

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 3



**Desarrollo del proceso Afiliación de
Empresas Cubanas del módulo
Registral del proyecto Sistema de
Informatización Registral de la
Cámara de Comercio.**

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

Autora: Dayami Darias Dávila

Tutora: Ing. Ana Cecilia Labrador Valdés

Ciudad de La Habana, 2012

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas hacer uso del mismo en su beneficio.

Para que así conste se firma la presente a los ___ días del mes de _____ del año _____.

Dayami Darias Dávila

(Autor)

Ing. Ana Cecilia Labrador Valdés

(Tutora)

AGRADECIMIENTOS

A Dios por sus infinitas misericordias para mi vida porque nuevas son cada mañana y porque dejó que pudiera hacer mía esta promesa de Isaías "... pero los que esperan en Jehová tendrán nuevas fuerzas; levantarán alas como las águilas; correrán, y no se cansarán; caminarán, y no se fatigarán."

A mi papá por ser la persona que más me adora, el que cree en mí, el que pone toda su confianza en mis actos, el que con tanto amor me ha dado todo en la vida, el que me cuida, el que me da todo su apoyo, el que me mimó y cumple todos mis caprichos, el que habla de mí con orgullo, aunque todavía pienses que soy una pequeña de 9 años y hables de mí diciendo "Mi niñita", a ti papá te debo todo en la vida el haberme graduado y más.

A mi mamá por ser la persona que me enseñó a leer y a escribir quien me hizo sentirme responsable de mis estudios desde pequeña, la que me dedicaba largas horas con mis tareas, la que diariamente tiene una enseñanza para darme, la que con consejos de amiga me ha hecho fuerte para enfrentar los obstáculos de la vida, la que no ha descansado ni un momento con tal de que yo siempre esté lista y dispuesta para estudiar. A ti mamá también te debo ser una profesional.

A mi novio por su paciencia, por dedicarme largas horas para que aprendiera mis exposiciones, por haber sido mi apoyo incondicional durante todo este tiempo, por enseñarme que el amor existe y que tiene mucho para darme. A ti que en tan poco tiempo te has convertido en alguien imprescindible e importante en mi vida. Te Amo.

A mi familia, mis abuelos, mis tíos y mis primos, los que me vieron volverse una joven universitaria.

A José Ángel por ser ese amigo incondicional de estos 5 años, el que me ha levantado cuando he flaqueado, y en el cual siempre he encontrado el apoyo necesario cuando lo he necesitado. Eres especial para mí.

A mi amiga Danay por ser mi hermana del alma, la que viene a mi lado desde que éramos bien pequeñas y nunca me ha fallado. La que escucha mis historias y me aconseja. Y a mi segunda hermanita más pequeña Daimi, mi pequeñita que ya es toda una adolescente a ti gracias por quererme tanto y llenar mi vida de alegría.

A mi amigo Rolando, porque aunque estemos lejos siempre te preocupas por mí y es por ti que estoy en esta universidad y no ha existido el día en que no hubiera querido que pudiéramos estar aquí los dos.

A mis compañeras de apartamento Fanelsi, Judith, Miliam y Lenis, por ser como mi familia.

A mis compañeros de aula, los que por 5 años estuvimos juntos compartiendo ratos tristes y alegres. Especialmente a mi amigo Judier, ese amigo que apareció casi de último pero llegó para quedarse en un pedacito de mi corazón.

A mi tutora por las largas horas de dedicación y empeño, por toda su experiencia, la cual puso a mi disposición para llevar a cabo esta investigación.

A los profesores que he tenido durante estos 5 años, a los profes del proyecto y especialmente al profesor Jorin porque en este poco tiempo de conocerlo me ayudó incondicionalmente y en gran manera.

A mis hermanos de la iglesia por sus infinitas oraciones por mí durante todo este tiempo, por su preocupación y todo su cariño.

DEDICATORIA

Esta tesis está dedicada a una persona que ya no está a mi lado, una mujer incondicional, hermosa y cariñosa, la que me llenaba de amor todos los días de mi vida. Tu presencia en mi vida fue importante, contigo aprendí el verdadero significado de querer verdaderamente a alguien, porque tú siempre fuiste todo amor para con todos y sobre todo para conmigo. A ti te recuerdo cada mañana llevándome a tomar helado o compartiendo ratos conmigo, te recuerdo preguntándome por mis amigos, por mi familia y por todos, porque tú eras así, siempre preocupada por los demás, siempre pensando en otros antes que en ti. A ti abuela te dedico mi tesis, mis estudios, mis cinco años en esta universidad, porque nunca hubo nadie que quisiera más que tú que terminara mis estudios y me graduara, se que siempre estuviste muy orgullosa de mí y aún en tus últimos momentos me pedías que nunca dejara mis estudios, por eso mi abuelita linda hoy creo que eres merecedora de todo mi esfuerzo y toda mi vida va dedicada a ti. Te amo y te amaré hasta el fin de mis días, nunca te olvidaré y siempre te tendré presente como hasta ahora. Eres y serás mi ejemplo a seguir. Te quiero...

A mi abuela Amelia... "Hoy no estás a mi lado de forma presente, pero tu recuerdo es tan grande que me impide olvidarte."

RESUMEN

La Cámara de Comercio de la República de Cuba es la encargada de orientar las mejores alternativas para el desarrollo de la actividad empresarial de las entidades que a esa asociación pertenecen, además, constituye una herramienta para la reinserción de la economía cubana en el mundo de las relaciones económicas internacionales, siendo un organismo de vital importancia para el desarrollo económico de las empresas. Actualmente este organismo no posee un sistema informático que permita gestionar toda la información que se genera al iniciarse el proceso de afiliación para las empresas que lo solicitan, llevándose todo el proceso de forma manual.

Se decidió crear un sistema que informatizará varios procesos relacionados con la actividad registral, entre ellos el proceso de afiliación de empresas cubanas, el mismo está concebido para minimizar la complejidad y heterogeneidad del proceso de afiliación. El desarrollo del sistema está guiado por el Proceso Unificado de Desarrollo y presenta un enfoque a procesos. Se utiliza Java como lenguaje de programación y NetBeans como herramienta de desarrollo. Como servidor de aplicaciones web se emplea el Apache Tomcat, mientras que para la gestión de procesos se utiliza Bonita Open Solution.

El resultado obtenido fue una aplicación Web que permite que la información del proceso Registral de Afiliación de Empresas Cubanas se encuentre accesible, centralizada y estandarizada, y que además se puedan realizar búsquedas rápidas y variadas utilizando uno o varios criterios para obtener los datos de las empresas registradas y generar reportes.

PALABRAS CLAVES

Asociación, Cámara de Comercio, proceso de afiliación, sistema.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Introducción	6
1.2 Sistemas para resolver problemáticas similares en países foráneos	6
1.3 Descripción del flujo del proceso de Afiliación de Empresas Cubanas	8
1.4 Criterios de selección de las herramientas y tecnologías	9
1.5 Metodología de desarrollo	11
1.5.1 Modelado del negocio	14
1.5.2 Modelado del sistema	16
1.6 Herramienta CASE para el desarrollo de software	18
1.6.1 Selección de la herramienta CASE para el desarrollo de software	19
1.7 Gestor de procesos	20
1.8 Lenguajes de programación	22
1.8.1 Selección del lenguaje de programación que será utilizado en la solución	22
1.8.2 Selección de la herramienta de desarrollo	23
1.9 Plataforma empresarial de java	24
1.10 Servidor de aplicaciones	25
1.11 Tecnología JPA	26
1.12 Framework	27
1.12.1 Spring	27
1.13 Patrones	28
1.13.1 Patrones de modelado de procesos	28
1.13.2 Patrones de casos de uso	29
1.14 Conclusiones	30
CAPÍTULO 2: REQUISITOS Y DISEÑO	32
2.1 Introducción	32
2.2 Modelo de proceso del negocio	32
2.3 Requisitos de software	33
2.3.1 Requisitos funcionales	34
2.3.2 Requisitos no funcionales	35
2.4 Modelo del sistema	36
2.4.1 Actores del sistema	37

ÍNDICE

2.4.2	Diagrama de casos de uso del sistema	37
2.4.3	Descripción de casos de uso del sistema	38
2.5	Arquitectura	48
2.5.1	Patrones de diseño	50
2.6	Modelo de diseño	52
2.6.2	Diagrama de paquetes o subsistemas de diseño	52
2.6.3	Diagrama de clases del diseño	53
2.6.4	Diagramas de secuencia	54
2.7	Conclusiones	54
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS		55
3.1	Introducción	55
3.2	Diagrama de componentes	55
3.3	Verificación	55
3.4.1	Métricas para la calidad de la especificación de los requisitos de software	56
3.4.2	Métricas para la calidad de la funcionalidad del diagrama de casos de uso de sistema	57
3.4.3	Revisión por pares	60
3.4.4	Listas de chequeo	61
3.4	Validación	62
3.4.1	Validación del modelado de proceso	62
3.4.2	Validación de requisitos	63
3.4.3	Pruebas de caja negra	63
3.4.4	Carta de aceptación del cliente	65
3.5	Conclusiones	65
CONCLUSIONES GENERALES		67
RECOMENDACIONES		68
BIBLIOGRAFÍA		69
ANEXOS		73
GLOSARIO DE TÉRMINOS		81

ÍNDICE DE FIGURAS

Figura 1: Fases e iteraciones de la metodología RUP.	13
Figura 2: Elementos principales de los diagramas de BPMN.	16
Figura 3: Elementos que componen a los diagramas de clases, de casos de uso y de interacción.....	18
Figura 4: Patrón secuencia.	28
Figura 5: Patrón ciclos arbitrarios.....	29
Figura 6: Patrón terminación implícita	29
Figura 7: Diagrama de proceso de negocio: Registrar afiliación de empresas cubanas.	33
Figura 8: Diagrama de casos de uso del sistema.	38
Figura 9: Vista de la Arquitectura.	50
Figura 10: Diagrama de paquetes.....	52
Figura 11: Diagrama de clases del diseño, CU Gestionar empresa.	53
Figura 12: Gráfica de resultados de la métrica para la calidad de la funcionalidad de los DCU.	60
Figura 13: Resultados de la aplicación de la técnica revisión por pares.	61

ÍNDICE DE TABLAS

Tabla 1: Descripción de los actores del sistema.	37
Tabla 2: Descripción del caso de uso, Gestionar Empresa.	48
Tabla 3: Descripción del grupo de revisores	56
Tabla 4: Resultado de las interpretaciones de requisitos.....	57
Tabla 5: Resultado de las métricas correspondientes a los atributos completitud, consistencia, correctitud y complejidad.	59
Tabla 6: Resultados de la revisión por pares.	61
Tabla 7: Resultado de la aplicación de las listas de chequeo.	62
Tabla 8: Caso de prueba: Requisito Registrar empresa.	65

INTRODUCCIÓN

El comercio desde sus inicios ha constituido la actividad fundamental para el desarrollo económico y social de las civilizaciones. Sus orígenes se deben a los excedentes de las cosechas, que empezaron a intercambiarse con otros objetos en los que otras comunidades estaban especializadas, de esta forma cada grupo se beneficiaba según su necesidad y su posibilidad de intercambio. El acelerado crecimiento de las formas de comercio dio paso a la llegada del dinero, que se convirtió en el objeto principal de las negociaciones. Es el comercio la actividad socioeconómica que consiste en la compra y venta de bienes, ya sea para su uso, para su venta o para su transformación. Se trata de la transacción de algo a cambio de otra cosa de igual valor (que puede ser dinero). (Definición.de, 2011)

Con el objetivo de controlar y proteger intereses personales, así como mejorar el intercambio comercial surgen las cámaras de comercio, las cuales agrupan a un conjunto de comerciantes. Estas son organismos extendidos por todo el mundo, regulados por diferentes leyes según el propósito para el cual fueron creadas y las disposiciones de comercio de cada país. En casi todos los países existen Cámaras de Comercio y Cuba no está exenta de estos avances.

La creación de la Cámara de Comercio en Cuba transitó por diversas etapas desde la época colonial hasta la actualidad, se fundó el 10 de mayo de 1876, siendo una agrupación que solo representaba a los habaneros, con el paso del tiempo y de los diversos gobiernos, fue cambiando sus características y los intereses que representaba. En 1963 bajo los albores del Triunfo Revolucionario la cámara anterior fue disuelta y creada la actual Cámara de Comercio de la República de Cuba con estructura y fines diferentes.

