Módulo de Reporte: El módulo consiste en generar todos los reportes que se van a generar en la empresa.

3.9 Definición de Casos de Uso del Sistema

El Modelo de Casos de Uso del Sistema es un modelo que contiene actores, casos de uso y sus relaciones.

3.9.1 Actores del Sistema

Cada trabajador del negocio (inclusive si fuera un sistema ya existente) que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema. Los Actores del sistema:

- ✓ No son parte de él.
- ✓ Pueden intercambiar información con él.
- ✓ Pueden ser un recipiente pasivo de información.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

- ✓ Pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado.

Actor	Descripción
Tramitadora Casa Comercial	Este actor es el encargado de cargar los datos hacia el TPL y la descarga de estos, corrige los errores que se produjeron en el proceso de lectura y las incidencias.
Analista del Sistema de la UEB Comercial	Este actor es el encargado de cargar los datos que son enviados por las Tramitadoras de la Casa Comercial, también realiza el proceso de Facturación.

3.10 Diagrama de Casos de Uso del Sistema

Un Diagrama de Casos de Uso del Sistema representa gráficamente a los procesos y su interacción con los actores. Cada Caso de Uso debe comunicarse con al menos un actor, si no aparece ningún actor que se comunique con un Caso de Uso, esto indica error en el Modelo de Caso de Uso o en los requerimientos planteados.

3.10.1 Proceso de Facturación

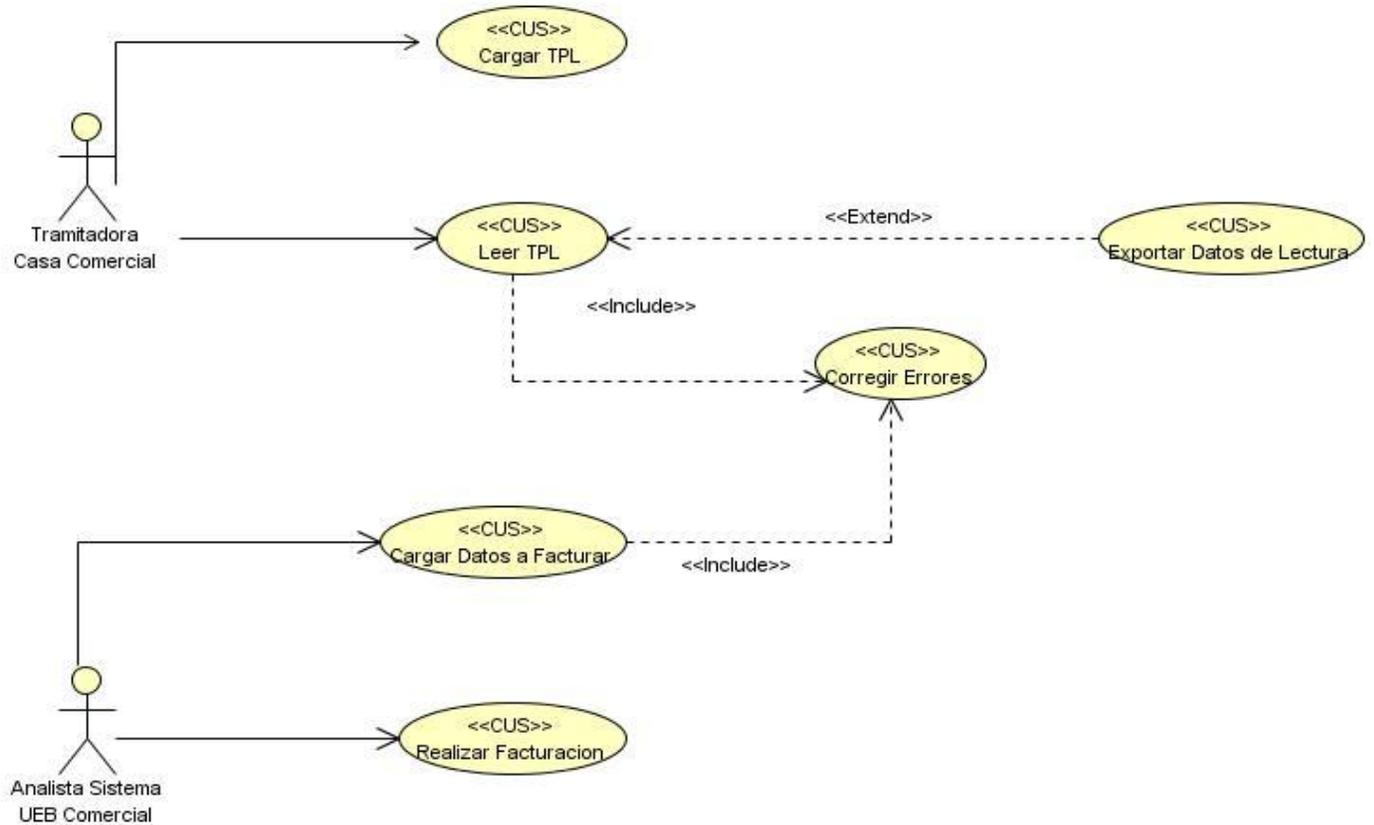


Figura 4 Diagrama de Caso de Uso del Sistema de Facturación.

Casos de Uso del Sistema

Los Casos de Uso del Sistema son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Cargar TPL
Actores:	Tramitadora de la Casa Comercial.
Resumen:	Este caso de uso comienza cuando la Tramitadora de la Casa Comercial va a cargar los datos de los clientes a los que se le va a efectuar la lectura, selecciona la ruta a la que se le va a efectuar la lectura, genera el fichero con los datos, crea el Modelo de Tramitadora y carga los datos en el TPL.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol de Tramitadora.
Referencias	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10.
Prioridad	

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. La Tramitadora de la casa comercial selecciona la opción Cargar TPL.	2. El sistema muestra un campo para Introducir el nombre del fichero a crear.
3. La Tramitadora introduce el nombre del fichero que se va a crear con los datos de los clientes a facturar.	
4. La Tramitadora da click en el botón de Aceptar.	5. El sistema muestra un listado con las rutas a facturar por el lector cobrador.
6. La Tramitadora selecciona la ruta que se va a facturar y da click en el botón de Aceptar.	7. El sistema pide la selección de la ruta y el folio por las que se va a facturar. (flujo alterno 7ª)
8. La Tramitadora selecciona la ruta y el folio a facturar por el Lector Cobrador.	9. El sistema prepara los datos de los clientes que se van a facturar por número de Ruta y Folio.
	10. El sistema busca los datos de los clientes que pertenecen a las rutas seleccionadas.
	11. El sistema carga los datos de los clientes que pertenecen a la ruta seleccionada.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	12. El sistema muestra los datos de los clientes que se van a facturar.
	13. El sistema crea el Modelo de Tramitadora con los datos del cliente que se va a facturar.
14. La Tramitadora escoge la opción de imprimir el Modelo de Incidencias.	15. El sistema crea e imprime el Modelo de Incidencias.
16. La Tramitadora escoge la opción de imprimir los datos de los clientes que serán facturados.	17. El sistema imprime los datos de los clientes que se van a facturar.
18. La Tramitadora escoge la opción de Exportar los Datos de los Clientes.	19. El sistema exporta los datos de los clientes que se van a facturar a un archivo .dbf
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	7ª El sistema pide la selección del identificador del cobrador y el libro.
8ª La Tramitadora selecciona el identificador del cobrador y el libro.	9ª El sistema prepara los datos del TPL por el identificador del cobrador y por libros.(Ir a la actividad 10)

Tabla 3.10.1.1.2 Caso de Uso Cargar TPL

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Leer TPL
Actores:	Tramitadora de la Casa Comercial
Resumen:	El caso de uso comienza cuando la Tramitadora de la Casa Comercial escoge la opción de Descargar las Lecturas de los Clientes del TPL, comprueba los datos de estas lecturas con los datos existentes en el Modelo de la Tramitadora, corrige los errores, las incidencias existentes, calcula el consumo de cada cliente y envía estos datos hacia a la UEB Comercial.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol de Tramitadora.
Referencias	R21, R22,R23,R25, R28, R29, R30
Prioridad	

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. La Tramitadora de la Casa Comercial escoge la opción de Cargar los Datos de la lectura efectuada desde un fichero (extensión .dbf).	
2. La Tramitadora presiona el botón Examinar para buscar la ubicación del fichero a cargar.	3. El sistema muestra una ventana pidiendo la dirección donde se encuentra el fichero.
4. La Tramitadora de la Casa Comercial busca la ubicación del fichero y presiona el botón Aceptar.	
	5. El sistema carga el fichero leyendo todos los datos de los clientes.
	6. El sistema verifica que todos los campos de cada cliente estén llenos y correctos.
	7. El sistema compara los datos cargados del fichero con los datos existentes en el Modelo de la Tramitadora, buscando

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	incongruencias. (Flujo Alternativo 7ª)
	8. El sistema comprueba que el % de lecturas efectuadas sea superior al % de lecturas válidas (Flujo Alternativo 8ª)
	9. Invoca al Caso de Uso <u>Corregir Errores</u>
	10. El sistema muestra un listado de todas las lecturas corregidas, marcando las que presentaron errores.
11. La Tramitadora de la Casa Comercial escoge la opción de corregir incidencias.	
	12. El sistema muestra un listado de las lecturas brindando la posibilidad de corregir los datos del modelo de incidencia que son la lectura y la clave.
13. La Tramitadora de la Casa Comercial corrige todas las lecturas que presentaron incidencias.	
14. La Tramitadora presiona el botón Aceptar para confirmar que todas las lecturas fueron corregidas.	
	15. El sistema valida que los datos corregidos estén correctos. (Invoca al Caso de Uso <u>Corregir Errores</u>) (Flujo Alternativo 15ª)
	16. El sistema calcula el consumo de cada cliente leído, restando la lectura actual con la lectura del mes anterior.
	17. El sistema guarda en la Base de Datos los datos de las lecturas y el consumo de cada cliente.
	18. El sistema muestra un listado con la lectura y el consumo de cada cliente.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	7ª El sistema encuentra incongruencias en los datos cargados y los datos existentes en el Modelo de la Tramitadora, reportando esto a la Tramitadora de la Casa Comercial.
	8ª El sistema detecta que el % de lecturas efectuadas no supera al % de lecturas esperado, reporta esto a la Tramitadora de la Casa Comercial.
8.1ª La tramitadora acepta o deniega el % de lecturas efectuadas.	
	15ª El sistema detecta que alguno de los datos corregidos están incorrectos, volviendo a pedir la entrada de estos datos.(Ir a la actividad 13)

Tabla 3.10.1.1.3 Caso de Uso Leer TPL

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Realizar Facturación
Actores:	Analista del Sistema de la UEB Comercial
Resumen:	El Caso de Uso comienza cuando el Analista selecciona la Casa Comercial que va a facturar, se actualiza los datos en la Base de Datos, selecciona la Casa Comercial que desea facturar, los criterios por los que va a facturar, se factura a cada cliente seleccionado, se corrigen los errores ocurridos en el cálculo de la Facturación, se guardan los datos, se imprimen la chequeras y los cobros, finaliza el Caso de Uso.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el rol de Analista del Sistema.
Referencias	R11, R12, R13, R14, R15, R16, R17, R18, R19, R20
Prioridad	

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1- El Analista del Sistema escoge la Casa Comercial que va a Facturar, da click en el botón de Aceptar para confirmar la selección.	
	2- El sistema muestra varios criterios para realizar la Facturación de los clientes. <ul style="list-style-type: none"> - Todos los libros - Todos los libros y lectores - Un Libro y un Lector - Todos los lectores
3- El Analista del Sistema escoge el criterio por el que desea Facturar a los clientes, dando click en el botón de Aceptar para confirmar la selección.	
	4- El sistema carga desde un fichero la Tarifa Escalonada por la que se va a facturar a

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	los clientes, teniendo en cuenta los datos de la Casa Comercial y los de los clientes.
	5- El sistema busca y carga el consumo desde la Base de Datos de SISMET de los clientes, teniendo en cuenta los criterios realiza la facturación.
	6- El sistema factura a cada cliente teniendo en cuenta el consumo de este y los datos cargados de la Tarifa Escalonada.
	7- El sistema muestra un reporte con los datos de la Facturación de cada cliente, permitiendo que estos datos sean corregidos.
8- El Analista corrige los errores que se produjeron en el proceso de Facturación, dando click en el botón de Aceptar para confirmar los cambios	
	9- El sistema guarda los datos de la facturación de los clientes en la Base de Datos de SISMET.
10- El Analista escoge la opción de Imprimir las Chequeras.	
	11- El sistema imprime las Chequeras.
12- El Analista escoge la opción de Imprimir el Listado de Cobros	
	13- El sistema imprime el Listado de Cobros.

Tabla 3.10.1.1.4 Caso de Uso Realizar Facturación.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Corregir Errores
Actores:	(Incluido)
Resumen:	El Caso de Uso comienza cuando se están corrigiendo los datos de cada una de las lecturas efectuadas por el Lector Cobrador, esta lectura se compara con la lectura del mes anterior y se determinan si presenta errores, se corrigen los errores encontrados, mostrando los errores encontrados.
Precondiciones:	Las lecturas efectuadas por el Lector – Cobrador ya se encuentran cargadas en el sistema. El Caso de Uso debe ser invocado
Referencias	R24, R25, R26, R27
Prioridad	

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
	1. Comprueba que no existan diferencias significativas entre la lectura del mes anterior y la lectura del mes actual(Flujo Alterno 1ª)
	2. Comprueba que después de una clave 1 se encuentre una clave 5(Flujo Alterno 2ª)
	3. Comprueba que la cantidad de claves 0 en una misma ruta no sea elevado.(Verificar cuando es elevado) (Flujo Alterno 3ª)

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	1ª Si la diferencia es significativa (averiguar cuando es significativa) se reporta el error

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	encontrado en la lectura. (Ir a actividad 2).
	2ª Si después de una clave 1 no aparece una clave 5, se reporta y se corrige el error encontrado en la lectura. (Ir a actividad 3).
	3ª En caso de que la cantidad de claves 0 en una misma ruta sea elevado, se reporta al Analista del Sistema.

Tabla 3.10.1.1.5 Caso de Uso Corregir Errores.

3.10.2 Proceso de Cobro

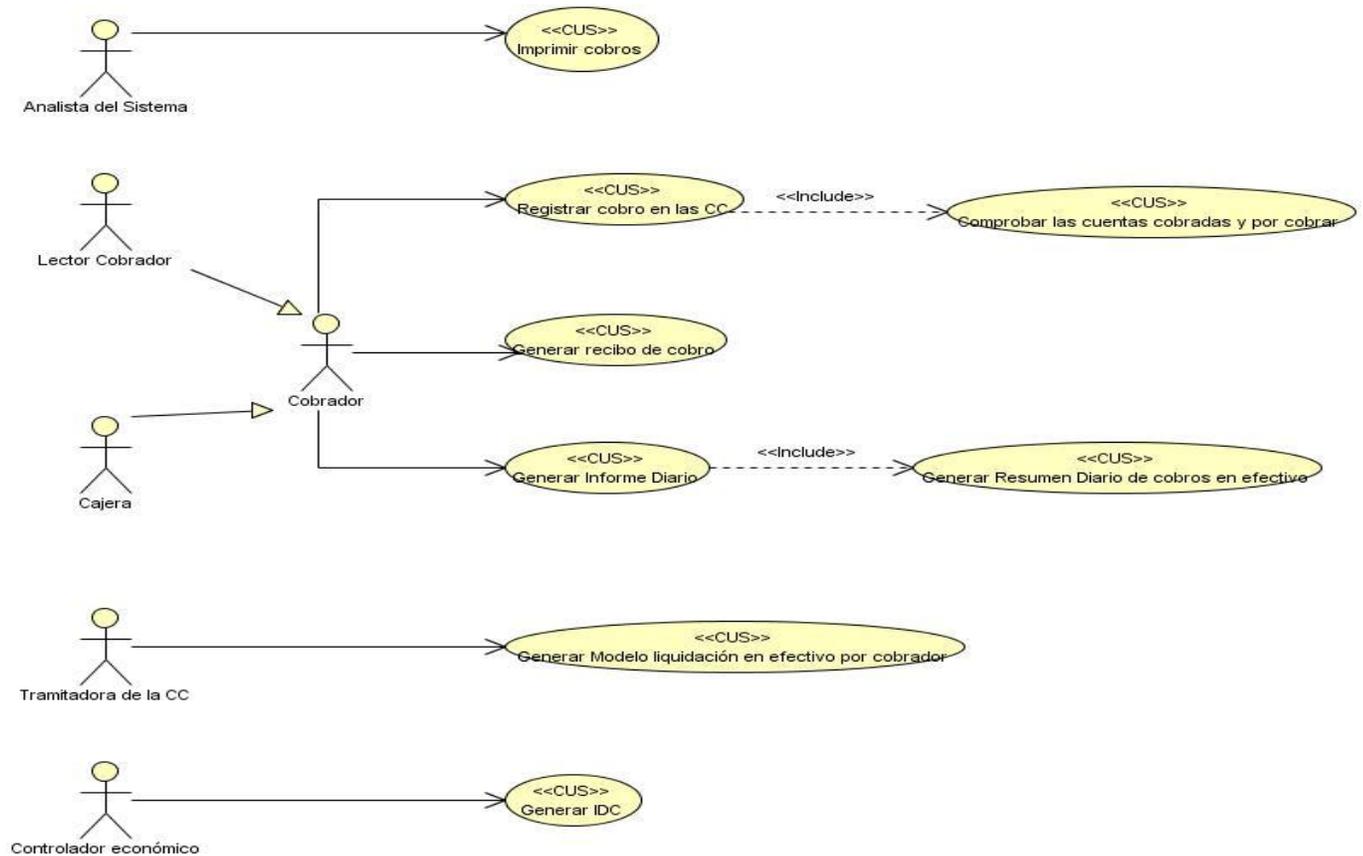


Figura 5 Diagrama de Caso de Uso del Sistema de Cobro.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Imprimir cobros
Actores:	Analista del Sistema.
Resumen:	Este Caso de Uso comienza cuando el Analista del Sistema manda a imprimir todos los talonarios de cobro realizados por el cobrador.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Analista del Sistema.
Referencias	R21
Prioridad	Crítico
Flujo Normal de Eventos	
	Respuesta del Sistema
1. El actor selecciona la opción "Imprimir talonario".	
	2. El sistema busca los clientes que han realizado el pago para imprimir los datos necesarios y su importe.
	3. El sistema muestra un mensaje con la opción de si desea realizar la impresión.
	4. Si acepta la opción, imprimir el cobro (En caso contrario ir a la sección 5ª).
	5. El sistema manda a imprimir los cobros de los clientes.
<i>Prototipo de Interfaz</i>	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	5ª No puede imprimirse el cobro realizado

Tabla 3.10.2.2.1 Caso de Uso Imprimir cobros.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Caso de Uso:	Generar Recibo de Cobro	
Actores:	Cajera	
Resumen:	El Caso de Uso comienza cuando la Cajera luego de haberse efectuado el cobro al cliente genera los Recibos de Cobro en la Casa Comercial que van a ser entregados al cliente que ha efectuado el pago en la misma.	
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Cajera.	
Referencias	R33	
Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- El actor selecciona la opción Generar Recibo de Cobro.		
2- El actor introduce los datos de los clientes que han pagado.		
	3- El sistema genera los Recibos de Cobro de los clientes a los cuales se les efectuó el cobro del consumo.	
	4- El sistema muestra los Recibos de Cobro generados.	
	5- El sistema brinda la opción de Imprimir los Recibos de Cobro.	
6- El actor selecciona la opción de Imprimir los Recibos de Cobro.		
	7- El sistema procede a realizar la impresión de los Recibos de Cobro	
	8- El sistema procede a guardar los Recibos de Cobro en un archivo.	

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
4.a El actor no desea imprimir los Recibos de Cobro en ese momento.(Ir a 6)	

Tabla 3.10.2.2.2 Caso de Uso Generar Recibo de Cobro.

Caso de Uso:	Generar Modelo Liquidación en Efectivo por Cobrador
Actores:	Tramitadora de la Casa Comercial.
Resumen:	El Caso de Uso comienza cuando la Tramitadora de la CC genera el Modelo de Liquidación del efectivo que le ha entregado cada cobrador al final del día.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Tramitadora de la Casa Comercial.
Referencias	R35
Prioridad	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El actor introduce la cantidad de efectivo entregado por el cobrador en el sistema.	2- El sistema valida que los datos sean correctos.
	3- El sistema almacena los datos introducidos por el actor.
4- Selecciona la opción "Imprimir modelo de liquidación en efectivo".	5- El sistema muestra la vista previa del modelo.
6- Selecciona la opción Aceptar.	

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	7- El sistema muestra el modelo de liquidación por cobrador.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
6- Selecciona la opción Cancelar	7- El sistema muestra un mensaje de alerta indicando que no se puede realizar la impresión.

Tabla 3.10.2.2.3 Caso de Uso Generar Modelo liquidación en efectivo por cobrador.

Caso de Uso:	Generar Informe Diario de Caja
Actores:	Controlador Económico
Resumen:	El caso de uso se inicia cuando al finalizar el día se le entrega al Controlador Económico todos los informes realizados por los cobradores y genera el Informe Diario de Caja para luego generar el reporte IDC.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Controlador Económico.
Referencias	R34
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El actor selecciona la opción generar IDC.	
2- El actor teclea todos los cobros que se han realizado en la caja.	
	3- El sistema crea los Informes Diarios de Caja.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

	4- El sistema envía los Informes Diarios de Caja.
	5- El sistema muestra el reporte generado de los Informes Diarios de Caja.

Tabla 3.10.2.2.4 Caso de Uso Generar IDC.

Según la metodología RUP en su disciplina de desarrollo, plantea que una vez comprendido el sistema, se da inicio a la próxima fase correspondiente al Análisis del Sistema. En esta etapa se analizan las exigencias que se describieron en la Captura de Requisitos; se presentan las clases que interactúan para realizar los distintos Casos de Uso, mediante el diagrama de clases del análisis y luego se realizan los diagramas de interacción, mediante los cuales se muestra una descripción gráfica de la interacción entre los actores y el sistema, además quedan reflejados los mensajes que se transmiten entre los objetos. Se concluye el capítulo con la validación de los Casos de Usos, los cuales se validan con la aplicación de métricas orientadas a objetos y con la aplicación de listas de chequeos..

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Los **artefactos** que se construyen son:

- **Modelo de análisis:** Contiene clases del análisis y sus objetos organizados en paquetes que colaboran.
- **Clases de análisis:** Se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Tienen atributos y entre ellas se establecen relaciones de asociación, agregación / composición, generalización / especialización y tipos asociativos. RUP propone clasificar a las clases en:

NOMBRE	CARACTERÍSTICAS	REPRESENTACIÓN
Entidad	Modelan información que posee larga vida y que es a menudo persistente.	 CE_Entidad
Interfaz	Modelan la interacción entre el sistema y sus actores.	 CI_Interfaz
Control	Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.	 CC_Controladora

3.11 Análisis del Sistema.

El objetivo de este flujo de trabajo es refinar y estructurar los requisitos obtenidos con anterioridad y traducirlos a una especificación que describe cómo implementar el sistema. El análisis consiste en obtener una visión del sistema y se preocupa por su funcionamiento, de modo que sólo se interesa por los requisitos funcionales.

3.11.1 Diagramas de clases del análisis y de interacción.

En la construcción del modelo de análisis se tienen que identificar las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas. Con esta información se construye el Diagrama de clases del análisis.

Las clases del análisis se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio.

Los diagramas de interacción (colaboración o secuencia) muestran las interacciones entre objetos mediante transferencia de mensajes entre estos o subsistemas.

3.12 Diagramas de clases del análisis

3.12.1 Modelo de Análisis

El modelo de análisis nos ayuda a refinar y estructurar los requisitos y nos proporciona una estructura centrada en el mantenimiento, en aspectos tales como la flexibilidad ante los cambios y la reutilización, además nos permite razonar sobre los aspectos internos del sistema, incluidos sus recursos compartidos internos. Se representa mediante un sistema de análisis que denota el paquete de más alto nivel del modelo. Es descrito con el lenguaje del desarrollador, estructurado por clases y paquetes estereotipados; proporciona la estructura a la vista interna, utilizado fundamentalmente para comprender cómo debería darse forma al sistema, es decir, cómo debería ser diseñado e implementado, no debe contener redundancias, inconsistencias, etc., entre requisitos, esboza como llevar a cabo la funcionalidad dentro del sistema, incluida la funcionalidad significativa para la arquitectura; sirve como una primera aproximación al diseño, define realizaciones de casos de uso, y cada una de ellas representa el análisis de un caso de uso

del modelo de casos de usos. [Tomado de “JACOBSON, Ivar, BOOCH, Grady, RUMBAUGH, James. El proceso unificado de desarrollo de software, Pearson Educación S.A., 2000”.]