La Cámara de Comercio de Cuba es una asociación de empresas vinculadas al comercio, la industria y los servicios, con reconocimiento ante los organismos del Estado; que permite orientar las mejores alternativas para el desarrollo de la actividad empresarial de las entidades que a esa asociación pertenecen, y constituye una herramienta para la reinserción de la economía cubana en el mundo de las relaciones económicas internacionales. Es una entidad que promueve las ofertas exportables de productos y servicios, así como las oportunidades de negocios e inversión de las empresas cubanas y la sustitución de importaciones, en beneficio de la economía nacional. Presta servicios a sus

INTRODUCCIÓN

asociados, a las sucursales de compañías extranjeras acreditadas en el país, así como a empresarios extranjeros interesados en negociar con Cuba. (Cámara de Comercio de Cuba, 2009)

Actualmente la Cámara de Comercio gestiona gran parte de los datos de las empresas asociadas de manera tradicional, es decir de forma manual, lo que no siempre brinda la fiabilidad necesaria, ni asegura que la información sea la correcta. Frecuentemente esta información enfrenta problemas como el difícil acceso, la redundancia y desactualización, el mal cuidado y en el peor de los casos la pérdida. Al concluir el proceso registral de las empresas se insertan los datos de las mismas en hojas de cálculo creadas en Microsoft Excel, herramienta de software propietario que se utiliza para crear hojas de cálculos y representar tablas complejas, el mismo es engorroso para gestionar el cúmulo de información que presenta dicha entidad. Debido a esta situación el manejo de estos datos se vuelve lento y difícil pues esta herramienta no es la adecuada para apoyar y agilizar el proceso de búsqueda y no permite gestionar la información requerida de inicio a fin. Todas estas condiciones conllevan a que el trabajo se vuelva complejo cuando la cantidad de empresas afiliadas crece y se requiere consultar los datos de las mismas, lo que provoca atraso y poca calidad de los procesos que allí se realizan.

Teniendo en cuenta esta problemática se decide crear un *Sistema para la Informatización Registral de la Cámara de Comercio de la República de Cuba (SIRECC)*, tarea que le es asignada a la *Universidad de las Ciencias Informáticas (UCI)*, entidad que ha desarrollado un número considerable de sistemas informáticos para diversos organismos y que ha obtenido importantes resultados para el estado cubano en materia de informatización del país y en la prestación de servicios asociados en el exterior.

Con el desarrollo del sistema registral se pretende lograr un producto que sirva de apoyo a los funcionarios de la Cámara de Comercio y facilite el trabajo relacionado con los procesos registrales. Este sistema informatizará varios procesos relacionados con la actividad registral, entre ellos el proceso de afiliación de las empresas cubanas a la cámara.

Teniendo en cuenta lo antes expuesto surge el siguiente **problema de investigación**: La forma en que se registran las empresas cubanas en la Cámara

INTRODUCCIÓN

de Comercio no permite gestionar la información de las mismas de manera que se encuentre centralizada, estandarizada y accesible, realizar búsquedas rápidas y variadas y generar reportes. Se propone como **objeto de estudio**: El proceso de desarrollo de software.

De esta forma, el **objetivo general** del presente trabajo es: Desarrollar el proceso Afiliación de Empresas Cubanas del módulo Registral del proyecto SIRECC, que permita gestionar de forma centralizada, estandarizada y accesible la información de las empresas; realizar búsquedas rápidas y variadas y generar reportes. Teniendo como **campo de acción**, la ingeniería de requisitos, diseño, implementación y pruebas.

Utilizando como referencia el objetivo general y el problema de investigación se ha obtenido como resultado la siguiente **idea a defender**: Con el desarrollo del proceso Afiliación de Empresas Cubanas del módulo Registral se contribuirá a una mejor gestión de la información, realizar búsquedas rápidas y variadas y generar reportes.

Los **objetivos específicos** que se persiguen son los siguientes:

- Elaborar el marco teórico de la investigación.
- Desarrollar los requisitos del proceso Afiliación de Empresas Cubanas.
- Implementar los requisitos obtenidos.
- Verificar y validar la solución propuesta.

Para dar cumplimiento a los objetivos específicos se trazaron las siguientes **tareas de la investigación**:

- Estudio de sistemas existentes para resolver problemáticas similares en países foráneos.
- Descripción del flujo del proceso de Afiliación de Empresas Cubanas.
- Dominio y aplicación de las metodologías, herramientas y lenguajes más utilizados a nivel mundial para el desarrollo de este tipo de aplicaciones.
- Identificación y modelación del proceso Afiliación de Empresas Cubanas.
- Identificación, análisis, especificación y validación de los requisitos del proceso Afiliación de Empresas Cubanas.
- Diseño de las clases del proceso Afiliación de Empresas Cubanas.
- Implementación del proceso Afiliación de Empresas Cubanas.

INTRODUCCIÓN

- Pruebas de calidad de software a través de métricas y diseños de casos de prueba basados en requisitos.

MÉTODOS Y TÉCNICAS DE INVESTIGACIÓN

Para validar metodológicamente la investigación se utilizaron los siguientes métodos:

Teóricos

- **Análisis Histórico-Lógico:** Permite estudiar la evolución de los procesos de afiliación de las empresas y el funcionamiento de estos en los diferentes organismos asociados a la cámara, para llegar a un análisis razonado de los mismos.
- **Analítico-Sintético:** Permite la recopilación y estudio de la información relacionada con el proceso de afiliación de empresas cubanas a la Cámara de Comercio.
- **Modelación:** Para el desarrollo de los artefactos es necesario la elaboración de diagramas, figuras y otros documentos importantes, por lo que se usará este método, para crear abstracciones con el propósito de explicar la realidad.

Empíricos

- **Observación:** Permite analizar cómo se desarrolla en la Cámara de Comercio el proceso de afiliación, además brinda la posibilidad de obtener conocimientos a partir de las situaciones que se presentan en la misma.
- **Entrevistas:** Es empleada como técnica de interrogación para recoger información de utilidad acerca del proceso de afiliación de empresas cubanas a la Cámara de Comercio, mediante preguntas que se formulan durante las visitas de los analistas al personal encargado de estos procesos en la misma.

RESULTADOS ESPERADOS

- Descripción de los procesos del negocio.
- Especificación de requisitos de software.
- Modelo de casos de uso del sistema.
- Prototipos de interfaz de usuario.

INTRODUCCIÓN

- Modelo de diseño.
- Diseño de casos de prueba basados en requisitos.

ESTRUCTURA DE LA TESIS

El presente trabajo, estructurado en 3 capítulos, resume la siguiente información:

Capítulo 1. Fundamentación teórica: Se realiza un estudio sobre sistemas existentes en otros países para la informatización registral de las Cámaras de Comercio, los cuales pudieran servir de guía para el desarrollo del sistema que se quiere obtener. Además se describen detalladamente la metodología, las herramientas, lenguajes y tecnologías que se aplicarán para obtener la solución deseada; y se exponen los principales patrones para modelar y describir procesos y casos de uso.

Capítulo 2. Requisitos y Diseño: Se exponen las funcionalidades y restricciones del sistema mediante los requisitos funcionales y no funcionales. Se obtienen los diagramas de procesos, el diagrama de casos de uso, y la respectiva descripción de cada uno, así como los diagramas de clases del diseño y diagramas de secuencias para cada escenario de cada caso de uso, logrando una entrada adecuada a las actividades de implementación. Además se describe la arquitectura propuesta para el desarrollo del sistema y se especifican los patrones de diseño que se utilizarán.

Capítulo 3. Implementación y Pruebas: Se presenta el diagrama de componentes relacionado con la implementación y se aplican las métricas para la especificación de requisitos y diagramas de casos de uso con el objetivo de asegurar la calidad de los artefactos obtenidos. Además se aplican listas de chequeo para la validación del modelo de diagramas de procesos, descripción de requisitos, y del modelo de casos de uso del sistema. Se validará la solución a través del proceso de aceptación con el cliente, avalado por la carta de aceptación para los artefactos que le fueron presentados.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 INTRODUCCIÓN

El presente capítulo constituye el marco teórico de la investigación a realizar. Se puntualizan una serie de conceptos necesarios para entender el objetivo fundamental del trabajo, profundizando en temas como la *Notación de Modelado de Procesos de Negocio (BPMN)* y el *Lenguaje Unificado de Modelado (UML)*, definiendo conceptos como metodologías de desarrollo de software, tratando particularmente RUP (*Proceso Unificado de Desarrollo*), además de los lenguajes de programación y frameworks. Finalmente se precisan y caracterizan las herramientas y tecnologías a utilizar en el desarrollo del software.

1.2 SISTEMAS PARA RESOLVER PROBLEMÁTICAS SIMILARES EN PAÍSES FORÁNEOS

Con la finalidad de entender el sistema que se quiere implementar se realizó un estudio sobre la tendencia actual de los sistemas camerales con respecto a la forma en que gestiona la información de sus procesos registrales.

El estudio realizado a diversas cámaras arrojó importantes resultados entre los que se encuentra específicamente la Cámara de Comercio de Lima, quien a fin de brindar mayor asesoría e información para hacer negocios, desarrolló un proyecto basado en una plataforma de servicios integrados, los cuales permiten a las empresas hacer negocios a nivel local e internacional, apoyadas con herramientas de primer nivel y a bajo costo. Comprende más de 30 servicios implementados en una plataforma web. El proyecto y los servicios presentados están basados en cuatro pilares: apoyo para aprovechar el crecimiento del volumen de negocios, exposición comercial, comercio internacional e internacionalización y mejora del entorno empresarial o para los negocios. (Cámara de Comercio - Lima, 2007)

Por otro lado la Cámara de Comercio de Medellín para Antioquia en Colombia es una entidad líder en la región, con una orientación en servicios empresariales. La misma avanza en la modernización de sus servicios e incorpora a su oferta de servicios virtuales del Registro Mercantil la plataforma de Inscripciones virtuales, una herramienta que permite realizar múltiples trámites en línea (Cámara de Comercio de Medellín para Antioquía, 2011). Este servicio tiene múltiples

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

beneficios para los comerciantes inscritos en el Registro Público Mercantil de la Cámara de Comercio. Estos son algunos de ellos: (Cámara de Comercio de Medellín para Antioquía, 2011)

- Los trámites realizados a través de la plataforma, tienen la misma validez jurídica y probatoria de un trámite realizado de manera presencial en las sedes de la Cámara de Comercio.
- En el servicio de inscripciones virtuales se incorporan la firma digital y el estampado cronológico, que garantizan la autenticidad, integridad y no repudio del trámite del documento presentado para el registro.
- La plataforma de inscripciones virtuales ofrece en línea los datos necesarios para construir el acto a registrar.
- Se dispone, en tiempo real, de la información registrada en la Cámara de Comercio, facilitando efectuar los cambios que sean necesarios.
- La plataforma contiene los requisitos formales requeridos para cada trámite, con lo cual se minimiza la devolución del documento por este tipo de requisitos.
- El pago correspondiente al trámite puede realizarse a través de internet, sin necesidad de dirigirse a las sedes de la Cámara de Comercio.

Además de esta plataforma, la gran mayoría de las cámaras se enfocan en brindar servicios a través de su página principal o por medio de centros de atención en línea. A continuación se mencionan algunos ejemplos:

La página web de la Cámara de Comercio de Bogotá ha dispuesto para sus usuarios la facilidad de poder realizar trámites online para el Registro Único de Proponentes a través de internet. Esta es una herramienta que le permite diligenciar el formulario de inscripción, renovación y actualización a través de su página web, evitando cometer errores de forma y agilizando los cálculos que se requieren para obtener su *capacidad de contratación (K de contratación)*. Como paso número 1 el sistema da la posibilidad de llenar el formulario digital para la inscripción e impresión del mismo. En el paso número 2 debe escoger la actividad a la cual se ajusta la empresa. El último paso será la inscripción, la cual se realizará en cualquiera de las sedes de la cámara. (CC de Bogotá, 2011)

El Centro de Atención Empresarial (CAE) es un servicio de la Cámara de Comercio de Florencia para el Caquetá en Colombia que se pone a disposición

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

del sector productivo, para adelantar el proceso de Creación y Constitución de Empresas. Este centro permite a los comerciantes realizar varios trámites ingresando en la página, entre ellos podemos encontrar los siguientes: (Cámara de Comercio de Florencia para el Caquetá, 2011)

- Inscripción en el registro mercantil.
- Consulta de homonimia¹.
- Matrícula mercantil del establecimiento comercial.
- Inscripción de libros de comercio.
- Inscripción en el registro único de proponentes.
- Inscripción en industria, comercio y complementarios del municipio de Florencia.
- Consulta de uso de suelo.
- Trámites notariales.

El uso de las nuevas tecnologías y la creación de sistemas informáticos para mejorar el servicio que brindan las Cámaras de Comercio se traduce en mejor calidad de vida y un servicio de excelencia enfocado al cliente. El estudio de estos sistemas ha permitido visualizar el funcionamiento de diversos procesos semejantes al proceso de afiliación en otros sistemas camerales, viendo como su objetivo principal es facilitar el proceso a través de la gestión online de la afiliación. En nuestro país actualmente se hace engorrosa la utilización de las tecnologías enfocadas al uso de internet, además la Cámara de Comercio de Cuba funciona de acuerdo a las leyes y regulaciones establecidas por el país, por lo que es necesario crear una solución a la medida para este tipo de entidad.