3.13 Diagrama de Interacción.

Un diagrama de interacción explica gráficamente las interacciones existentes entre las instancias (y las clases) del modelo de éstas.

El UML define dos tipos de estos diagramas; ambos sirven para expresar interacciones semejantes o idénticas del mensaje:

– **Diagramas de secuencia.**

– **Diagramas de colaboración.**

3.13.1 Diagrama de Secuencia.

Un diagrama de secuencia muestra un conjunto de mensajes, dispuestos en una secuencia temporal. Cada rol en la secuencia se muestra como una línea de vida, es decir, una línea vertical que representa el rol durante cierto plazo de tiempo, con la interacción completa. Los mensajes se muestran como flechas entre las líneas de vida. Un diagrama de secuencia puede mostrar un escenario, es decir, una historia individual de una transacción. Un uso de un diagrama de secuencia es mostrar la secuencia del comportamiento de un caso del uso. Cuando está implementado el comportamiento, cada mensaje en un diagrama de secuencia corresponde a una operación en una clase, a un evento disparador, o a una transición en una máquina de estados. (21)

3.13.2 Diagrama de Colaboración.

Modela los objetos y los enlaces significativos dentro de una interacción. Los objetos y los enlaces son significativos solamente en el contexto proporcionado por la interacción. Un rol describe un objeto, y un rol en la asociación describe un enlace dentro de una colaboración. Un diagrama de colaboración muestra los roles en la interacción en una disposición geométrica. Los mensajes se muestran como flechas, ligadas a las líneas de la relación, que conectan a los roles. La secuencia de mensajes, se indica con los números secuenciales que preceden a las descripciones del mensaje. Un uso de un diagrama de colaboración es mostrar la implementación de una operación. La colaboración muestra los parámetros y las variables locales de la operación, así como asociaciones más permanentes. Cuando se implementa el

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

comportamiento, la secuencia de los mensajes corresponde a la estructura de llamadas anidadas y el paso de señales del programa. (21)

3.14 Diagramas de colaboración de algunos Casos de Uso.

A continuación se mostrarán los diagramas de colaboración de algunas clases del análisis, y en el anexo podrán encontrarse otros.

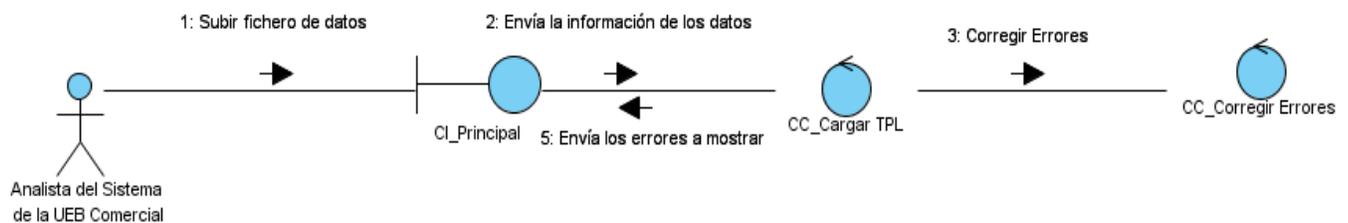


Figura 3. Caso de Uso Cargar Datos a facturar.

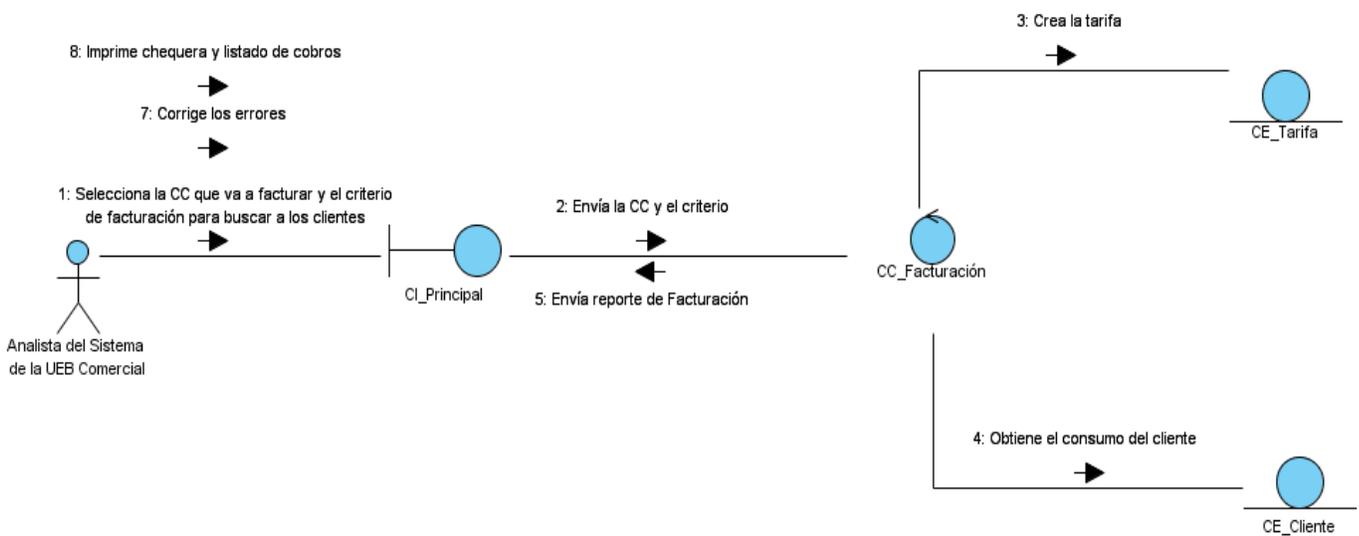


Figura 4. Caso de Uso Realizar Facturación

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

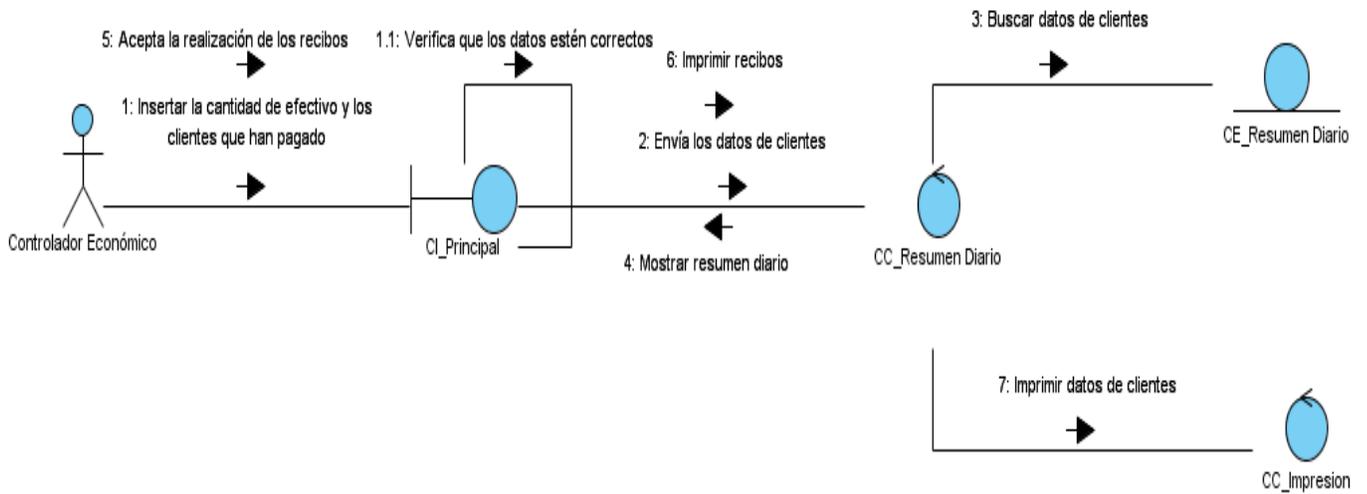


Figura 5. Caso de Uso Generar resumen Diario de cobros en efectivo

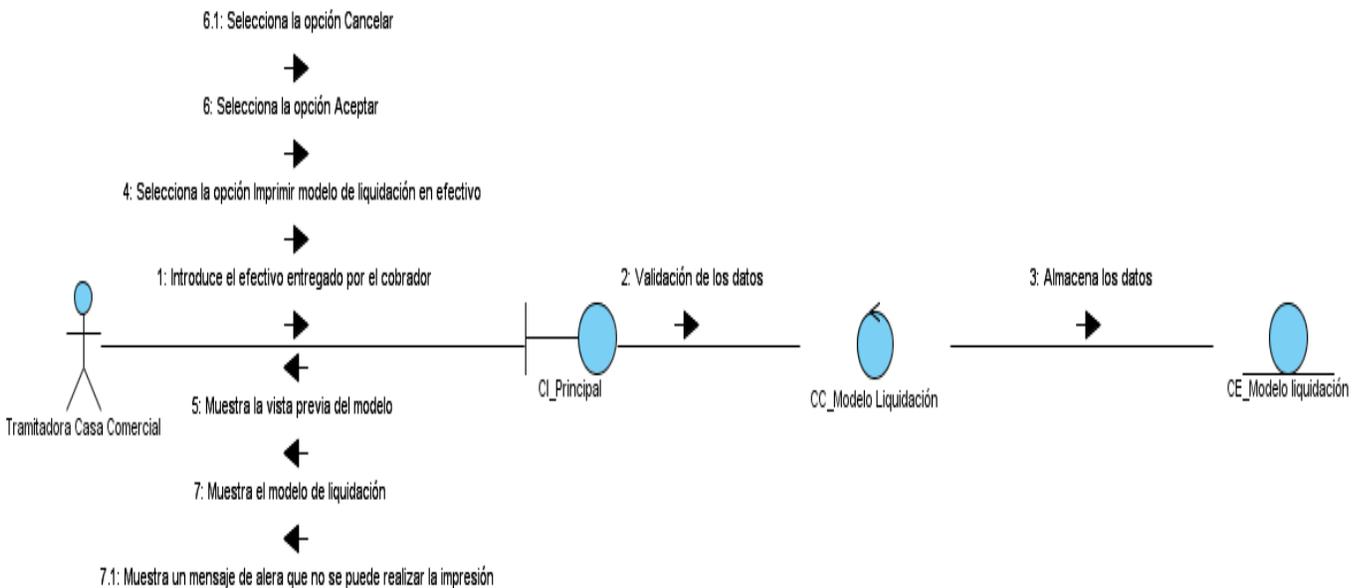


Figura 6. Caso de Uso Generar Modelo liquidación en efectivo por cobrador

3.15 Diagramas de Clases del Diseño.

La definición de este tipo de diagramas se lleva a cabo en la fase de diseño del ciclo de desarrollo. Su preparación exige crear antes: (22)

- Diagramas de interacción: a partir de ellos el diseñador identifica las clases de software que intervienen en la solución, así como los métodos de las clases.
- Modelo conceptual: a partir de éste el diseñador agrega detalles a la definición de las clases.

El diagrama de clases del diseño describe gráficamente las especificaciones de las clases del software y de las interfaces en una aplicación. Normalmente contiene la siguiente información: (22)

- clases, asociaciones y atributos
- interfaces, con sus operaciones y constantes
- métodos
- información sobre los tipos de los atributos
- navegabilidad
- dependencias

Extensiones para diseños Web

Debido a que la solución informática a realizar que se plantea es una aplicación web se decidió que la realización de este modelado sea a través de estereotipos web.



Server Page: la página Web que tiene código que se ejecuta en el servidor.



Client Page: una instancia de Página Cliente es una página Web, con formato HTML.



Form: colección de elementos de entrada que son parte de una página cliente.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Descripciones de las clases genéricas para los Casos de Uso

Nombre: Layout
Tipo de clase: Clase Vista
Descripción:
Clase relacionada con el framework Symfony, utilizada como plantilla para algunas operaciones dentro y fuera de la misma.

Nombre: Adicionar Success
Tipo de clase: Página servidora
Descripción:
Página servidora encargada de la construcción de la interfaz a través de la implementación de los métodos.

Nombre: adicionarSuccess
Tipo de clase: Página Cliente
Descripción:
Clase construida por la página servidora y encargada de la realización del formulario donde se mostrarán los datos pertenecientes a cada Caso de Uso.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Nombre: Modelo
Tipo de clase: Clase Modelo
Descripción:
Clase encargada de la lógica para la interacción con la BD así como las clases que necesita el sistema para la gestión de la información en la BD, donde están los principales métodos de consulta.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

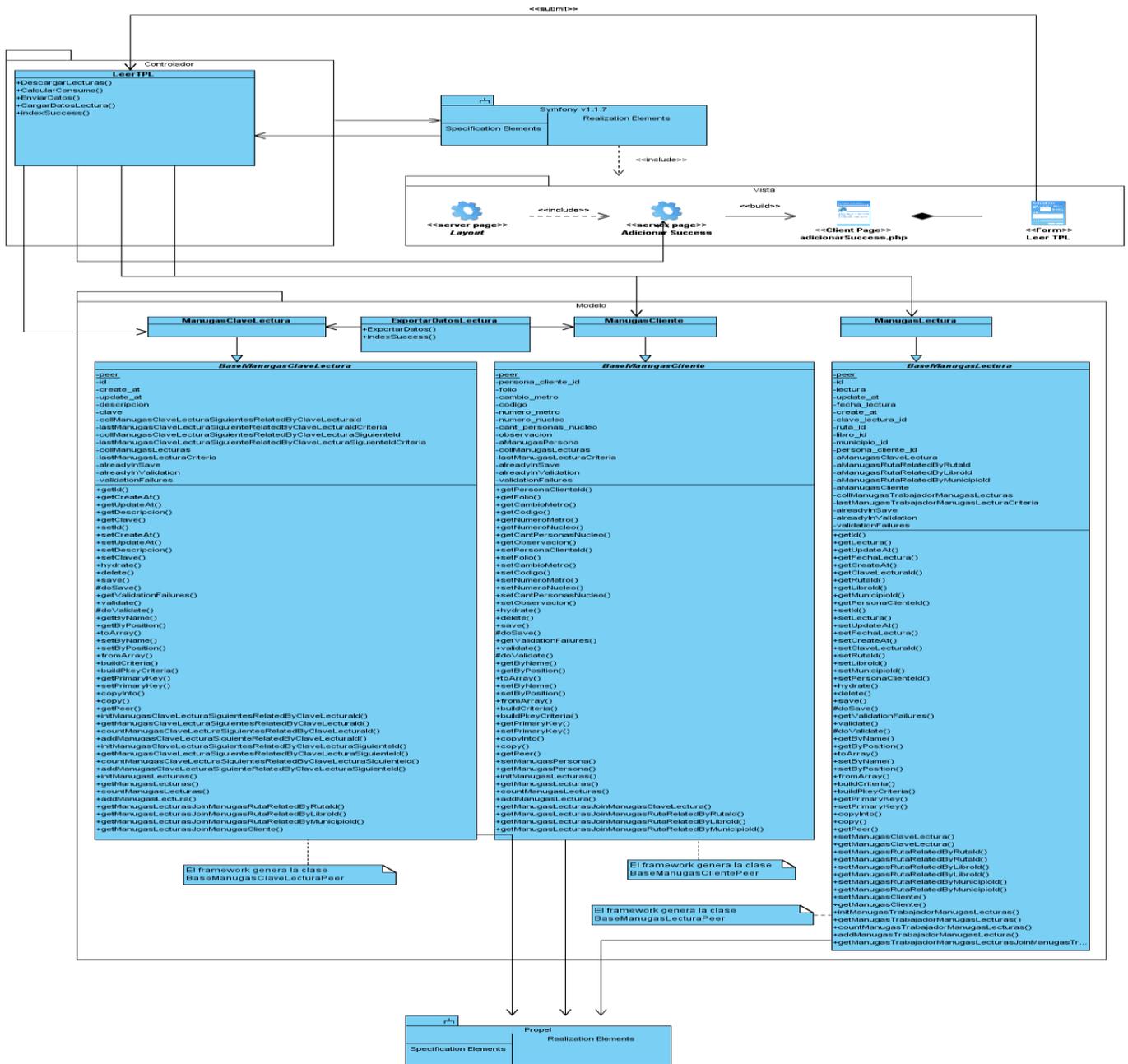


Figura 7. Caso de Uso Leer TPL

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Nombre: Adicionar Success
Tipo de clase: Página servidora
Descripción:
Página servidora encargada de la construcción de la interfaz.

Nombre: LeerTPL	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	Descargar Lecturas()
Descripción:	Descarga las lecturas del TPL para el sistema.
Nombre:	CargarDatosLectura()
Descripción:	Carga los datos de las lecturas que se encuentran en el sistema para el TPL

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Nombre: Modelo
Tipo de clase: Clase Modelo
Descripción:
Clase encargada de la lógica para la interacción con la BD así como las clases que necesita el sistema para la gestión de la información en la BD, donde están los principales métodos de consulta.

Nombre: Modelo	
Tipo de clase: Modelo	
Para cada clase:	
Nombre:	BaseManugasLectura
Descripción:	Clase que contiene la información de las lecturas que se realizan a los clientes en el mes.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

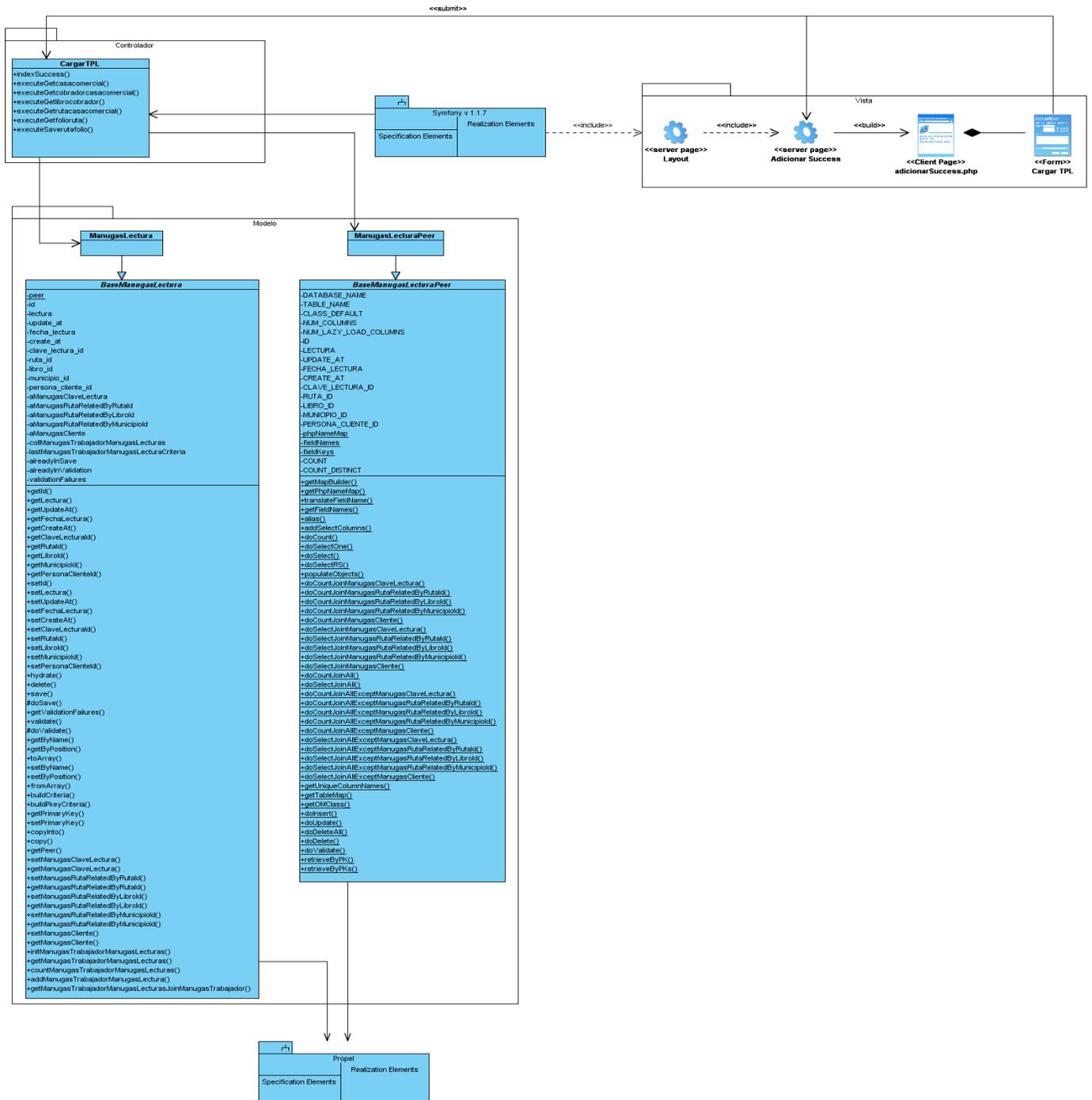


Figura 8. Caso de Uso Cargar TPL

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Nombre: Adicionar Success
Tipo de clase: Página servidora
Descripción:
Página servidora encargada de la construcción de la interfaz.

Nombre: CargarTPL	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	executeGetfolioruta()
Descripción:	Obtiene todas las rutas de un folio.
Nombre:	executeSaverutafolio()
Descripción:	Salva en un fichero todas las rutas y folios para cargarlos en el TPL

Nombre: Modelo
Tipo de clase: Clase Modelo
Descripción:
Clase encargada de la lógica para la interacción con la BD así como las clases que necesita el sistema para la gestión de la información en la BD, donde están los principales métodos de consulta.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

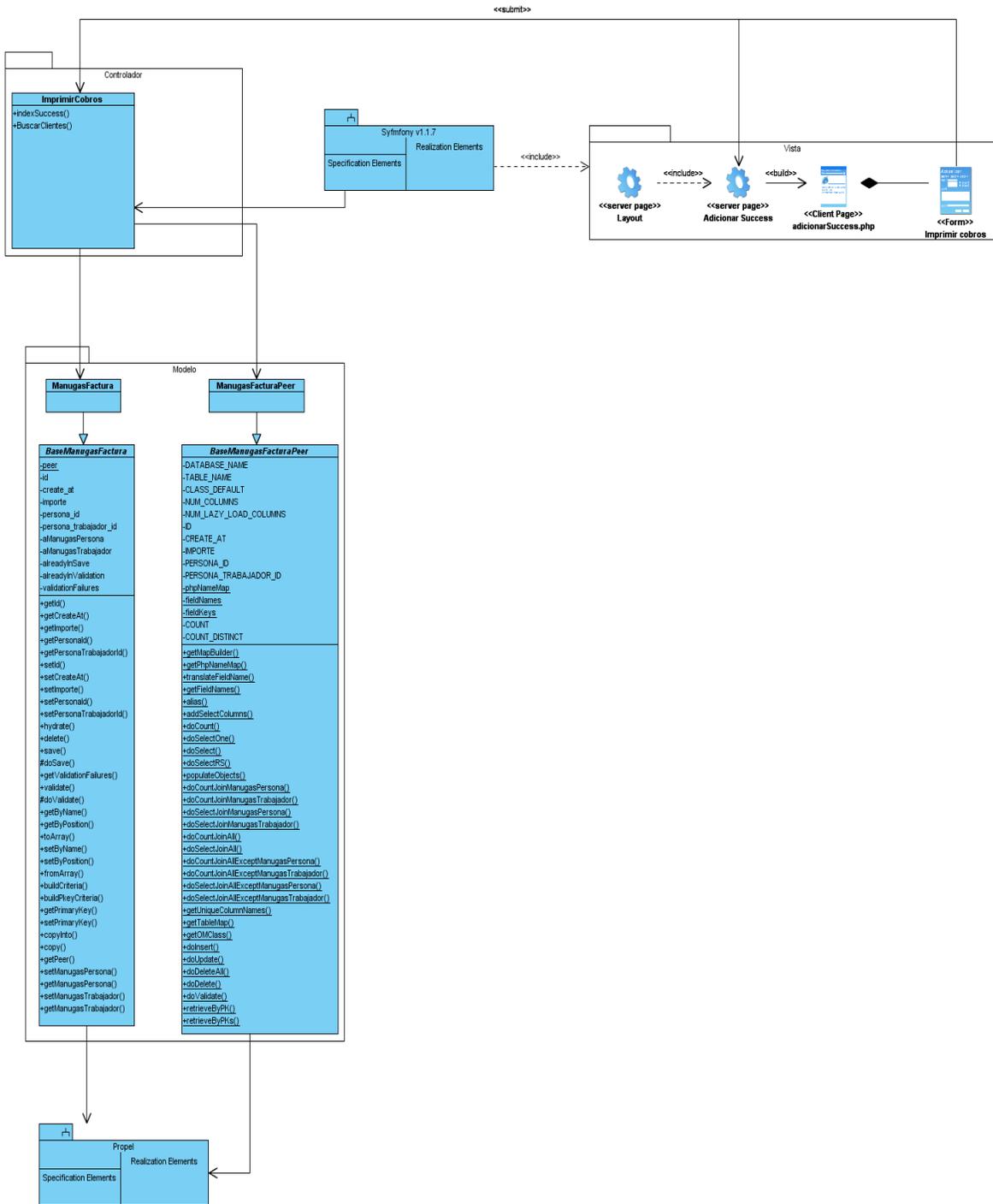


Figura 9. Caso de Imprimir Cobros.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Nombre: ImprimirCobros	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	BuscarCliente()
Descripción:	Busca al cliente que necesita para efectuarle el cobro.