1.3 DESCRIPCIÓN DEL FLUJO DEL PROCESO DE AFILIACIÓN DE EMPRESAS CUBANAS

El proceso de afiliación de las empresas nacionales a la Cámara de Comercio de la República de Cuba se realiza de la forma siguiente: En un principio debe explicársele detalladamente a las empresas que lo soliciten, cada uno de los servicios que brinda la Cámara de Comercio a sus asociados, que documentos debe presentar la empresa interesada y en qué consiste el proceso de afiliación,

¹ Cualidad de homónimo. Dicho de dos o más personas o cosas que llevan un mismo nombre, según el Diccionario de la Real Academia Española.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

encargándose de esta tarea el Especialista en Desarrollo Empresarial. Las empresas interesadas en asociarse presentan ante la cámara todos los documentos necesarios para la solicitud de afiliación, ellos son: Carta de solicitud de afiliación, Ficha de la entidad, Copia de la Resolución de constitución de la Empresa y Copia de la Resolución de nombramiento del Director, los cuales son revisados rigurosamente (Director de Formación y Membrecía, 2009). Con estos documentos se prepara un expediente por parte de los funcionarios de la cámara, donde se resumen los datos principales de la empresa que solicita la afiliación; documentación que también es revisada y firmada por el Director de Formación y Membrecía de la cámara. Posteriormente se envía el expediente a la Junta Directiva para que se realice la Admisión de Asociación, una vez revisado el expediente, la Junta Ejecutiva emite el documento Criterio de la Junta Ejecutiva sobre Admisión de Asociación (Director de Formación y Membrecía, 2009). En caso de no ser aprobado el expediente se le comunica a la empresa la denegación de su solicitud. Si la solicitud es aprobada se disponen a elaborar los documentos asociados al proceso de afiliación con las firmas pertinentes. Al terminarse el proceso de aprobación y asentamiento jurídico el Técnico en Información Comercial registra en la base de datos toda la información de la empresa y pasa copia de la ficha de la empresa y Criterio de la Junta Ejecutiva sobre Admisión de Asociación al área de finanzas en la Dirección de Contabilidad y Finanzas, para la confección del Aviso de cobro. Concluido este proceso, se comunica a la empresa solicitante la aprobación de la solicitud y se le entregan además los documentos asociados al proceso de afiliación, publicaciones y actividades programadas para su participación. (Director de Formación y Membrecía, 2009)

1.4 CRITERIOS DE SELECCIÓN DE LAS HERRAMIENTAS Y TECNOLOGÍAS

Con el objetivo de tener una selección acertada de las herramientas y tecnologías a aplicar en el desarrollo del sistema registral se definieron diferentes criterios que dichas herramientas debían cumplir para su elección. Los criterios por los cuales se deben regir son los siguientes:

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Mantenibilidad: El IEEE ²(1990) define mantenibilidad como: “La facilidad con la que un sistema o componente software puede ser modificado para corregir fallos, mejorar su funcionamiento u otros atributos o adaptarse a cambios en el entorno” (Veronica de la Morena, 2008). Esta definición enuncia el por qué se desea que las herramientas o tecnologías escogidas cumplan con este criterio.

Independencia tecnológica: La política de la universidad es la migración a software libre, requisito indispensable para lograr la independencia tecnológica, por lo que es necesario seleccionar herramientas que garanticen las siguientes libertades: la libertad de usarlo, estudiarlo, compartirlo y modificarlo.

Portabilidad: Se define como: “La capacidad del software para ser trasladado de un entorno a otro. El entorno puede incluir entornos organizacionales, de hardware o de software” (ISO 9126, 2005). La selección de este criterio facilita el uso de las herramientas y tecnologías seleccionadas en diferentes entornos de trabajo.

Enfocado a procesos: Un enfoque a procesos es un camino para organizar y gestionar las actividades que crean valor en la empresa obteniendo eficientemente un resultado cuando las actividades y los recursos se gestionan como un proceso, siendo este último precisamente como un conjunto de actividades mutuamente relacionadas o que interactúan, las cuales transforman elementos de entrada en resultados (ISO/TC 176, 2009). Las empresas están organizadas como áreas dentro de una jerarquía funcional. El enfoque de procesos elimina las barreras entre diferentes áreas funcionales y unifica sus enfoques hacia las metas principales de la organización (Grupo Kaizen Calidad, 2005). Las organizaciones tienen la visión de usar las tecnologías de la información para informatizar sus procesos. Sin embargo, este objetivo no es tan fácil de lograr. Estas organizaciones dependen cada vez más de sistemas informáticos utilizados en la consecución de sus metas, pero dichos sistemas son usados independientemente o de forma separada estableciendo entre los mismos, dependencias complejas. Se hace necesario por consiguiente que los procesos sean definidos y gestionados por las personas que tienen el conocimiento sobre ellos y que la informatización de dichos procesos sea realizada sobre la base de esa definición de acuerdo a los cambios de la

² Institute of Electrical and Electronics Engineers

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

tecnología sin afectar dicha definición y al mismo tiempo minimizar el impacto de los cambios en los procesos en la implementación (Delgado, 2007), aunque siempre dicha implementación de acuerdo a este enfoque será más económica ya que aunque un proceso afecte a otros, es mucho más sencillo cambiar o mejorar el proceso o partes del mismo, sin afectar al resto (Colectivo de Autores, 2011). Este último aspecto es muy importante ya que actualmente existe mucha dinámica en el área empresarial y las empresas deben cambiar y/o mejorar sus procesos con el objetivo de tener un lugar en la competencia y brindar mejores servicios al cliente. Comúnmente en el desarrollo de software se identifican cuáles son las actividades de un determinado proceso que serán objeto de automatización, el resultado es un software que automatiza partes del proceso de negocio y no el proceso en sí. Existe por consiguiente el problema de que al cambiar la definición del proceso de negocio, el software quede obsoleto o sea necesario realizar una actualización que conlleva gastos y esfuerzo adicional.

Documentación: Bajo este término genérico se agrupan todos los manuales, guías de referencia, libros de ayuda, etc., que suelen entregarse con cada programa, de manera que el usuario pueda aprender su manejo y consultar cualquier duda ante un problema desconocido (MasterMagazine, 2004). Para el desarrollo del sistema se seleccionaron herramientas con suficiente documentación que pueda ser consultada ante cualquier duda.

1.5 METODOLOGÍA DE DESARROLLO

Luego de la realización de un estudio de las metodologías más usadas en la actualidad, se concluyó que la metodología a emplear será *RUP* (*Proceso Unificado de Desarrollo, Rational Unified Process*) pues presenta características perfectamente ajustables al proyecto y es la más conocida por el equipo de desarrollo.

RUP provee un marco de trabajo genérico que puede especializarse para una gran cantidad de sistemas software. Se caracteriza por ser: (Jacobson, 2000)

- Iterativo e incremental: Cada fase se desarrolla en iteraciones. Se divide el trabajo en partes más pequeñas o mini-proyectos, donde cada uno es una iteración que resulta en un incremento.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Centrado en la arquitectura: La arquitectura 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.
- Guiado por casos de uso: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requisitos.

Como RUP es un proceso define “quién” está haciendo “qué”, “cuándo” y “cómo” para alcanzar un determinado objetivo. (Jacobson, 2000)

En RUP se han reunido las actividades en grupos lógicos, definiéndose nueve flujos de trabajo principales. Los seis primeros son conocidos como flujos de ingeniería y los tres últimos como flujos de apoyo. Estos flujos de trabajo son: (Rational Software Corporation, 2003)

- Modelado del negocio: Describe los procesos del negocio, identificando quiénes participan y las actividades que requieren automatización.
- Requisitos: Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- Análisis y diseño: Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requisitos), por lo que indica con precisión lo que se debe programar.
- Implementación: Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- Pruebas: Busca los defectos a lo largo del ciclo de vida del producto de software.
- Despliegue: Produce liberaciones del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.) para entregar el software a los usuarios finales.
- Administración de configuración y cambios: Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a control de versiones, utilización y actualización concurrente de elementos y otros.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Administración del proyecto: Involucra actividades con las que se busca construir un producto que satisfaga las necesidades de los clientes.
- Ambiente: Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

El ciclo de vida del software de RUP es descompuesto en cuatro fases secuenciales, cada una posee varios hitos. Al final de cada fase se realiza una evaluación para determinar que los objetivos de la fase han sido cumplidos. Una evaluación satisfactoria permite al proyecto avanzar a la próxima fase. (Jacobson, 2000)

- Inicio: El objetivo en esta etapa es determinar la visión del proyecto.
- Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción: En esta etapa el objetivo es llegar a obtener la capacidad operacional inicial.
- Transición: El objetivo es llegar a obtener el reléase del proyecto.

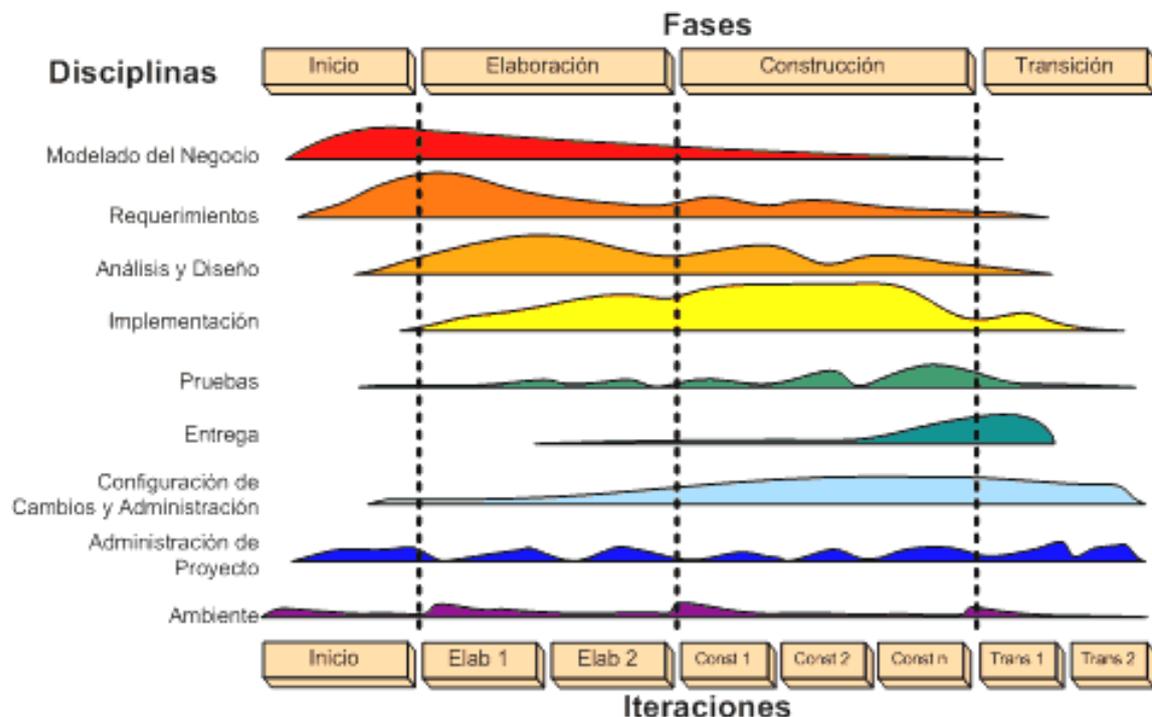


Figura 1: Fases e iteraciones de la metodología RUP.

Esta metodología constituye uno de los estándares internacionales que más aceptación ha tenido últimamente en el desarrollo de software. Además, varias herramientas CASE (*Computer Aided Software Engineering, Ingeniería de*

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Software Asistida por Ordenador) soportan dicha metodología, permitiendo el trabajo en equipo y la capacidad de generar código en distintos lenguajes de programación a partir de un diseño *UML (Lenguaje Unificado de Modelado)*.

Gracias al proceso de mejora que se lleva a cabo en la universidad se podrán, para el diseño del sistema registral, utilizar los flujos de trabajo que propone RUP comprendidos a partir de Modelado del negocio, Requisitos, Diseño, Implementación y Pruebas.

1.5.1 Modelado del negocio

El modelado del negocio se orientó a procesos por lo que se utilizó *BPMN 2.0 (Business Process Modeling and Notation, Notación de modelado de procesos de negocio)*, es una notación gráfica para el modelado conceptual de procesos de negocio. Proporciona la capacidad de entender y definir procesos de negocio, tanto internos como externos, a través de un diagrama de procesos de negocio. Es un estándar que provee una representación gráfica (mediante diagramas) para expresar procesos de una empresa. Se diseñó con el objetivo de facilitar la comprensión por parte de todos los implicados (expertos TIC, analistas de negocio, directivos, etc.) que participan en el proceso. (Experiencia en transformación de modelos de procesos, 2007)

BPMN define la notación y semántica de un *Diagrama de Procesos de Negocio (Business Process Diagram, BPD)*. Este último consta de cuatro categorías básicas de elementos que son:

- Elementos de flujo: Son elementos de modelado que sirven para controlar como interactúa el flujo de secuencia mientras converge y diverge dentro de un proceso. Consiste en una colección de puertas que controlan la salida y la decisión, determinando como estarán dispuestas las mismas. Constituyen elementos de flujo dentro de un BPD los eventos, las actividades y las decisiones.
 - Eventos: Es “algo que pasa” durante el curso de un proceso de negocio. Afectan el flujo del proceso y usualmente tienen una causa y un impacto. Existen tres tipos de eventos: *Comienzo* indica donde empieza el proceso y no tiene flujos de secuencia entrantes, *Intermedio* ocurre entre los eventos de inicio y fin y es usado para interrumpir el flujo normal de

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

eventos y *Final* indica el fin del proceso, no tiene flujos de secuencia de salida y en un mismo diagrama pueden haber muchos eventos de fin.

- Actividades: Trabajo que es ejecutado dentro de un proceso de negocio. Pueden ser atómicas o no. Existen tres tipos de actividades: *Proceso* que es la actividad ejecutada dentro de la compañía, *Subproceso* definido como una actividad compuesta en la que adentro tiene en detalle un flujo de otras actividades y *Tarea* que es una actividad atómica incluida dentro de un proceso.
- Decisiones: Son lugares dentro de un proceso de negocio donde el flujo de secuencia puede tomar dos o más alternativas. Puede ser visto como una pregunta hecha en un punto del proceso, cuyas respuestas son las puertas.
- Elementos de conexión: Definen los objetos gráficos usados para conectar dos objetos juntos y como progresa el flujo dentro de un proceso. Existen dos formas de conectar objetos: Un *Flujo de secuencia o de mensajes*, el primero se usa para mostrar el orden en que las actividades se ejecutan y el segundo para mostrar el flujo de mensajes entre dos pools (calles) que están preparadas para mandarlos/recibirlos y una *Asociación* es usada para relacionar información y artefactos con objetos de flujo.
- Elementos de división: Son usados para ayudar a particionar y organizar actividades. Los *Pool* son contenedores gráficos para particionar un conjunto de actividades de otros pools, *Lanes* (sendas) es una sub-partición dentro de un pool, se extiende a toda la longitud del pool y son usadas para organizar y categorizar dentro del pool.
- Elementos de artefactos: Proveen la capacidad de mostrar información adicional acerca de los procesos. Existen tres tipos de artefactos: *Objetos de datos* proveen información acerca de lo que hace el proceso, *Grupos* proveen un mecanismo visual para agrupar elementos de un proceso informalmente y *Anotación* brinda información adicional para el lector de un diagrama BPMN.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

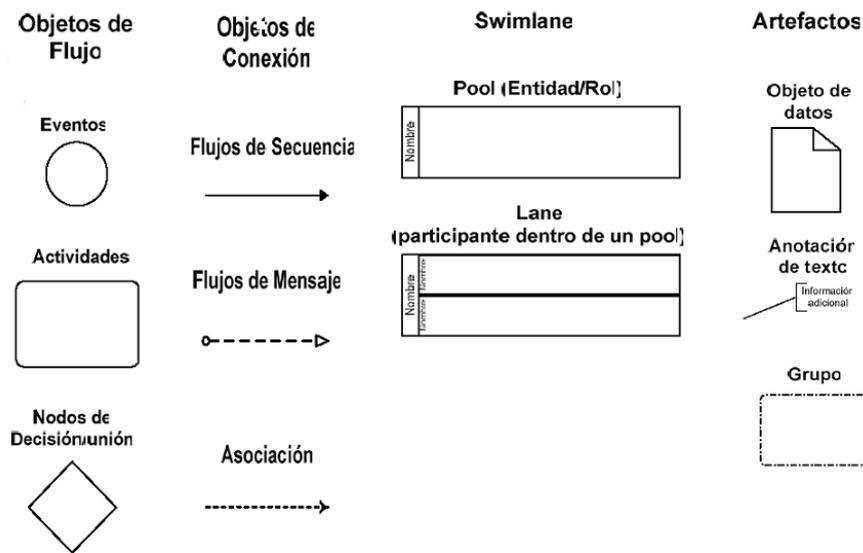


Figura 2: Elementos principales de los diagramas de BPMN.

1.5.2 Modelado del sistema

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. Es probablemente una de las innovaciones conceptuales en el mundo tecnológico del desarrollo de software que más expectativas y entusiasmos haya generado en muchos años. (Jacobson, 2000)

Considerado un estándar en la industria del software creado por Grady Booch, James Rumbaugh e Ivar Jacobson.

Entre sus principales características tenemos las siguientes: (Jacobson, 2000)

1. Lenguaje unificado para la modelación de sistemas.
2. El cliente participa en todas las etapas del proyecto.
3. Permite documentar todos los artefactos de un proceso de desarrollo.
4. Corrección de errores viables en todas las etapas.

Entre las funcionalidades que brinda UML, se puede encontrar la creación de diagramas de clases, diagramas de casos de uso, diagramas de interacción, entre otros. (Colectivo de Autores, 1999)

Los diagramas de clases están compuestos por los siguientes elementos: (Colectivo de Autores, 1999)

- Clase: Es la unidad básica que encapsula toda la información de un objeto³.
Compuesta por:

³ Objeto es una instancia de una clase.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Atributos: Características de una clase.
- Métodos: Forma en cómo la clase interactúa con su entorno.
- Visibilidad: Grado de comunicación de los atributos y métodos con el entorno.
- Relaciones: Interrelacionar dos o más clases (cada una con características y objetivos diferentes).
 - Herencia: Indica que una subclase hereda los métodos y atributos especificados por una súper clase, la subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la súper clase.
 - Composición: Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido está condicionado por el tiempo de vida del que lo incluye.
 - Agregación: Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye.
 - Asociación: Permite asociar objetos que colaboran entre sí. No es una relación fuerte, pues el tiempo de vida de un objeto no depende del otro.
 - Uso: Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase).

Los diagramas de casos de uso constan de los siguientes elementos: (Colectivo de Autores, 1999)

- Actores: Rol que un usuario juega con respecto al sistema.
- Caso de uso: Operación o tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación de otro caso de uso.
- Relaciones: Interrelacionar actores con casos de uso y casos de uso entre sí.
 - Asociación: Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.
 - Dependencia o instanciación: Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Generalización: Este tipo de relación es una de las más utilizadas, cumple una doble función dependiendo de su estereotipo, que puede ser de uso (<<uses>>) o de herencia (<<extends>>). Este tipo de relación está orientada exclusivamente para casos de uso (y no para actores).

Los componentes de un diagrama de interacción son: (Colectivo de Autores, 1999)

- Un objeto o actor.
- Mensaje de un objeto a otro objeto.
- Mensaje de un objeto a sí mismo.

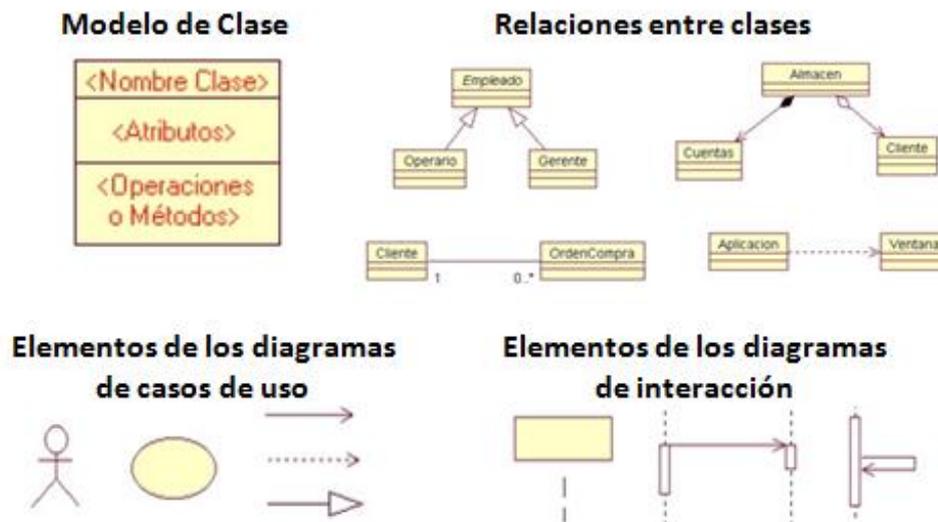


Figura 3: Elementos que componen a los diagramas de clases, de casos de uso y de interacción.

1.6 HERRAMIENTA CASE PARA EL DESARROLLO DE SOFTWARE

Las herramientas CASE son aplicaciones que facilitan el desarrollo del software, reduciendo el esfuerzo, el costo y el tiempo, además de que estructuran la documentación asociada a los artefactos generados.

El beneficio adicional obtenido por la utilización de un CASE actual (si se compara con la utilización de una metodología sin el uso de un CASE) se representa en los siguientes aspectos: (Departamento de Sistemas Informáticos y Computación., 2002)

- Facilita la verificación y mantenimiento de la consistencia de la información del proyecto.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Facilita el establecimiento de estándares en el proceso de desarrollo y documentación.
- Facilita el mantenimiento del sistema y las actualizaciones de su documentación.
- Facilita la aplicación de las técnicas de una metodología.
- Disponibilidad de funciones automatizadas tales como: obtención de prototipos, generación de código, generación de pantallas e informes, generación de diseños físicos de bases de datos y verificadores automáticos de consistencia.
- Facilita la aplicación de técnicas de reutilización y reingeniería.
- Facilita la planificación y gestión del proyecto informático.

1.6.1 Selección de la herramienta CASE para el desarrollo de software

Visual Paradigm (VP), es la herramienta CASE seleccionada para modelar todos los diagramas del sistema. Entre las principales funcionalidades del Visual Paradigm 5.0 se encuentran capturar, diseñar, gestionar y documentar los artefactos del software en un único entorno compatible con varias metodologías de desarrollo como RUP y XP y generar diagramas de proceso de negocios y UML.

Tiene la ventaja de ser multiplataforma, soporta el ciclo de vida completo del desarrollo de software. Es una herramienta de gran utilidad para el proyecto, por su nivel de integración con el entorno de desarrollo y por la posibilidad que brinda de trabajar sobre la plataforma libre GNU/Linux, sistema operativo sobre el cual se desarrolla el proyecto. (Equipo de Desarrollo de VP, 2012)

Otras de las funcionalidades brindadas por la herramienta son las siguientes:

- Administración de requisitos.
- Modelado de procesos del negocio.
- Modelado de base de datos.
- Generación de código desde diagramas.
- Generación de documentación.
- Ingeniería inversa.

Entre las funcionalidades de VP se encuentra la de crear prototipos de interfaz de usuario. En un ambiente sencillo se pueden definir todos los elementos que

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

conforman una interfaz. Permite insertar información adicional a los diagramas mediante notas y comentarios para describir sus elementos lo que facilita la revisión de los prototipos así como el trabajo en equipo.

Después de realizar un análisis, por parte de los analistas y el equipo de desarrollo, de las herramientas para el modelado de prototipos de interfaz de usuario se puede concluir que VP 5.0 resuelve la necesidad de una herramienta que facilite el diseño de interfaces de usuario en la realización de los prototipos.

1.7 GESTOR DE PROCESOS

El gestor de procesos evaluado y escogido por el equipo de desarrollo fue Bonita Open Solution (BOS) 5.6 por adecuarse a las características de la solución que se desea obtener. Esta herramienta se destaca sobre todo por la facilidad en su utilización debido al diseño intuitivo de los diferentes elementos que lo componen y por el bajo coste de su implantación (al no requerir ninguna inversión adicional, solamente tiempo de aprendizaje).

Entre sus características más notables se observan: (Equipo de desarrollo BOS, 2011)

- Open Source.
- Ligera.
- Compatible con BPMN 2.0.
- Interfaz intuitiva.
- Potente.
- Fácil importación de procesos desde otras herramientas.
- Personalización de las herramientas.
- Integración de los procesos con reglas de negocio.
- Conectores propios (realizados por la comunidad de BonitaSoft).

Bonita Open Solution está compuesto por 4 módulos: (Equipo de desarrollo BOS, 2011)

1. Bonita Execution Engine: Es el motor de procesos de Bonita y se encarga de la conexión de los procesos que existen en el sistema así como el despliegue y ejecución de los procesos.
2. Bonita Studio: Es la aplicación gráfica que tiene como función el diseño de los procesos usando la notación BPMN sobre un área de diseño (pizarra) de

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

forma muy intuitiva basada en "arrastrar" los elementos y en su configuración lo especifica mediante una o varias pestañas habilitadas para ello.

3. Bonita Form Builder: Se encarga de mostrar los formularios a los usuarios de la aplicación, ya que muchos de los pasos que se producen en un proceso requieren de la entrada de datos por parte del usuario implicado.
4. Bonita User Experience (User XP): Es la encargada de la gestión de todo lo relacionado con los procesos desplegados.

Como se había mencionado anteriormente el sistema a desarrollar estará caracterizado marcadamente por un enfoque a procesos, por lo que para la implementación se decidió utilizar por parte del equipo de desarrollo el gestor de procesos de negocio BOS en su versión 5.6. El mismo ofrece algunas ventajas que son significativas en el ámbito del proyecto:

- Cambios en los procesos: Los procesos de las organizaciones cambian continuamente con el objetivo de mejorar los mismos o por decisiones a nivel legal, institucional o gubernamental. BOS permite modificar el flujo de un proceso previamente diseñado con mucha facilidad, concretamente, modificando un diagrama de procesos en notación BPMN. Se pueden cambiar el orden de las actividades, el sentido del flujo entre las mismas y otros elementos sin afectar directamente las funcionalidades del sistema asociadas a dichas actividades.
- Generación de interfaces y manejo básico de datos: Permite generar automáticamente las interfaces de usuario (particularmente los formularios de entrada de datos) y el flujo básico de movimiento entre las mismas. Los formularios de captura y recuperación de datos llevan también asociadas variables para la captura de los datos a nivel de actividades y a nivel de procesos. Las plantillas predeterminadas pueden ser modificadas haciendo uso de *HTML (Hypertext Markup Language, Lenguaje de Formato de Documentos de Hipertexto)*, *CCS (Cascading Style Sheets, Hojas de Estilo en Cascada)* y Java Script.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.8 LENGUAJES DE PROGRAMACIÓN

Los lenguajes de programación constituyen un conjunto de sintaxis y reglas semánticas que definen los programas para ordenadores. Es una técnica estándar de comunicación para entregarle instrucciones al computador. Un lenguaje le da la capacidad al programador de especificarle al computador, qué tipo de datos actúan y qué acciones tomar bajo una variada gama de circunstancias, utilizando un lenguaje relativamente próximo al lenguaje humano. (UTN, 2012)

En la actualidad se destacan potentes lenguajes de programación como Java, CSharp, C++, Phyton y *PHP (Hypertext Preprocessor)*. Cada uno de ellos con sus propias características que los hacen únicos según el objetivo con que se empleen.