Nombre: Modelo	
Tipo de clase: Modelo	
Para cada Clase:	
Nombre:	BaseManugasFactura
Descripción:	Clase encargada de manejar la información de los datos de las facturas que se les realizan a los clientes mensualmente.

3.16 Patrones

¿Qué es un Patrón?

Un patrón es un par problema/solución con nombre que se puede aplicar en nuevos contextos, con consejos sobre cómo aplicarlo en nuevas situaciones, o sea, un patrón es una descripción de un problema bien conocido que suele incluir: descripción, escenario de uso, solución concreta, consecuencias de utilizar el patrón, ejemplos de implementación y lista de patrones relacionados. (Larman, 2008)

3.16.1 Patrones de Diseño

Un patrón de diseño es una descripción de clases y objetos, comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular. Identifica clases, instancias, roles, colaboraciones y la distribución de responsabilidades. (Gracia, 2005)

Un patrón de diseño es:

- Una solución estándar para un problema común de programación.
- Una técnica para flexibilizar el código.
- Una manera más práctica de describir ciertos aspectos de la organización de un programa.
- Conexiones entre componentes de programas.

Características de los patrones de diseño:

- Son soluciones concretas y técnicas.
- Se utilizan en situaciones frecuentes.
- Favorecen la reutilización de código.
- Es difícil reutilizar la implementación de un patrón.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. (22)

Experto: Es un patrón que se usa más que cualquier otro al asignar responsabilidades en un principio básico que suele utilizarse en el diseño orientado a objetos. Con él no se pretende designar una idea oscura ni extraña; expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen.

Creador: El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que debemos conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento.

Bajo Acoplamiento: El Bajo Acoplamiento es un principio que debemos recordar durante las decisiones de diseño: es la meta principal que es preciso tener presente siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño.

Alta Cohesión: Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme.

Controlador: Este patrón ofrece una guía para tomar decisiones apropiadas que generalmente se aceptan. La misma clase controlador debería utilizarse con todos los eventos sistémicos de un caso de uso, de modo que podamos conservar la información referente al estado del caso.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Los patrones que se utilizaron para la realización de los diagramas de las clases del diseño fueron los siguientes.

Creador:

En la clase Actions se encuentran las acciones definidas para los procesos de facturación y cobro de la empresa de Gas Manufacturado y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Actions es "creador" de dichas entidades.

Experto:

Es un patrón que se usa más que cualquier otro al asignar responsabilidades. Con él se pretende designar simplemente la "intuición" de que los objetos hacen las tareas relacionadas con la información que poseen.

Alta Cohesión:

Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las properties, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.

Controlador:

Todas las peticiones Web son manejadas por un solo controlador frontal (sfActions), que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Los patrones GoF, descritos en el libro Design Patterns y conocidos como “Gang of Four” (Pandilla de los Cuatro), son un conjunto de 23 patrones de diseños de gran utilidad agrupados en tres tipos fundamentales; patrones de creación, patrones estructurales y patrones de comportamiento. (22)

1. Creacionales.

- ✓ Abstract Factory.
- ✓ Builder.
- ✓ Factory Method.
- ✓ Prototype.
- ✓ Singleton.

2. Estructurales.

- ✓ Adapter.
- ✓ Bridge.
- ✓ Composite.
- ✓ Decorator.
- ✓ Facade.
- ✓ Flyweight.
- ✓ Proxy.

3. De Comportamiento.

- ✓ Chain of Responsibility.
- ✓ Command.
- ✓ Interpreter.
- ✓ Iterator.
- ✓ Mediator.
- ✓ Memento.
- ✓ Observer.
- ✓ State.
- ✓ Strategy.
- ✓ Template Method.
- ✓ Visitor.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Dentro de los patrones GoF que se utilizaron se tiene:

Creacionales:

Singleton (Instancia única):

Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el controlador frontal hay una llamada a `sfContext: getInstance ()`. En una acción, el método `getContext ()`, un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony.

Abstract Factory (Fábrica abstracta):

Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando. Cuando el framework necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea.

Como patrón arquitectónico Modelo Vista Controlador (MVC) que pertenece al estilo arquitectónico llamada y retorno, el cual dentro de algunas de sus ventajas cuenta con la de separar en 3 capas las diversas responsabilidades que presenta una aplicación Web como son la lógica del negocio, la capa de acceso a Datos y la Presentación.

CAPÍTULO 3: Análisis y Diseño de la solución propuesta.

Estructurales:

Decorator (Envoltorio):

Añade funcionalidad a una clase, dinámicamente. El archivo layout.php, que también se denomina plantilla global, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. Este comportamiento es una implementación del patrón de diseño llamado.

Composite (Objeto compuesto):

Permite tratar objetos compuestos como si de uno simple se tratara. Sirve para construir objetos complejos a partir de otros más simples y similares entre sí, gracias a la composición recursiva y a una estructura en forma de árbol. Esto simplifica el tratamiento de los objetos creados, ya que al poseer todos ellos una interfaz común, se tratan todos de la misma manera.

CONCLUSIONES

Con la realización del presente trabajo se arriba a las siguientes Conclusiones Generales:

- La utilización de RUP, como metodología de desarrollo de Software, permitió obtener los artefactos necesarios para elaborar el análisis y modelado de la Solución Informática correspondiente a los Procesos de Facturación y Cobro en la Empresa de Gas Manufacturado. Su particularidad de ser un Proceso iterativo – incremental, facilitó la organización, obtención y perfeccionamiento de los mismos. Además, al seguir el orden de dichas Fases y la continuidad de cada artefacto mediante los ciclos de desarrollo, se ha logrado una mayor calidad en el resultado final de los mismos.
- Con el estudio del Negocio se pudo identificar las actividades correspondientes a los Procesos de Facturación y Cobro en la Empresa de Gas Manufacturado, las fases por donde debe transitar el documento, se logró un mejor entendimiento del entorno donde deberá implantarse el Sistema, así como la determinación de las Reglas del Negocio, que resultan un conjunto de precondiciones a cumplir.
- Se logró a partir del intercambio con los clientes y el entendimiento del Negocio, la definición de los Requerimientos Funcionales y No Funcionales de la Solución Informática, garantizando el cumplimiento de las necesidades del cliente.
- Se logró una propuesta de Análisis y Diseño para los Procesos de Facturación y Cobro en la Empresa de Gas Manufacturado, que constituye una base para el trabajo de los diseñadores e implementadores del Sistema, que se encargarán de continuar los próximos ciclos definidos por RUP.

Bibliografía referenciada

1. **González, José Angel Álvarez.** *Introducción a la industria petrolera*. Ciudad de La Habana : s.n., 2007.
2. **Gasur.** *Gasur*. [En línea] AENOR, 12 de febrero de 2009. [Citado el: 14 de febrero de 2009.] <http://www.gasur.com.uy/marcos/nuestraempresa.html>.
3. **Mayoral, María Julia.** *CubAhora*. [En línea] Centro de Información para la Prensa, 29 de diciembre de 2005. [Citado el: 21 de marzo de 2009.] http://www.cubahora.cu/index.php?tpl=buscar/ver-not_buscar.tpl.html&newsid_obj_id=597.
4. **Maturana, Sergio.** *Gescoop*. [En línea] 12 de febrero de 2009. [Citado el: 14 de febrero de 2009.] http://www2.ing.puc.cl/gescopp/Sergio_Maturana/SAG/ClaseSCM.pdf.
5. **BETSIME, Revista.** *Revista BETSIME*. [En línea] Enero de 2006. [Citado el: 10 de Febrero de 2009.] http://www.betsime.disaic.cu/secciones/eco_enemar_07.htm.
6. **Rasson, Damien.** *softonic. softonic*. [En línea] 3 de diciembre de 2008. [Citado el: 14 de febrero de 2009.] <http://express-invoice-windows.softonic.com/>.
7. **Navarro, David.** *softonic. softonic*. [En línea] 2 de 2008 de octubre. [Citado el: 14 de febrero de 2009.] <http://zfactura.softonic.com/>.
8. **Santos, Elena.** *softonic. softonic*. [En línea] 18 de octubre de 2006. [Citado el: 14 de febrero de 2009.] <http://pagares-y-gestion-de-cobros.softonic.com/>.
9. **Adobe.** *adobe*. [En línea] 19 de marzo de 2009. [Citado el: 20 de marzo de 2009.] http://livedocs.adobe.com/dreamweaver/8_es/using/wwhelp/wwhimpl/common/html/wwhelp.htm?context=LiveDocs_Parts&file=gs_12_u5.htm.
10. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *"El Proceso Unificado de Desarrollo de Software"*. Madrid, España : Addison Wesley, 2000. ISBN 84-7829-036-2.
11. **Schmuller, Joseph.** *"Aprendiendo UML en 24 horas"*. México : Pearson Educacion, 2000. ISBN: 968-444-63-X.
12. **Gracia, Joaquin.** *IngenieroSoftware. UML: Diagramas UML. ¿Qué es UML?* [En línea] 7 de Mayo de 2005. [Citado el: 17 de Enero de 2009.] <http://www.ingenierosoftware.com/analisisydiseno/uml.php>.
13. **ScRum, Proyecto.** *Proyecto ScRum*. [En línea] diciembre de 2008. [Citado el: 15 de febrero de 2009.] <http://oeguzman.googlepages.com/3metodologiasagilesdesoftware>.

14. **extrema, PX programación.** *PX programación extrema*. [En línea] 22 de diciembre de 2007. [Citado el: 4 de marzo de 2009.] <http://www.programacionextrema.org/>.
15. **Penadés, Letelier & willydev.** *willydev*. [En línea] diciembre de 2008. [Citado el: 4 de marzo de 2009.] <http://www.willydev.net/descargas/masyxp.pdf>.
16. **engineering, baufest software.** *baufest software engineering*. [En línea] 1991-2003. [Citado el: 4 de marzo de 2009.] http://www.baufest.com/spanish/scrum/scrumconference2006/Que_es_scrum.pdf.
17. **Palacio, Juan.** *Navegapolis.net*. [En línea] 2006. [Citado el: 4 de marzo de 2009.] <http://www.navegapolis.net/>.
18. **Informáticas, LaPepa.EU Desarrollo de Aplicaciones.** *LaPepa.EU Desarrollo de Aplicaciones Informáticas*. [En línea] 2007. [Citado el: 26 de febrero de 2009.] <http://lapepa.eu/dai/contenidos/4gl06/>.
19. **Manrique, Jorge.** IES Jorge Manrique. [En línea] diciembre de 2008. [Citado el: 18 de Enero de 2009.] <http://www.iesjorgemanrique.es/hda/descargas/>.
20. **MeRinde.** *MeRinde*. [En línea] GNU FDL, noviembre de 2008. [Citado el: 22 de febrero de 2009.] http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=137&Itemid=192.
21. El Lenguaje Unificado de Modelado. Manual de Referencia. [aut. libro] Ivar Jacobson, Grady Booch James Rumbaugh. *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Adison Wesley.
22. **Larman, Craig.** UML y Patrones. *UML y Patrones*. La Habana : Félix Varela, 2004.