1.8.1 Selección del lenguaje de programación que será utilizado en la solución

Después de analizar varios lenguajes de programación, teniendo en cuenta las necesidades del cliente y el tipo de aplicación a realizar, se decidió utilizar Java como el lenguaje de programación para implementar la solución. Además su selección está condicionada por el conocimiento y dominio que ya poseen los desarrolladores acerca de este lenguaje.

Java es un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. (Java, 2012)

Dentro de sus características más significativas se encuentran las siguientes: (Java, 2012)

- Simple: Elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos.
- Orientado a objetos: En Java, todo es un objeto. Estos se definen en clases, con atributos y métodos, agrupándose en paquetes, y exponiendo interfaces, permitiendo herencia y polimorfismo, implementando así la tecnología de objetos de una manera muy completa, y a la vez sencilla y agradable.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Distribuido: Java dispone de una serie de librerías para que los programas se puedan ejecutar en varias máquinas y puedan interactuar entre sí.
- Robusto: Java está diseñado para crear software altamente fiable.
- Seguro: Java cuenta con ciertas políticas que evitan que se puedan codificar virus con este lenguaje, sin olvidar que existen otras restricciones que limitan lo que se puede o no hacer con los recursos críticos de una máquina.
- Portable: Como el código compilado de Java (conocido como byte code) es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el intérprete de Java.
- Independiente de la Arquitectura: El proceso de compilación no produce código para un procesador en particular, sino que genera código para un procesador ideal, denominado máquina virtual de Java, lo que le permite poder ejecutarse en cualquier plataforma, sin necesidad de cambiar el código fuente, ni recompilar.

1.8.2 Selección de la herramienta de desarrollo

NetBeans IDE es una herramienta diseñada para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Es un entorno de desarrollo libre y gratuito, escrito en Java y además presenta particularidades perfectamente ajustables a cualquier otro lenguaje de programación.

Entre sus características más notables se pueden encontrar las siguientes: (ABCdatos, 2011) (Corporación Oracle, 2012)

- Fácil de instalar, de uso inmediato y ofrece características tales como funciones de edición enriquecida de JavaScript, soporte para el uso del framework web Spring e integración con MySQL.
- NetBeans IDE 7.1 es la última versión estable del entorno de desarrollo integrado que ofrece características para ayudar en la construcción y mantenimiento de aplicaciones web y móviles.
- Entre los cambios implementados en NetBeans IDE 7.1 se destaca el soporte para JavaFX 2.0 con lo que es posible llevar a cabo los procesos completos de compilación y depuración de las aplicaciones desarrolladas en esta plataforma, soporte para CSS3, soporte para *JDK7 (Java Development*

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Kit, Kit de Desarrollo de Java), mejoras en el constructor de interfaces Swing, mejoras en el debug de PHP o integración con Git⁴, entre otras.

- Admite el Oracle WebLogic Server 12c, lo cual incluye el soporte de Java Platform Enterprise Edition 6 (Java EE) para los desarrolladores que crean aplicaciones en la plataforma Oracle WebLogic Server.

Teniendo en cuenta las funcionalidades y beneficios de NetBeans IDE 7.1, además de la facilidad de su uso y la forma en que se ajusta a las necesidades de la aplicación que se quiere desarrollar, se decidió seleccionarla como la herramienta que en conjunto con las demás permitirá dar solución a la necesidad de los clientes de la Cámara de Comercio.

1.9 PLATAFORMA EMPRESARIAL DE JAVA

Java Platform, Enterprise Edition o Java EE (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4), es una plataforma de programación, parte de la Plataforma Java, para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java con arquitectura de N capas distribuidas y que se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. La plataforma Java EE está definida por una especificación similar a otras especificaciones del Java Community Process, Java EE es también considerada informalmente como un estándar debido a que los proveedores deben cumplir ciertos requisitos de conformidad para declarar que sus productos son conformes a Java EE; estandarizado por The Java Community Process / JCP. (Oracle, 2011)

Java EE incluye varias especificaciones de *API (Application Programming Interface, Interfaz de Programación de Aplicaciones)*, tales como JDBC (Java Database Connectivity), RMI (Remote Method Invacation), e-mail, JMS (Java Message Service, servicio de mensajes Java), Servicios Web, XML (Extensible Markup Language, lenguaje de marcas extensible), etc y define cómo coordinarlos. También configura algunas especificaciones únicas para Java EE para componentes, estas incluyen Enterprise JavaBeans, servlets, portlets⁵ (siguiendo la especificación de portlets Java), *JSP (Java Server Pages)* y varias

⁴ Es un software de sistema de control de versiones diseñado por Linus Torvalds.

⁵ Componentes modulares de las interfaces de usuarios gestionadas y visualizadas en un portal web.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

tecnologías de servicios web. Ello permite al desarrollador crear una Aplicación de Empresa portable entre plataformas y escalable, y a la vez que sea integrable con tecnologías anteriores. Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, significando que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes en lugar de las tareas de mantenimiento de bajo nivel. (Oracle, 2011)

1.10 SERVIDOR DE APLICACIONES

Tomcat es un servidor web con soporte de servlets⁶ y JSPs desarrollado por la fundación Apache, es multiplataforma y funciona en cualquier sistema operativo que disponga de la JVM. Incluye un compilador Jasper, que compila JSPs convirtiéndolas en servlets. El motor de servlets de Tomcat en ocasiones se presenta en combinación con el servidor web Apache aunque Tomcat puede funcionar como servidor web por sí mismo. En sus inicios el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones. En la actualidad es utilizado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad. (Apache Foundation, 2011)

Entre las principales funcionalidades que brinda la versión 6.0 de Tomcat se encuentran: (Apache Foundation, 2011)

- Implementado de servlet 2.5 y JSP 2.1.
- Soporte para Unified Expression Language 2.1.
- Diseñado para funcionar en Java SE 5.0 y posteriores.
- Recarga de servlets.
- Funciones básicas HTTP (Hypertext Transfer Protocol, protocolo de transferencia de hipertexto).

Teniendo en cuenta que las características de esta versión de Tomcat se ajustan a las necesidades del producto a desarrollar se decidió utilizar el mismo como servidor Web.

⁶ Programa ejecutado en el servidor.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.11 TECNOLOGÍA JPA

Se seleccionó la tecnología *JPA (Java Persistence API)* por ser el estándar Java para hacer mapeo objeto-relacional. El mapeo objeto-relacional permite acceder a los datos de una base de datos relacional con un estilo orientado a objetos. (OpenXava, 2011)

JPA proporciona un modelo de persistencia basado en POJO's (Plain Old Java Object) para mapear bases de datos relacionales en Java. JPA fue desarrollado por el grupo de expertos de *EJB 3.0 (Enterprise JavaBeans)* como parte de *JSR 220 (Java Specification Request)*, aunque su uso no se limita a los componentes software EJB. También puede utilizarse directamente en aplicaciones web y aplicaciones clientes; incluso fuera de la plataforma *Java EE (Enterprise Edition)*, por ejemplo, en aplicaciones *Java SE (Standard Edition)*. (JAVA J2EE, 2009)

En su definición, se han combinado ideas y conceptos de los principales frameworks de persistencia como Hibernate, Toplink y *JDO (Java Data Objects)*, y de las versiones anteriores de EJB. Todos estos cuentan actualmente con una implementación JPA. (JAVA J2EE, 2009)

Java Persistence API consta de tres áreas:

- El Java Persistence API.
- El lenguaje de consultas.
- El mapeo de los metadatos objeto/relacional.

Básicamente JPA es una API para *ORMs (Object-Relational mapping)* y permite: (Compujuy, 2010)

- Abstraerse del proveedor de persistencias.
- Eliminar la necesidad de escribir el *SQL (Structured Query Language, lenguaje de consulta estructurado)* que mapea la Base de Datos a los objetos utilizando simples anotaciones.
- Crear relaciones entre cualquier cantidad de objetos solo con anotaciones.
- Controlar la transaccionalidad ayudando en el tema de la concurrencia y en la forma en que se gestiona.
- Presenta un caché⁷ de primer nivel que evita consultar la Base de Datos en reiteradas ocasiones para acceder a los mismos datos.

⁷ Sistema especial de almacenamiento de alta velocidad.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- Utiliza un caché de segundo nivel que incrementa el rendimiento de la aplicación y su escalabilidad.
- Lenguaje de consultas *OO (Orientado a Objetos)* fácil de utilizar que permite consultar las tablas como si fuesen objetos.

1.12 FRAMEWORK

Framework es un concepto sumamente genérico, se refiere a “ambiente de trabajo, y ejecución”. (...) En general los frameworks son soluciones completas que contemplan herramientas de apoyo a la construcción (ambiente de trabajo o desarrollo) y motores de ejecución (ambiente de ejecución). (SOA Agenda, 2011)

1.12.1 Spring

El Framework Spring ayuda a construir aplicaciones Java robustas utilizando menos tiempo, permite al desarrollador enfocarse en el proceso de negocio y abstraerse del código que se encarga de implementar las conexiones entre los componentes y el sistema. Spring ofrece un amplio soporte para la integración de componentes que exigen las aplicaciones de hoy en día, proporcionando un modelo sencillo para la creación de soluciones de integración empresarial. Spring Security proporciona un marco de seguridad más robusta disponible para los desarrolladores de Java, proporciona *SPI (Security Protocol Implementation)* para manejar la autorización y la autenticación con numerosas implementaciones prácticas e integra en consonancia todas las tecnologías de Spring, incluyendo Spring *MVC (Modelo Vista Controlador)*, Spring Integration, Spring Web Services, y Spring Flex. (Spring Source, 2011)

Entre los patrones de diseño implementados por Spring se encuentra Inversion of Control y el Dynamic proxy. Inversion of Control se ocupa de proporcionar un medio normalizado para la composición de componentes dispares en una aplicación completamente funcional, lista para usar donde el Framework de Spring codifica los patrones de diseño formalizados como objetos primarios que se puede integrar en una misma aplicación. Dynamic proxy se utiliza a menudo para implementar el patrón decorator. Un ejemplo de esto es cuando ciertas funcionalidades se implementan en una clase y se utiliza en distintos lugares del

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

código en la *Programación Orientada a Aspectos (AOP)* de Spring. (Spring Source, 2011)

1.13 PATRONES

Se conoce como patrón al conjunto de elementos que forman una unidad diferenciada y que se repiten a lo largo del tiempo, por lo que pueden tomarse como modelo o punto de referencia: un patrón de comportamiento. (Diccionario Manual de la Lengua Española Vox, 2012)

Un patrón es la abstracción de una forma concreta el cual tiende a repetirse en un contexto específico no-arbitrario. (Bizagi, 2009)

1.13.1 Patrones de modelado de procesos

A continuación se ilustra la implementación de los patrones bajo el estándar de BPMN. Estos patrones se encuentran agrupados de la siguiente manera: (Bizagi, 2009)

1. Patrones básicos de control de flujo.
2. Patrones de sincronización y enrutamiento avanzado.
3. Patrones estructurales.
4. Patrones que involucran múltiples instancias.
5. Patrones que se basan en el estado del sistema.
6. Patrones de cancelación.

Para modelar los procesos de negocio, se utilizaron los siguientes patrones con sus diferentes especificaciones:

Patrones básicos de control de flujo

Secuencia: Patrón básico de todo flujo de trabajo. Se requiere cuando hay una dependencia entre dos actividades, de tal forma que una actividad no pueda iniciarse antes de que otra haya terminado. (Bizagi, 2009)

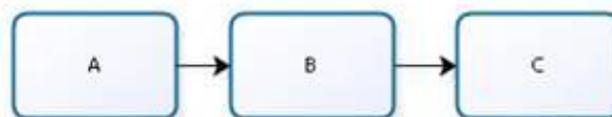


Figura 4: Patrón secuencia.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Patrones estructurales

Ciclos arbitrarios: Este patrón es un punto en el proceso en donde una o más actividades pueden ser ejecutadas repetidamente. No hay restricciones estructurales en los ciclos. (Bizagi, 2009)

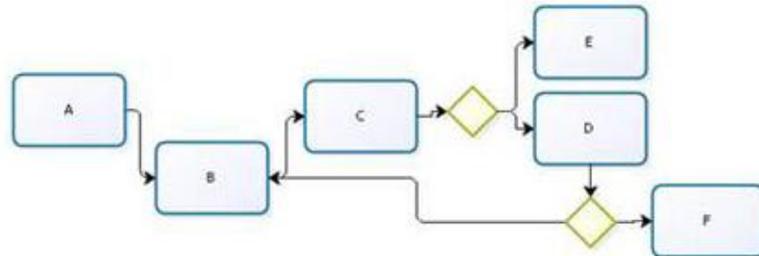


Figura 5: Patrón ciclos arbitrarios.