Bibliografía Consultada

1. **Adobe.** adobe. [En línea] 19 de marzo de 2009. [Citado el: 20 de marzo de 2009.] http://livedocs.adobe.com/dreamweaver/8_es/using/wwhelp/wwhimpl/common/html/wwhelp.htm?context=Livedocs_Parts&file=gs_12_u5.htm.
2. **baufest** software engineering. baufest software engineering. [En línea] 1991-2003. [Citado el: 4 de marzo de 2009.] http://www.baufest.com/spanish/scrum/scrumconference2006/Que_es_scrum.pdf.
3. **BETSIME**, Revista. Revista BETSIME. [En línea] Enero de 2006. [Citado el: 10 de Febrero de 2009.] http://www.betsime.disaic.cu/secciones/eco_enemar_07.htm.
4. **Corba**, OMG. 2000. Corba. [En línea] 2009. [Citado el: 10 de Febrero de 2009.] <http://www.corba.org/>
5. **El Lenguaje Unificado de Modelado. Manual de Referencia.** [aut. libro] Ivar Jacobson, Grady Booch James Rumbaugh. El Lenguaje Unificado de Modelado. Manual de Referencia. s.l.: Adison Wesley.
6. **extrema, PX programación.** PX programación extrema. [En línea] 22 de diciembre de 2007. [Citado el: 4 de marzo de 2009.] <http://www.programacionextrema.org/>.
7. **Fabien Potencier, François Zaninotto.** librosweb. librosweb. [En línea] 13 de julio de 2008. [Citado el: 14 de febrero de 2009.] <http://www.librosweb.es/>.
8. **García, Yadira.** tvsantiago. tvsantiago. [En línea] 18 de marzo de 2009 <http://www.tvsantiago.co.cu/?mod=noticias&id=14887>
9. **Gasur.** Gasur. [En línea] AENOR, 12 de febrero de 2009. [Citado el: 14 de febrero de 2009.] <http://www.gasur.com.uy/marcos/nuestraempresa.html>.
10. **González, José Angel Álvarez.** Introducción a la industria petrolera. Ciudad de La Habana: s.n., 2007.
11. **Gracia, Joaquin.** IngenieroSoftware. UML: Diagramas UML. ¿Qué es UML? [En línea] 7 de Mayo de 2005. [Citado el: 17 de Enero de 2009.] <http://www.ingenierosoftware.com/analisisydiseno/uml.php>.
12. **Informáticas, LaPepa.** EU Desarrollo de Aplicaciones. LaPepa.EU Desarrollo de Aplicaciones Informáticas. [En línea] 2007. [Citado el: 26 de febrero de 2009.] <http://lapepa.eu/dai/contenidos/4gl06/>.
13. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** "El Proceso Unificado de Desarrollo de Software". Madrid, España: Addison Wesley, 2000. ISBN 84-7829-036-2.

14. **Jiménez**, MSc. Hugo F. Arboleda. 2005. Modelos de ciclo de vida en desarrollo de software. Asociación Colombiana de Ingenieros de Sistemas (ACIS). [Online] Enero 2009. <http://www.acis.org.co/index.php?id=551>.
15. **Larman**, Craig. UML y Patrones. Introducción al análisis y diseño orientado a objetos. 1999
16. **Letelier**, P. and M.C. Penadés, Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP).
17. **Maturana**, Sergio. Gescoop. [En línea] 12 de febrero de 2009. [Citado el: 14 de febrero de 2009.] http://www2.ing.puc.cl/gescopp/Sergio_Maturana/SAG/ClaseSCM.pdf.
18. **Mayoral**, María Julia. CubAhora. [En línea] Centro de Información para la Prensa, 29 de diciembre de 2005. [Citado el: 21 de marzo de 2009.] http://www.cubahora.cu/index.php?tpl=buscar/ver-not_buscar.tpl.html&newsid_obj_id=597.
19. **Manrique**, Jorge. IES Jorge Manrique. [En línea] diciembre de 2008. [Citado el: 18 de Enero de 2009.] <http://www.iesjorgemanrique.es/hda/descargas/>.
20. MeRinde. MeRinde. [En línea] GNU FDL, noviembre de 2008. [Citado el: 22 de febrero de 2009.] http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=137&Itemid=192.
21. Navarro, David. softonic. softonic. [En línea] 2 de 2008 de octubre. [Citado el: 14 de febrero de 2009.] <http://zfactura.softonic.com/>.
22. **NNOVA**, G.d.S. Rational Rose Enterprise [disponible en: <http://www.rational.com.ar/herramientas/roseenterprise.html>.]
23. **Palacio**, Juan. Navegapolis.net. Navegapolis.net. [En línea] 2006. [Citado el: 4 de marzo de 2009.] <http://www.navegapolis.net/>.
24. **PDF**. <http://www.pdf-search-engine.com>. <http://www.pdf-search-engine.com>. [En línea] Google, 2008. [Citado el: 29 de marzo de 2009.] http://www.ingenieria.cl/escuelas/informatica/apuntes_curso_uml/DIAGRAMA%20DE%20SECUENCIASinfondo.pdf.
25. **Penadés, Letelier & willydev**. [En línea] diciembre de 2008. [Citado el: 4 de marzo de 2009.] <http://www.willydev.net/descargas/masyxp.pdf>.
26. **Pressman, Roger S**. Ingeniería del Software. Un enfoque Práctico.

27. **Rasson, Damien.** softonic. softonic. [En línea] 3 de diciembre de 2008. [Citado el: 14 de febrero de 2009.] <http://express-invoice-windows.softonic.com/>.
28. **Sanchez, María A. Mendoza.** willydev. willydev. [En línea] 2002 de Noviembre de 2002. [Citado el: 6 de Febrero de 2009.] <http://www.willydev.net/descargas/cualmetodologia.pdf>.
29. **Santos, Elena.** softonic. softonic. [En línea] 18 de octubre de 2006. [Citado el: 14 de febrero de 2009.] <http://pagares-y-gestion-de-cobros.softonic.com/>.
30. **Schmuller, Joseph.** "Aprendiendo UML en 24 horas". México : Pearson Educacion, 2000. ISBN: 968-444-63-X.
31. **ScRum, Proyecto.** Proyecto ScRum. [En línea] diciembre de 2008. [Citado el: 15 de febrero de 2009.] <http://oeguzman.googlepages.com/3metodologiasagilesdesoftware>.

Anexos

Capítulo 3: Características del Sistema.

Anexo 3.1

Caso de Uso:	Cargar Datos a Facturar	
Actores:	Analista del Sistema de la UEB Comercial	
Resumen:	El caso de uso comienza cuando el analista selecciona la opción de cargar los datos que fueron enviados por la Tramitadora de la Casa Comercial, se actualiza los datos en la Base de Datos, finaliza el caso de uso.	
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el rol de analista del sistema.	
Referencias	R11, R12	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “”		
Acción del Actor	Respuesta del Negocio	
1- El Analista del Sistema escoge la opción de cargar lecturas y consumo desde el fichero enviado por la Tramitadora de la Casa Comercial, dando click en el botón de Examinar		
	2- El sistema muestra una ventana pidiendo la localización del fichero que se desea cargar.	
3- El Analista del Sistema busca la dirección donde se encuentra el fichero que desea cargar, confirmándolo dando click en el botón de		

Aceptar	
	4- El sistema carga los datos de todos los clientes que se encuentran en el fichero(lectura, consumo)
	5- El sistema invoca al <u>CU Corregir errores</u>
	6- El sistema actualiza en la Base de Datos de SISMET los datos de los clientes.

Tabla A3.5 Descripción del Caso de Uso Cargar Datos a Facturar.

Anexo 3.2

Caso de Uso:	Exportar Datos de Lectura
Actores:	Tramitadora Casa Comercial
Resumen:	El caso de uso comienza cuando la Tramitadora Casa Comercial exporta los datos de las lecturas.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el rol de Tramitadora Casa Comercial.
Referencias	R29
Prioridad	Secundario

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
1. La Tramitadora de la Casa Comercial escoge la opción de exportar la lectura y el consumo de cada cliente hacia un fichero.	

	2. El sistema exporta los datos hacia un fichero.
3. La Tramitadora de la Casa Comercial envía el fichero con los datos al Analista del Sistema de la UEB Comercial.	

Tabla A3.6 Descripción del Caso de Uso Exportar Datos de Lectura.

Anexo 3.3

Caso de Uso:	Imprimir cobros
Actores:	Analista del Sistema.
Resumen:	Este caso de uso comienza cuando el Analista del Sistema manda a imprimir todos los talonarios de cobro realizados por el cobrador.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Analista del Sistema.
Referencias	R21
Prioridad	Secundario
Flujo Normal de Eventos	
Sección "Imprimir cobros"	
	Respuesta del Sistema
1. El actor selecciona de la lista de clientes, el cliente que ha realizado el pago.	
2. El actor selecciona la opción "Imprimir".	
	3. El sistema busca los clientes que han realizado el pago para imprimir los datos

	necesarios y su importe.
	4. El sistema muestra un mensaje con la opción de si desea realizar la impresión.
	5. Si acepta la opción, imprimir el cobro (En caso contrario ir a la sección 5ª).
	6. El sistema manda a imprimir los cobros de los clientes.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	5ª No puede imprimirse el cobro realizado

Tabla A3.7 Descripción del Caso de Imprimir cobros.

Anexo 3.4

Caso de Uso:	Registrar cobro en las CC.
Actores:	Cajera
Resumen:	El caso de uso comienza cuando el cobrador realiza el cobro al cliente que se presenta en la Casa Comercial para efectuar el pago del servicio según la tarifa registrada en el mes.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Cajera.
Referencias	R32
Prioridad	Secundario
Flujo Normal de Eventos	
Sección “”	

Acción del Actor	Respuesta del Sistema
1. El actor selecciona la opción "Registrar cobro".	
2. El actor inserta los datos del cliente para luego efectuarle el cobro del consumo perteneciente a la lectura del mes.	
3. El actor selecciona la opción "Insertar cliente".	4. El sistema busca en la Base de Datos para ver si el cliente existe. (En caso contrario ir a la sección 3ª).
	5. Invoca al <u>Caso de Buscar Cliente</u> .
6. El actor efectúa el cobro al cliente y selecciona la opción "Adicionar cobro".	
	7. El sistema guarda los datos insertados.
	8. El sistema muestra un mensaje indicando que el cobro se ha insertado correctamente.
Acción del Actor	Respuesta del Sistema
	3ª Muestra un mensaje indicando que el cliente no existe.

Tabla A3.8 Descripción del Caso de Registrar cobro en las CC.

Anexo 3.5

Caso de Uso:	Generar Recibo de Cobro
Actores:	Cajera
Resumen:	El caso de uso comienza cuando la Cajera luego de haberse efectuado el cobro al cliente genera los recibos de cobro en la Casa Comercial que van a ser entregados al cliente que ha efectuado el pago en la misma.
Precondiciones:	
Referencias	R33
Prioridad	Crítico

Flujo Normal de Eventos

Sección “”

Acción del Actor	Respuesta del Sistema
1- El actor selecciona la opción Generar Recibo de Cobro.	
2- El actor introduce los datos de los clientes que han pagado.	
	3- El sistema genera los recibos de cobro de los clientes a los cuales se les efectuó el cobro del consumo.
	4- El sistema muestra los recibos de cobro generados.
	5- El sistema brinda la opción de imprimir los recibos de cobro.
6- El actor selecciona la opción de imprimir los recibos de cobro.	

	7- El sistema procede a realizar la impresión de los recibos de cobro
	8- El sistema procede a guardar los recibos de cobro en un archivo.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
4.a El actor no desea imprimir los recibos de cobro en ese momento.(Ir a 6)	

Tabla A3.9 Descripción del Caso de Uso Generar Recibo de Cobro.