Terminación implícita: Un subproceso debería terminar cuando no hay nada más por hacer. En otras palabras, no hay actividades activas y ninguna actividad puede pasar a activa. (Bizagi, 2009)

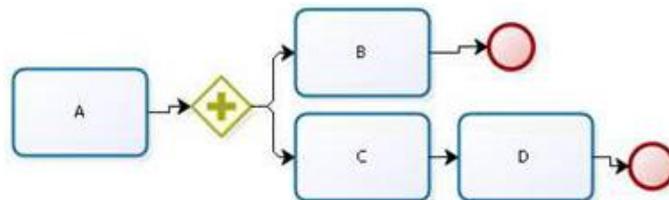


Figura 6: Patrón terminación implícita

1.13.2 Patrones de casos de uso

Para modelar el diagrama de casos de uso del sistema, se utilizaron los siguientes patrones con sus diferentes especificaciones:

Concordancia: Reutilización (Commonality: Reuse): Re-uso es un patrón de estructura que consiste en tres casos de uso. El primero llamado "Sub-secuencia Común", modela la secuencia de acciones que aparecen en múltiples casos de uso del modelo. Los otros dos casos de uso comparten de esta sub-secuencia común de acciones (dos es la menor cantidad que puede existir). La sub-secuencia tiene que estar en un fragmento, es decir, todo lo que requiere estar incluido tiene que estar en un único fragmento completo. Además no se puede hacer referencia desde la sub-secuencia a donde esta es utilizada, porque el caso de uso incluido tiene que ser independiente del caso de uso base. (Gunnar Övergaard, 2004)

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Extensión o Inclusión Concreta: Extensión (Concrete Extension or Inclusion: Extension): Extensión consiste en dos casos de uso y una relación de extensión entre ellos. El caso de uso extendido es concreto; es decir, puede ser instanciado por el mismo, así como también puede extender del caso de uso base. El patrón es aplicable cuando un flujo puede extender el flujo de otro caso de uso, así como también puede ser realizado por el mismo. (Gunnar Övergaard, 2004)

Extensión o Inclusión Concreta: Inclusión (Concrete Extension or Inclusion: Inclusion): En este patrón, existe una relación de inclusión desde el caso de uso base al caso de uso incluido. El último puede ser instanciado por el mismo. El caso de uso base podría ser cualquiera, concreto o abstracto. El patrón es usado cuando un flujo podría ser incluido en el flujo de otro caso de uso y además puede ser realizado por el mismo. (Gunnar Övergaard, 2004)

CRUD: Completo: Consiste en un caso de uso, llamado Gestionar Información, modelando todas las diferentes operaciones que pueden ser realizadas en una pieza de información de cierto tipo, como crearla, leerla, actualizarla y eliminarla. (Gunnar Övergaard, 2004)

Actores Múltiples: Roles comunes (Multiple Actors: Common Roles): Consiste en un caso de uso y al menos dos actores. A un caso de uso ingresan más de dos actores y estos tienen un rol común. (Gunnar Övergaard, 2004)

1.14 CONCLUSIONES

En este capítulo se definieron las metodologías, las tecnologías y herramientas a utilizar en el diseño de la solución. Se arribó a las siguientes conclusiones:

- Emplear RUP como metodología de desarrollo de software en sus flujos de trabajo Modelado de negocio, Requisitos, Diseño, Implementación y Prueba permitió seguir un proceso de desarrollo conocido, que a través de una secuencia de actividades indica qué hacer, cómo y en qué momento, obteniendo como resultado un grupo de artefactos, los cuales fueron visualizados, especificados, construidos y documentados a través BPMN como notación para el modelado de procesos de negocio y UML para el modelado del sistema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

- El estudio del funcionamiento de sistemas con características semejantes al proceso de Afiliación de Empresas Cubanas permitió obtener una visión general de su uso a nivel internacional.
- Aplicando los criterios de selección especificados se obtuvieron las herramientas apropiadas para el desarrollo del proceso Afiliación de Empresas Cubanas del módulo Registral del sistema SIRECC.
- Aplicar patrones de modelado bajo el estándar de BPMN y patrones de casos de uso permitió la identificación y elaboración de los diagramas de procesos del negocio y el diagrama de casos de uso del sistema.

Estas conclusiones sirven de guía para continuar con el desarrollo de la solución al problema planteado.

CAPÍTULO 2: REQUISITOS Y DISEÑO

2.1 INTRODUCCIÓN

En el presente capítulo se describe la solución que se propone para la realización de la ingeniería de requisitos y el diseño del proceso Afiliación de Empresas Cubanas del módulo Registral del proyecto SIRECC. Se exponen las funcionalidades y restricciones del sistema mediante los requisitos funcionales y no funcionales. Luego se muestra el análisis del sistema correspondiente a dichos requisitos, mediante el modelo de sistema; y se analizan los casos de uso y sus especificaciones para modelar los artefactos imprescindibles para obtener un adecuado modelo de diseño.

2.2 MODELO DE PROCESO DEL NEGOCIO

El proceso Afiliación de Empresas Cubanas es el encargado de gestionar la información de cada una de las empresas que solicitan afiliarse a la Cámara de Comercio. Una vez presentada esta solicitud se procede a aprobar la documentación que se le exige a la empresa como parte del proceso de afiliación, los documentos que deben presentar son: Solicitud de afiliación, Ficha de la Entidad, Copia de Resolución de constitución de la empresa, Copia de Resolución de nombramiento del Director, y en el caso de las empresas radicadas en provincia deben presentar además la Argumentación con Criterio del Delegado; una vez revisados y aceptados estos documentos se procede a aprobar la afiliación, subproceso en el cual se elaboran cada uno de los documentos que dan validez a la afiliación. Los documentos son: Argumentación, Carta de aprobación, Criterio de la junta sobre admisión de asociación y Certificado de registro, cada uno de ellos conformado y firmado por los funcionarios de la cámara encargados de esa parte del proceso. Una vez que la empresa queda afiliada a la cámara debe brindar sus datos de contacto, datos de registro y datos de actividad para que los mismos sean registrados. Terminando el proceso con la notificación a la empresa del estado de su solicitud de afiliación.

CAPÍTULO 2: REQUISITOS Y DISEÑO

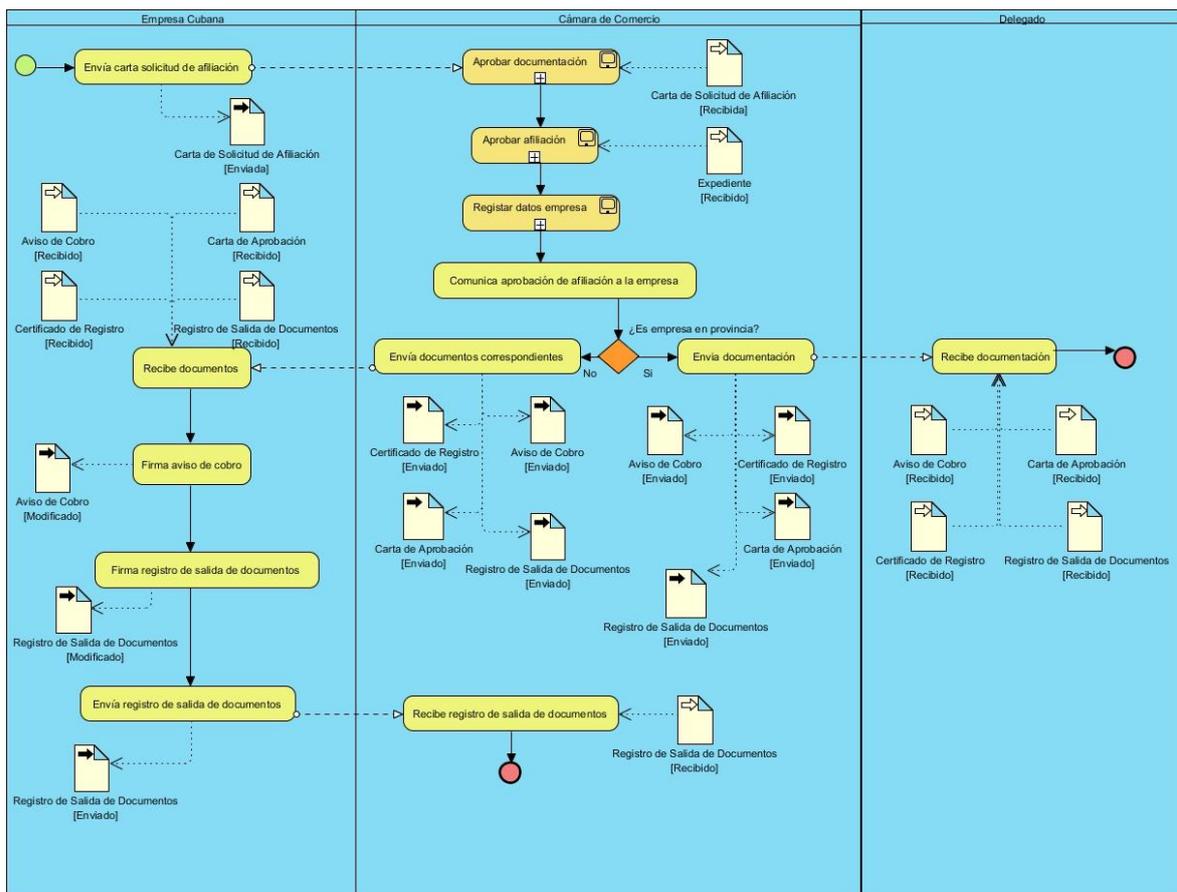


Figura 7: Diagrama de proceso de negocio: Registrar afiliación de empresas cubanas.

Para informarse de la descripción detallada de cada una de las tareas del proceso Registrar afiliación de empresas cubanas y los subprocesos, Aprobar documentación, Aprobar afiliación, Realizar asentamiento jurídico y Registrar datos de empresa, consultar el documento Modelo de Procesos de Negocio, ver **Anexo 1**.

2.3 REQUISITOS DE SOFTWARE

Los requisitos se obtienen de recopilar, analizar y verificar las necesidades del cliente para un sistema. Estos son analizados y agrupados en funcionales y no funcionales. Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. Los requisitos no funcionales son propiedades o cualidades que el producto debe tener.

Para la captura de requisitos se aplicaron las siguientes técnicas:

- **Entrevistas:** La entrevista consiste en el intento sistemático de recoger información de otra persona a través de una comunicación interpersonal que

CAPÍTULO 2: REQUISITOS Y DISEÑO

se lleva a cabo por medio de una conversación estructurada, técnica que fue empleada en las visitas a la cámara.

- **Tormenta de ideas:** La tormenta de ideas es una técnica simple para la generación de ideas. Se reúnen un conjunto de personas relacionadas al producto y éstas sugieren y exploran sus ideas libremente. Con el empleo de esta técnica se generaron una amplia variedad de puntos de vista, se estimuló el pensamiento creativo, se logró construir una imagen más completa del problema y se proveyó un ambiente social mucho más confortable y de fácil aprendizaje.
- **Prototipos:** Un prototipo es una pequeña muestra, de funcionalidad limitada, de cómo sería el producto final una vez terminado. Ayuda a conocer la opinión de los usuarios y rectificar algunos aspectos antes de llegar al producto terminado. Los prototipos no funcionales de la aplicación fueron presentados a los funcionarios de la cámara quienes quedaron satisfechos con las propuestas.
- **Talleres:** Los requisitos tienen frecuentemente implicaciones cruzadas desconocidas para las personas implicadas individuales y que en ocasiones no se descubren en las entrevistas o quedan definidas de manera incompleta durante la misma. Estas implicaciones cruzadas se descubrieron realizando talleres facilitados por el analista del negocio, en donde las personas involucradas participaron en discusiones para descubrir requisitos, analizando sus detalles y las implicaciones cruzadas.
- **Arqueología de documentos:** Con la aplicación de esta técnica se tratan de determinar posibles requisitos sobre la base de inspeccionar la documentación utilizada por la empresa. En el caso del proceso Afiliación de Empresas Cubanas se consultaron los expedientes y documentos que se generan durante el proceso, los cuales ayudaron a obtener información que de otra manera sería sumamente difícil conseguir.

2.3.1 Requisitos funcionales

- **RF_2:** Registrar datos de la empresa: El sistema registra los datos que intervienen en el momento de guardar los datos de la empresa: Datos de contacto (entidad, siglas, directivo, cargo del directivo, representante, cargo

CAPÍTULO 2: REQUISITOS Y DISEÑO

de representante, teléfono, móvil, fax, email, sitio web, provincia, municipio, dirección), Datos de registro (nombre oficial, siglas oficiales, año de creación, número de tarjeta, ministerio, delegación, años de establecimiento, objeto social, mixta, ingreso anual (Moneda Nacional), ingreso anual (Moneda libremente convertible), cuota, CUP, deudora), Datos de actividad (Calificadores (productor, comercializador, importador, exportador, servicios), sectores, anotaciones, secciones y subsecciones). Además se registra el estado de la empresa.

- **RF_14:** Buscar empresa: El sistema permitirá buscar una empresa según los criterios de búsquedas especificados.
- **RF_15:** Listar empresas: El sistema permitirá listar todas las empresas registradas según la búsqueda realizada.
- **RF_16:** Cancelar empresa: El sistema permitirá modificar el estado de una empresa registrada a pasiva cuando deje de ser miembro de la cámara.
- **RF_17:** Mostrar datos de la empresa: El sistema permitirá mostrar los siguientes datos que se tienen registrados de la empresa seleccionada: Datos de contacto, Datos de registro y Datos de la actividad.
- **RF_21:** Aprobar o denegar afiliación de empresa: El sistema permite aprobar o denegar la afiliación de la empresa.
- **RF_25:** Modificar datos de la empresa: El sistema permitirá actualizar los datos de la empresa, teniendo como punto de partida los datos ya insertados los cuales también se pueden modificar.

Se obtuvieron 33 requisitos funcionales, algunos de ellos mostrados ya anteriormente, los cuales se seleccionaron por ser los más importantes para el proceso y los más significativos para el cliente. Para obtener más información de la descripción detallada de todos los requisitos, consultar el documento Especificación de requisitos de software, ver **Anexo 2**.

2.3.2 Requisitos no funcionales

Usabilidad

- **RnF1:** El sistema será usado por el personal de la Cámara de Comercio que se encuentre involucrado en los procesos de Registro de Afiliación de

CAPÍTULO 2: REQUISITOS Y DISEÑO

Empresas Cubanas, Registro de Agencias de Viajes, Registro de Importadores y Exportadores y Registro de Sucursales Extranjeras.

- **RnF3:** El producto a entregar será una aplicación web enfocada en las necesidades de la Cámara de Comercio de la República de Cuba.

Confiabilidad

- **RnF7:** El tiempo máximo de inactividad es de 5 minutos, y una vez que el sistema quede inactivo el usuario deberá autenticarse nuevamente.

Eficiencia

- **RnF11:** El número de usuarios concurrentes que el sistema puede alojar es 30.

Soporte

- **RnF12:** El uso del sistema requerirá un tiempo de preparación previa por los usuarios finales para su correcta explotación.

Interfaz

- **RnF18:** El sistema deberá tener un diseño de interfaz de usuario basado en las características de BonitaSoft.

Seguridad

- **RnF23:** El sistema brindará la posibilidad de establecer permisos sobre acciones, garantizando que solo acceda a la información quien esté autorizado.

Se obtuvieron un total de 25 requisitos no funcionales algunos de ellos mostrados ya anteriormente, para más información de la descripción detallada de cada uno de ellos, consultar el documento Especificación de requisitos de software, ver **Anexo 2**.

2.4 MODELO DEL SISTEMA

Este modelo permite que los desarrolladores del software y los clientes lleguen a un acuerdo sobre las condiciones y posibilidades que este debe cumplir. Entre los principales objetivos que tiene el modelo está identificar los requisitos funcionales y no funcionales. Luego se definen los actores, casos de uso y la descripción detallada de cada uno de ellos, que junto al diagrama de casos de uso del sistema, permitirá el posterior modelado del diseño.

CAPÍTULO 2: REQUISITOS Y DISEÑO

2.4.1 Actores del sistema

Un actor no es más que un conjunto de roles que los usuarios desempeñan cuando interactúan con los casos de uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que se identificaron los actores del sistema, se tiene identificado el entorno externo del sistema (Pressman, 2005). Los actores del sistema que interactúan con las funcionalidades del proceso Registrar Afiliación de Empresas Cubanas son:

Actor	Descripción
Especialista en Desarrollo Empresarial	Encargado de revisar la documentación que se presenta al comenzar el proceso de afiliación, y elaborar el documento Argumentación si aprueba la documentación presentada por la empresa.
Director de Formación y Membrecía	Encargado de aprobar la documentación y elaborar el documento Criterio de admisión y Carta de aprobación una vez que se decida aprobar la afiliación.
Junta Ejecutiva	Encargado de revisar el expediente con los documentos anexados y emitir su aprobación o no de la empresa.
Abogado	Encargado de realizar el asentamiento jurídico de la empresa, proceso durante el cual elabora el documento Certificado de Registro.
Usuario	Es un actor común del cual heredan los demás actores del sistema. Además es el encargado de generar todos los reportes que brinda la aplicación.
Departamento de Formación y Membrecía	Es un actor común del cual heredan los actores Director de Formación y Membrecía y Especialista en Desarrollo Empresarial, es el encargado de gestionar los datos de la empresa y actualizar los datos del expediente.

Tabla 1: Descripción de los actores del sistema.

2.4.2 Diagrama de casos de uso del sistema

Luego de identificados los requisitos, los actores y los casos de usos del sistema, se estructura el diagrama de casos de usos del sistema, que representa la relación de los actores del sistema con los casos de uso. Fueron identificados un total de 22 casos de uso. A continuación se muestra el diagrama representado en

CAPÍTULO 2: REQUISITOS Y DISEÑO

forma de paquetes, donde cada paquete contiene varios casos de uso y se muestra su relación con el actor del sistema correspondiente. Para consultar el diagrama con los casos de usos de cada paquete, ver **Anexo 3**.

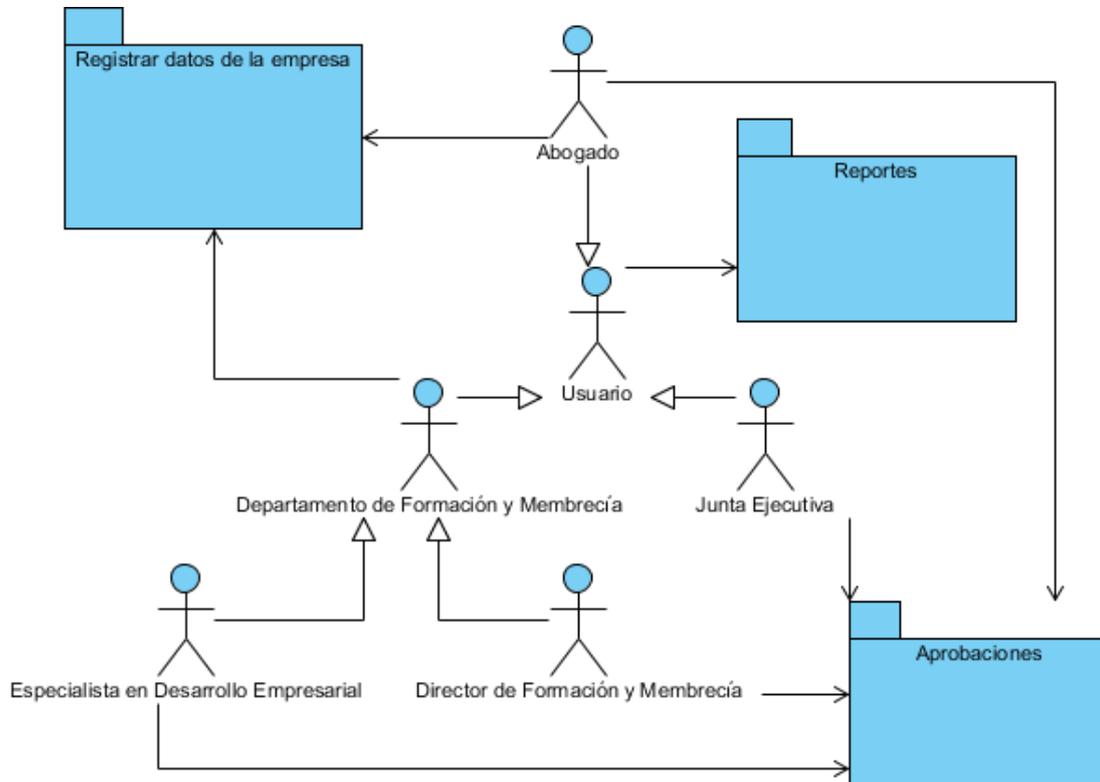


Figura 8: Diagrama de casos de uso del sistema.

2.4.3 Descripción de casos de uso del sistema

Mediante la descripción del siguiente caso de uso se especifica la secuencia de eventos que los actores utilizan para completar un proceso a través del sistema. Para consultar todas las descripciones de casos de uso, revisar el documento Especificación de casos de uso, ver **Anexo 4**.

A continuación, la descripción del caso de uso Gestionar empresa:

Caso de Uso:	Gestionar empresa.
Actores:	Departamento de Formación y Membrecía.
Resumen:	El caso de uso se inicia cuando un usuario necesita gestionar una empresa. El usuario selecciona la opción correspondiente al registro o modificación de una empresa y llena los campos requeridos para la creación en caso que no exista una anteriormente o para la modificación de la existente.

CAPÍTULO 2: REQUISITOS Y DISEÑO

Precondiciones:	El usuario debe estar registrado en el sistema.	
Referencias	RF2, RF16, RF22.	
Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El usuario selecciona la opción que le permite Gestionar una empresa.	2. El sistema muestra una interfaz que permite realizar las siguientes operaciones: Adicionar empresa, ver sección Adicionar. Modificar empresa, ver sección Modificar. Mostrar datos de empresa, ver sección Mostrar datos de empresa.	
Sección: “Adicionar”		
Acción del Actor	Respuesta del Sistema	
1. El usuario elige la opción Adicionar. Si desea cancelar la operación, ver flujo alternativo 1.	2. El sistema muestra una interfaz que permite registrar los datos de la empresa.	
3. El usuario introduce los datos correspondientes a los campos: <i>Entidad, Siglas, Directivo, Cargo del directivo, Representante, Cargo del representante, Teléfono, Dirección, Móvil, Fax, Email, Sitio web, Municipio, Provincia, Nombre oficial, Siglas oficiales, Año de creación, Año de establecimiento, Número de tarjeta, Ingreso anual (MN), Ingreso anual (CUC), Objeto social, Cuota, Ministerio, Delegación, Mixta, CUP, Subsección, Anotaciones, Productor,</i>		

CAPÍTULO 2: REQUISITOS Y DISEÑO

<i>Comercializador, Importador, Exportador, Servicios, Sector, Sección.</i>	
4. El usuario elige la opción Registrar .	5. El sistema guarda los datos de registro de la empresa y genera el estado creado de la misma.
6. El usuario elige la opción gestionar expediente.	7. El sistema muestra la interfaz que permite gestionar un expediente. Ver caso de uso incluido Gestionar expediente . Terminando así el caso de uso.

Prototipo de Interfaz

Cámara de Comercio de la República de Cuba user | Logout

Aprobar afiliación

Gestionar empresa

Nombre de empresa

Siglas

Provincia

Ministerio

Cuota

Buscar

Listado de empresas

No expediente	Entidad	Provincia	Ministerio	Estado	Cuota	Objeto social
1252-01	Desoft	Habana	Informática	Creada	10000	la empresa....

Adicionar **Modificar** **Cancelar** **Cerrar**

CAPÍTULO 2: REQUISITOS Y DISEÑO

 **Cámara de Comercio de la República de Cuba** user
Logout

Registrar datos de la empresa

Datos de Contacto

Entidad	<input type="text"/>	Móvil	<input type="text"/>
Siglas	<input type="text"/>	FAX	<input type="text"/>
Directivo	<input type="text"/>	Email	<input type="text"/>
Cargo del Directivo	<input type="text"/>	Sitio Web	<input type="text"/>
Representante	<input type="text"/>	Provincia	<input type="text"/>
Cargo del Representante	<input type="text"/>	Municipio	<input type="text"/>
Teléfono	<input type="text"/>	Estado	<input type="text" value="Creada"/>
Dirección	<input type="text"/>		

 **Cámara de Comercio de la República de Cuba** user
Logout

Registrar datos de la empresa

Datos de Registro

Nombre Oficial	<input type="text"/>	Ingreso Anual (MN)	<input type="text"/>
Siglas Oficiales	<input type="text"/>	Ingreso Anual (CUC)	<input type="text"/>
Año de Creación	<input type="text"/>	Mixta	<input type="checkbox"/>
Año de Establecimiento	<input type="text"/>	CUP	<input type="checkbox"/>
No. de Tarjeta	<input type="text"/>	Cuota	<input type="text"/>
Ministerio	<input type="text"/>		
Delegación	<input type="text"/>		
Objeto Social	<input type="text"/>		

CAPÍTULO 2: REQUISITOS Y DISEÑO

Sección: "Modificar"

Acción del Actor	Respuesta del Sistema
1. El usuario elige la opción Buscar empresa . Si desea cancelar la operación, ver flujo alternativo 1 .	2. El sistema muestra una interfaz que permite buscar una empresa, ver caso de uso extendido Buscar empresa .
3. El usuario selecciona la empresa y luego selecciona la opción Modificar .	4. El sistema muestra la interfaz que permite actualizar los datos de la empresa.
5. El usuario modifica los datos de la empresa. Si desea cancelar la operación, ver flujo alternativo 1 .	
6. El usuario selecciona la opción gestionar expediente para modificar los datos de los	7. El sistema muestra una interfaz que permite gestionar el expediente de la empresa. Ver

CAPÍTULO 2: REQUISITOS Y DISEÑO

documentos.	caso de uso incluido Gestionar expediente.
8. El actor selecciona la opción Actualizar. Si desea cancelar la operación, ver flujo alternativo 1.	9. El sistema guarda los datos modificados. Terminando así el caso de uso.

Prototipo de Interfaz

Cámara de Comercio de la República de Cuba user Logout

Aprobar afiliación

Gestionar empresa

Nombre de empresa

Siglas

Provincia

Ministerio

Cuota

Buscar

Listado de empresas

No expediente	Entidad	Provincia	Ministerio	Estado	Cuota	Objeto social
1252-01	Desoft	Habana	Informática	Creada	10000	la empresa....