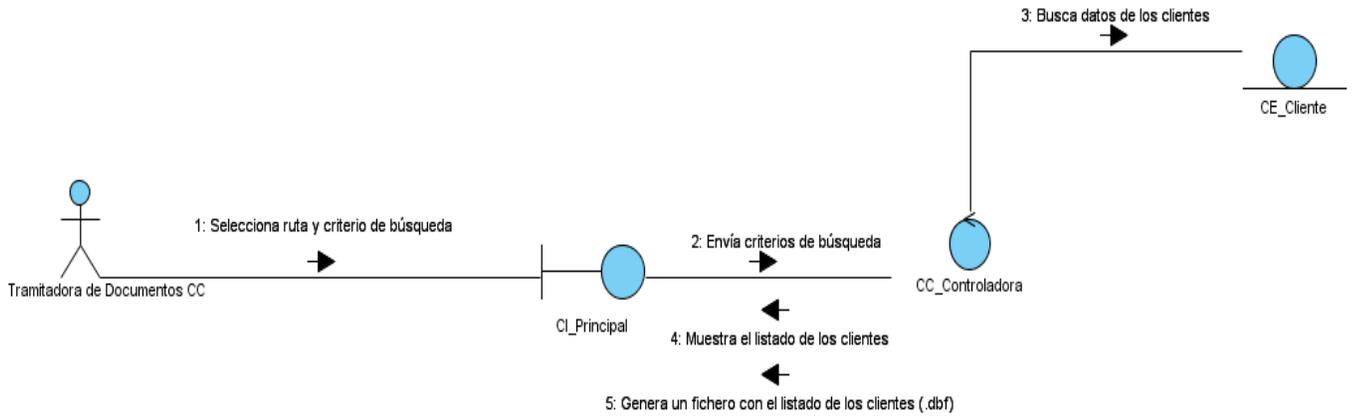
Anexo 3.6

Caso de Uso:	Generar Informe Diario por cobrador
Actores:	Controlador Económico
Resumen:	El caso de uso comienza cuando el Controlador Económico genera el Informe Diario del Trabajo realizado por los cobradores en el día.
Precondiciones:	Se encuentre autenticado en el sistema un usuario con el Rol Controlador Económico.
Referencias	R36
Prioridad	Secundario
Flujo Normal de Eventos	
Sección “”	
Acción del Actor	Respuesta del Sistema
1. El actor selecciona la opción de generar el Informe Diario.	

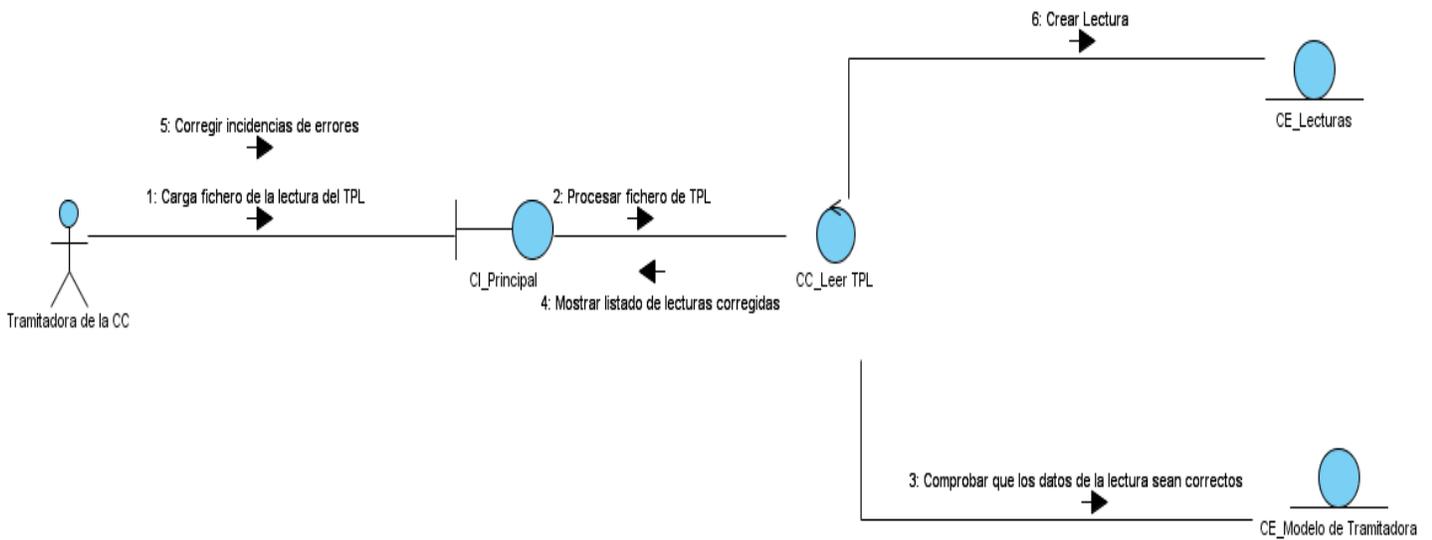
	2. El sistema genera un informe de los cobros realizados en el día.
	3. El sistema muestra el informe diario generado.
4. El actor selecciona la opción de generar un resumen diario de los cobros que fueron realizados.	
	5. El sistema genera un resumen diario de los cobros realizados en efectivo.
	6. El sistema muestra el resumen diario de los cobros realizados en efectivo.

Tabla A3.10 Descripción del Caso de Uso Generar Informe Diario por Cobrador.

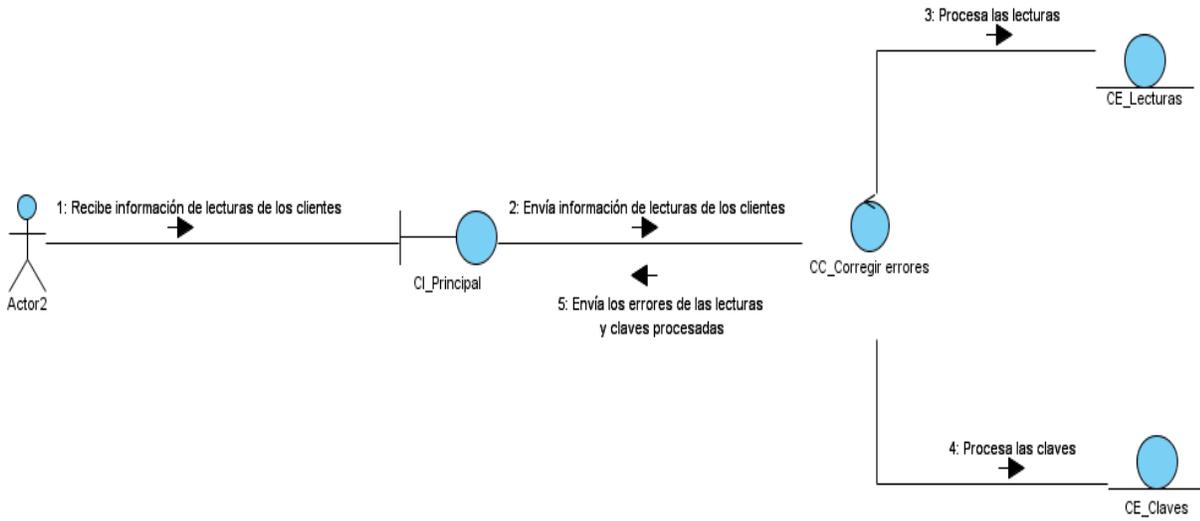
Anexo 3.7 Caso de Uso Cargar TPL



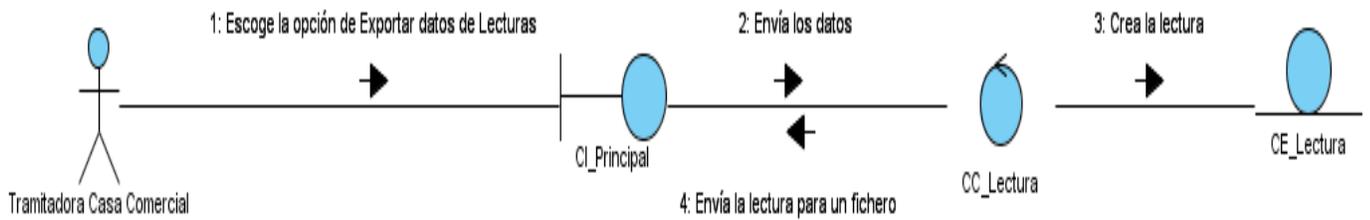
Anexo 3.8 Caso de Uso Leer TPL



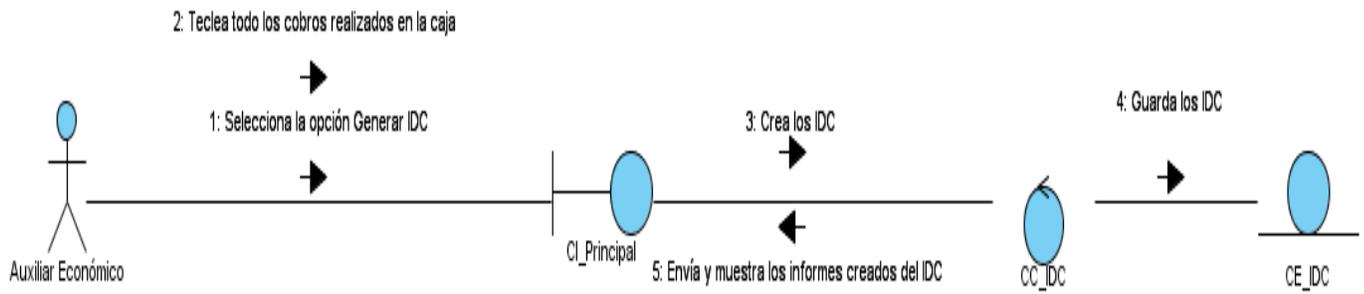
Anexo 3.9 Caso de Uso Corregir Errores



Anexo 3.10 Caso de Uso Exportar Datos de Lectura



Anexo 3.11 Caso de Uso Generar IDC



Diagramas de Clases del Diseño

Descripciones de clases genéricas

Nombre: Layout
Tipo de clase: Clase Vista
Descripción:
Clase relacionada con el framework Symfony, utilizada como plantilla para algunas operaciones dentro y fuera de la misma.

Tabla A3.12 Descripción de la server page Layout.

Nombre: Adicionar Success
Tipo de clase: Página servidora
Descripción:
Página servidora encargada de la construcción de la interfaz a través de la implementación de los métodos.

Tabla A3.13 Descripción de la server page Adicionar Success.

Nombre: adicionarSuccess
Tipo de clase: Página Cliente
Descripción:
Clase construida por la página servidora y encargada de la realización del formulario donde se mostrarán los datos pertenecientes a cada Caso de Uso.

Tabla A3.14 Descripción de la client page adicionarSuccess

Nombre: Modelo
Tipo de clase: Clase Modelo
Descripción:
Clase encargada de la lógica para la interacción con la BD así como las clases que necesita el sistema para la gestión de la información en la BD, donde están los principales métodos de consulta.

Tabla A3.15 Descripción de la modelo.

Nombre: CargarDatosFacturar	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	ActualizarBD()
Descripción:	Se actualiza la BD con los datos que han sido cargados
Nombre:	BuscarDireccionFichero()
Descripción:	Se le da la dirección del fichero con los datos que se desean cargar.

Tabla A3.16 Descripción de la controladora.

Nombre: Modelo	
Tipo de clase: Modelo	
Para cada responsabilidad:	
Clase:	BaseManugasClaveLectura
Descripción:	Clase que maneja la información correspondiente a las lecturas mensuales realizadas a los clientes.
Nombre:	BaseManugasCliente
Descripción:	Clase que maneja la información correspondiente a los clientes que se le realizan las lecturas.
Nombre:	BaseManugasRuta

Descripción:	Clase que maneja la información correspondiente a la dirección de cada uno de los clientes a los que se le presta el servicio
--------------	---

Tabla A3.17 Descripción de la modelo.

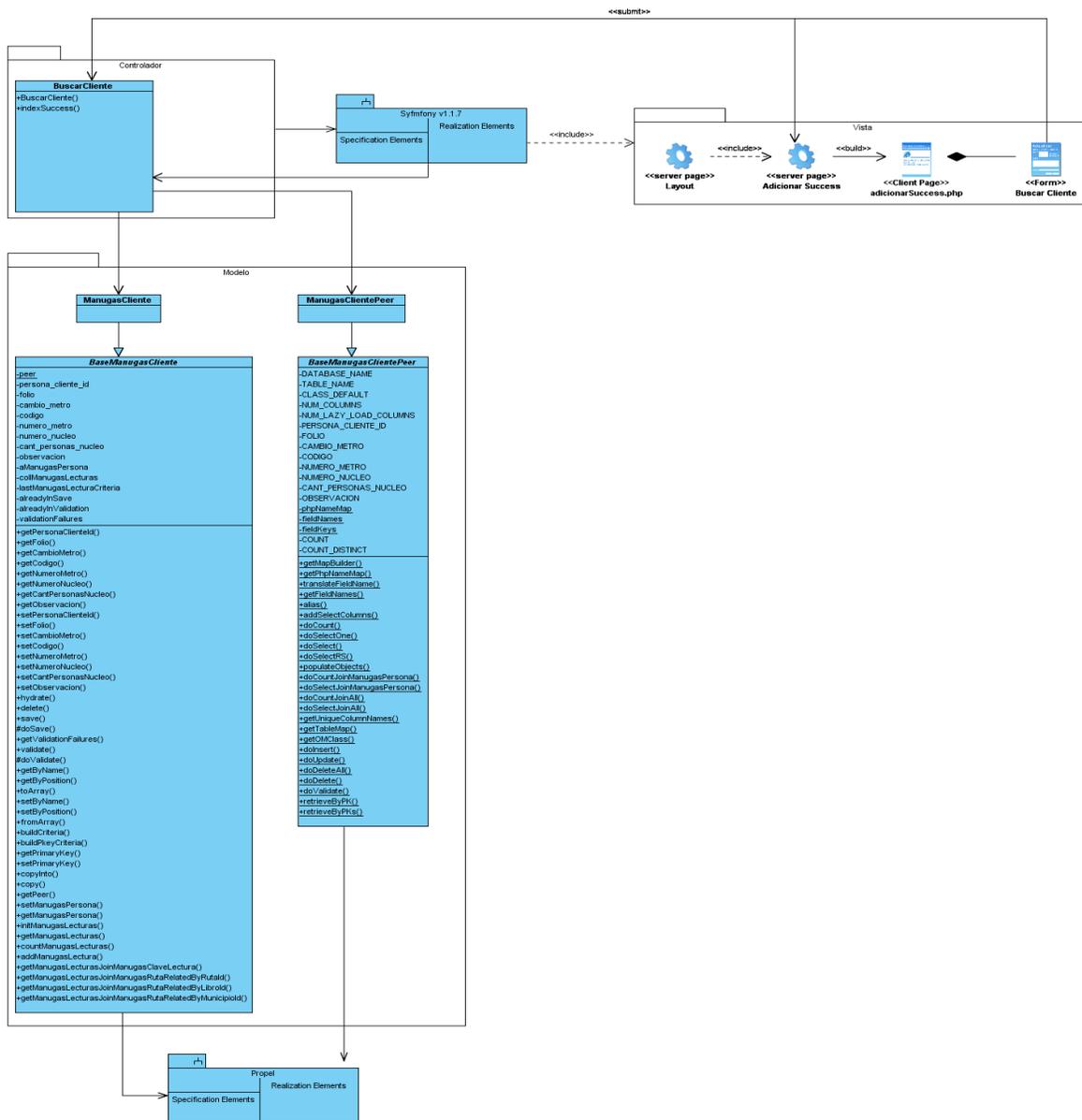
Nombre: ExportarDatosLectura	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	ExportarDatos()
Descripción:	Exporta los datos que van a facturarse para el sistema.
Nombre:	MostrarDatos()
Descripción:	Muestra los datos que ya se han exportado.

Tabla A3.18 Descripción de la controladora.

Nombre: Modelo	
Tipo de clase: Modelo	
Para cada responsabilidad:	
Clase:	BaseManugasClaveLectura
Descripción:	Clase que maneja la información correspondiente a las lecturas mensuales realizadas a los clientes.
Nombre:	BaseManugasRuta
Descripción:	Clase que maneja la información correspondiente a la dirección de cada uno de los clientes a los que se le presta el servicio

Tabla A3.19 Descripción de la modelo.

Anexo 3.18 Caso de Uso Buscar Clientes



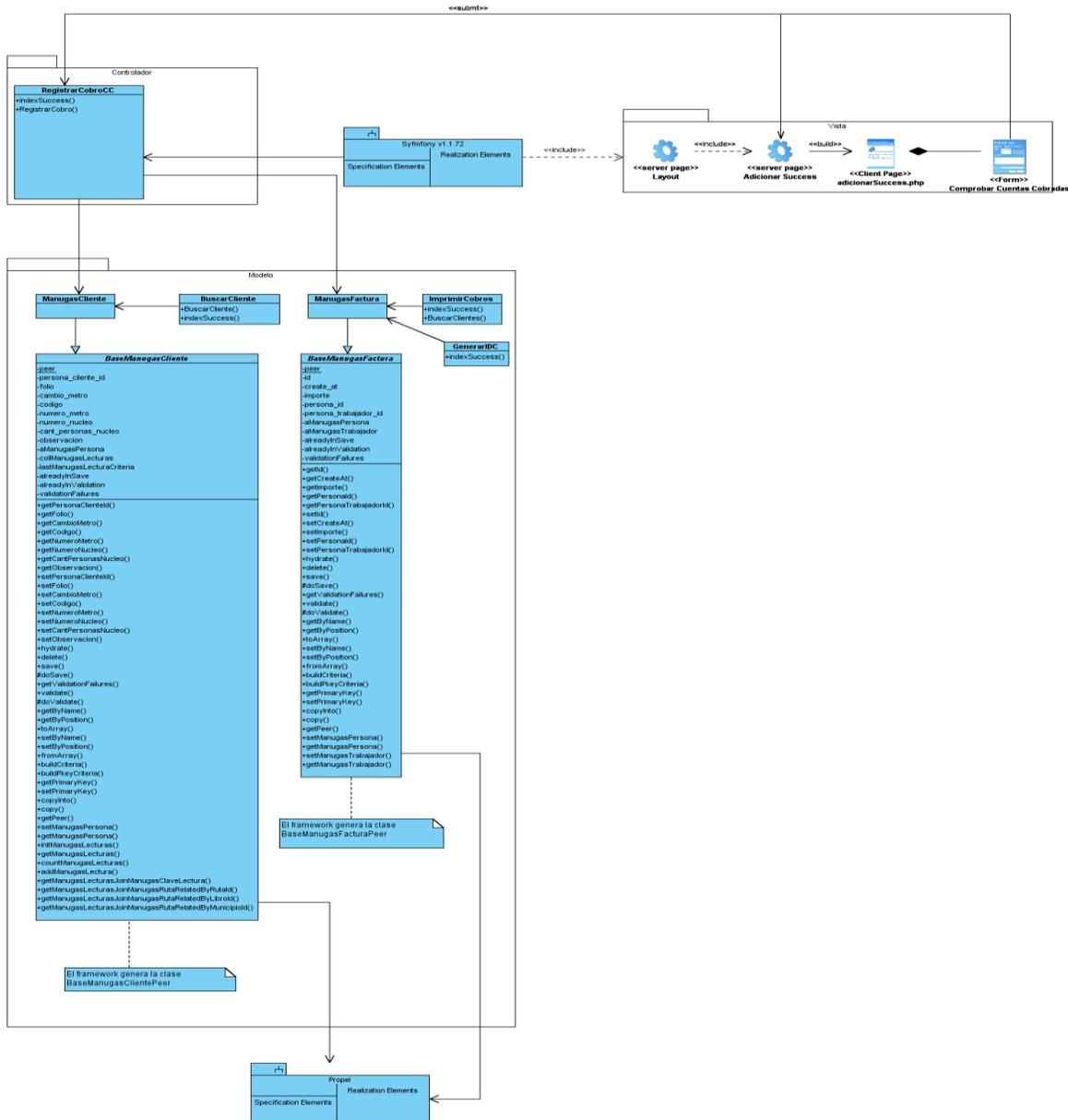
Nombre: BuscarCliente	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	BuscarCliente()
Descripción:	Busca al cliente que necesita para efectuarle el cobro.

Tabla A3.20 Descripción de la controladora.

Nombre: Modelo	
Tipo de clase: modelo	
Para cada responsabilidad:	
Clase:	BaseManugasCliente
Descripción:	Clase que maneja la información correspondiente a los clientes que se le realizan las lecturas.

Tabla A3.21 Descripción de la modelo.

Anexo 3.19 Caso de Uso Registrar Cobro en las Casas Comerciales.



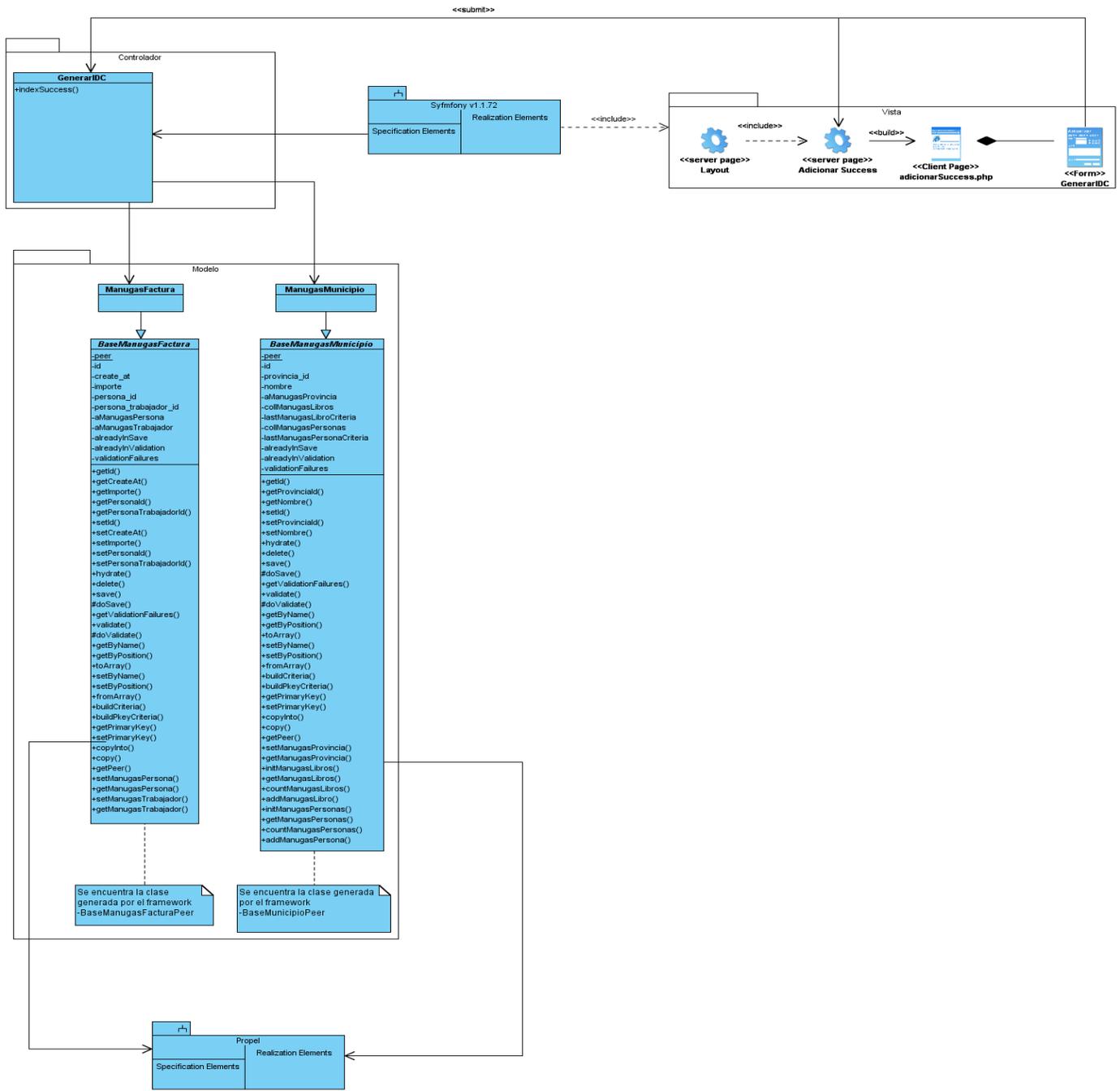
Nombre: RegistrarCobroCC	
Tipo de clase: Controladora	
Para cada responsabilidad:	
Nombre:	RegistrarCobro()
Descripción:	Registra los cobros que se hayan realizado en cada una de las Casas Comerciales.

Tabla A 3.22 Descripción de la controladora.

Nombre: Modelo	
Tipo de clase: Modelo	
Para cada responsabilidad:	
Clase:	BaseManugasCliente
Descripción:	Clase encargada de saber quiénes son las personas que son clientes de la empresa.
Clase:	BaseManugasFactura
Descripción:	Clase encargada de guardar todos los datos de facturación correspondientes a los clientes.

Tabla A3.23 Descripción de la modelo.

Anexo 3.20 Caso de Uso Generar IDC.



Nombre: Modelo	
Tipo de clase: Modelo	
Para cada responsabilidad:	
Clase:	BaseManugasFactura
Descripción:	Clase encargada de guardar todos los datos de facturación correspondientes a los clientes.