Adicionar Modificar Cancelar Cerrar

CAPÍTULO 2: REQUISITOS Y DISEÑO

 **Cámara de Comercio de la República de Cuba** user
Logout

Registrar afiliación

Modificar datos de Contacto

Entidad	<input type="text"/>	Móvil	<input type="text"/>
Siglas	<input type="text"/>	FAX	<input type="text"/>
Directivo	<input type="text"/>	Email	<input type="text"/>
Cargo del Directivo	<input type="text"/>	Sitio Web	<input type="text"/>
Representante	<input type="text"/>	Provincia	<input type="text"/>
Cargo del Representante	<input type="text"/>	Municipio	<input type="text"/>
Teléfono	<input type="text"/>	Estado	Creada
Dirección	<input type="text"/>		

 **Cámara de Comercio de la República de Cuba** user
Logout

Modificar datos de Registro

Nombre Oficial	<input type="text"/>	Ingreso Anual (MN)	<input type="text"/>
Siglas Oficiales	<input type="text"/>	Ingreso Anual (CUC)	<input type="text"/>
Año de Creación	<input type="text"/>	Mixta	<input type="checkbox"/>
Año de Establecimiento	<input type="text"/>	CUP	<input type="checkbox"/>
No. de Tarjeta	<input type="text"/>	Cuota	<input type="text"/>
Ministerio	<input type="text"/>		
Delegación	<input type="text"/>		
Objeto Social	<input type="text"/>		

CAPÍTULO 2: REQUISITOS Y DISEÑO

Sección: "Mostrar datos de empresa"

Acción del Actor	Respuesta del Sistema
1. El usuario elige la opción Buscar empresa . Si desea cancelar la operación, ver flujo alternativo 1 .	2. El sistema muestra una interfaz que permite buscar una empresa, ver caso de uso extendido Buscar empresa .
3. El usuario elige la opción Mostrar datos de empresa . Si selecciona la opción cancelar , ver flujo alternativo 1 .	4. El sistema muestra los datos registrados de la empresa seleccionada.
5. El usuario selecciona la opción Cerrar .	6. El sistema guarda los datos.

Prototipo de Interfaz

CAPÍTULO 2: REQUISITOS Y DISEÑO

 **Cámara de Comercio de la República de Cuba** user
Logout

Aprobar afiliación

Gestionar empresa

Nombre de empresa

Siglas

Provincia

Ministerio

Cuota

Buscar

Listado de empresas

No expediente	Entidad	Provincia	Ministerio	Estado	Cuota	Objeto social
1252-01	Desoft	Habana	Informática	Creada	10000	la empresa...

Adicionar **Modificar** **Cancelar**

Cerrar

 **Cámara de Comercio de la República de Cuba** user
Logout

Aprobar afiliación

Datos de Contacto

Entidad Cooperativa Agramonte

Siglas -

Directivo Jorge Hernández Camero

Cargo del Directivo Director General

Representante Dayami Darias Dávila

Cargo del Representante Administrador

Teléfono 076825421

Dirección

Móvil 53538699

FAX -

Email ddavila89@gmail.com

Sitio Web -

Provincia Habana

Municipio Cotorro

Estado Aprobado

Siguiente

CAPÍTULO 2: REQUISITOS Y DISEÑO

 **Cámara de Comercio de la República de Cuba** user
Logout

Aprobar afiliación

Datos de Registro

Nombre Oficial *Cooperativa Agramonte*
Siglas Oficiales *-*
Año de Creación *1987*
No. de Tarjeta *522112*
Ministerio *Ministerio de la Agricultura*
Delegación *Villa Clara*
Año de Establecimiento *2000*
Ingreso Anual (MN) *250000.00*
Ingreso Anual (CUC) *25000.00*
Cuota *1000.00*
CUP *Si*
Mixta *Si*
Deudora *Si*
Estado *Aprobado*

Objeto Social

 **Cámara de Comercio de la República de Cuba** user
Logout

Aprobar afiliación

Datos de Actividad

Calificadores:

Productor *Si*
Comercializador *Si*
Importador *Si*
Exportador *No*
Servicios *No*

Sectores *Agricultura, Pesca y Alimentación*
Secciones *Sección Agroindustria Alimentaria*
Subsecciones *Subsecciones*
Anotaciones *Anotaciones*

CAPÍTULO 2: REQUISITOS Y DISEÑO

Flujos Alternos	
Flujo Alterno 1	
Acción del actor	Respuesta del sistema
1. El usuario elige cancelar la operación.	2. El sistema cancela la operación. Terminando así el caso de uso.
Poscondiciones	Queda registrada la empresa. Quedan modificados en el sistema los datos de una empresa ya registrada con anterioridad. Se mostraron los datos de la empresa.

Tabla 2: Descripción del caso de uso, Gestionar Empresa.

2.5 ARQUITECTURA

La arquitectura de un sistema está compuesta por varios elementos entre los que se encuentran los patrones arquitectónicos y los patrones de diseño, cuyos niveles de abstracción entre estos dos están definidos en el mismo orden en que se mencionaron (Camacho, y otros, 2004). Se abordará a continuación el patrón arquitectónico usado en la implementación del sistema describiendo la arquitectura del mismo, así como los patrones de diseño empleados.

Los patrones arquitectónicos se definen como la descripción de un problema particular y recurrente de diseño, que aparece en un contexto específico y presenta una solución demostrada. Los patrones arquitectónicos expresan el esquema de organización estructural fundamental para sistemas de software. Provee un conjunto de subsistemas predefinidos, especifica sus responsabilidades e incluye reglas y pautas para la organización de las relaciones entre ellos. (Camacho, y otros, 2004)

El patrón arquitectónico usado en la implementación del sistema es Arquitectura en capas, su selección está condicionada por las características del software que se está desarrollando así como las cualidades o atributos de calidad que se desean alcanzar.

Este patrón consiste en estructurar aplicaciones que pueden ser descompuestas en grupos de subtareas, las cuales se clasifican de acuerdo a un nivel particular de abstracción, además posee atributos como la reusabilidad, la portabilidad y la facilidad de prueba. (Camacho, y otros, 2004)

CAPÍTULO 2: REQUISITOS Y DISEÑO

A continuación se describen brevemente las capas que conforman la arquitectura, donde las mismas tienen un orden lógico de precedencia, es decir las funcionalidades y recursos están encapsulados en cada nivel de acuerdo a la responsabilidad que se establece para cada uno de ellos.

- **Servicio:** Esta capa ofrece un conjunto de servicios y operaciones de entrada y salida de datos a los que se acceden a través de interfaces. Las interfaces contenidas en esta capa son las que se encargan de realizar la comunicación entre el Gestor de Procesos Bonita y la capa de Negocio mediante el acceso a los servicios y métodos que esta brinda.
- **Negocio:** Esta capa ofrece un conjunto de interfaces que son usadas por la capa de servicios para invocar las funcionalidades que responden a la lógica del negocio. La misma contiene a los BO (Business Objects, objetos de negocio), los que típicamente son referidos como clases controladoras. Esta capa recibe peticiones del nivel superior, gestiona o procesa las mismas, de ser necesario solicitándolo a la capa de Acceso a Datos y finalmente envía una respuesta continuando el proceso en el punto donde se inició dicha petición.
- **Acceso a datos:** Esta es la capa que controla todo lo concerniente a la información que se encuentra en la fuente de almacenamiento (Capa de datos). Al ser la capa inferior los componentes que contiene desconocen los niveles superiores y únicamente se limitan al manejo de la información, ya sea para persistirla u obtenerla para su procesamiento y propagación por la aplicación. Todo este manejo es responsabilidad de los Objetos de Acceso a Datos (DAOs por sus siglas en inglés). En la misma se hace uso de la especificación de JPA para manejar la persistencia de los objetos.
- **Entidades del negocio:** Las entidades del negocio son objetos persistentes del dominio que representan conceptos reales tales como una empresa o una persona. Las mismas son agrupadas y representadas transversalmente a las capas de negocio y acceso a datos pues ambas las utilizan, garantizando que las mismas sean accesibles sin violar la comunicación entre las capas.
- **Base:** Contiene las clases y servicios comunes a todas las capas entre los que se encuentran los relacionados con la mensajería, el manejo de

CAPÍTULO 2: REQUISITOS Y DISEÑO

excepciones y algunas características de seguridad como el control de las trazas y las transacciones, para ello, se hace uso del framework Spring, aprovechando las ventajas que ofrece el mismo.

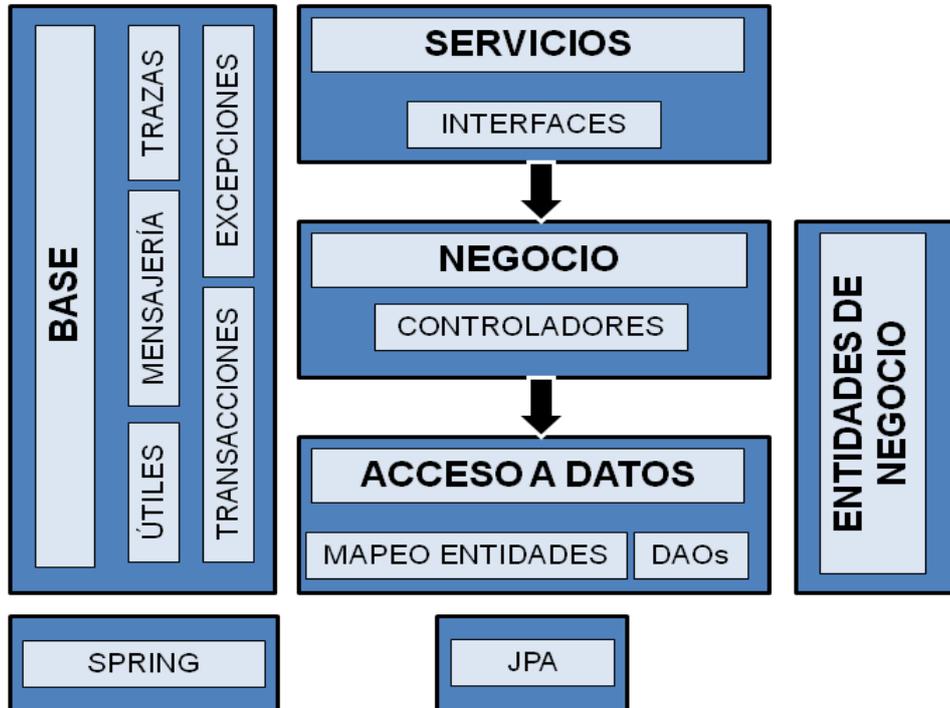


Figura 9: Vista de la Arquitectura.

2.5.1 Patrones de diseño

A un nivel menor de abstracción de los patrones arquitectónicos se encuentran los patrones de diseño. Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de software, o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación, que resuelve un problema general de diseño en un contexto particular. Tienden a ser independientes de los lenguajes y paradigmas de programación y su aplicación no afecta necesariamente al sistema completo pero sí a un subsistema o parte del mismo. (Buschmann, y otros, 1996)

Los patrones de diseño utilizados son los siguientes:

Patrones de Creación: Tratan la creación de instancias o sobre qué objetos un objeto delegará responsabilidades. (E, y otros, 2003)

- **Abstract Factory:** Proporciona una interfaz para crear familias de objetos relacionados o dependientes sin especificar su clase concreta. Este patrón se empleó en la construcción de los objetos en la capa de negocio a través

CAPÍTULO 2: REQUISITOS Y DISEÑO

de los ficheros de configuración que usa el framework Spring, sin conocer los detalles concretos de su instanciación.

- Singleton: Garantiza que solamente se cree una instancia de la clase y provee un punto de acceso global a él. Este patrón se utilizó en la capa de servicio para crear una instancia única de las clases controladoras.

Patrones Estructurales: Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad, describiendo así la forma en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. (E, y otros, 2003)

- Composite: Permite gestionar objetos complejos e individuales de forma uniforme. La utilización de este patrón se evidencia en el trabajo con las entidades de negocio, específicamente con las clases que representan documentos que tienen una asociación recursiva.
- Decorator: Extiende la funcionalidad de un objeto dinamizante de tal modo que es transparente a sus clientes. Se utilizó en las entidades del negocio para obtener las salidas de datos en un formato específico.
- Facade: Simplifica los accesos a un conjunto de objetos relacionados proporcionando un objeto de comunicación. Es utilizado entre las capas que forman la arquitectura para establecer un único punto de acceso a las interfaces definidas.

Patrones de Comportamiento: Tratan la interacción y cooperación entre clases. Organizan, manejan y combinan comportamientos. (E, y otros, 2003)

- Iterator: Acceso secuencial a los elementos de una colección. Su utilización se evidencia específicamente en la capa de negocio para el trabajo con las listas que han sido definidas.
- State: Modifica el comportamiento de un objeto cuando su estado interno cambia. Se utilizó para modificar los estados de las empresas, los expedientes y los documentos.
- Strategy: Define una familia de algoritmos, encapsula cada uno y los hace intercambiables. Fue empleado en la capa de negocio para ordenar los elementos representados en el diseño de las clases como colecciones, sobre los cuales se aplican algoritmos de ordenamiento.

CAPÍTULO 2: REQUISITOS Y DISEÑO

2.6 MODELO DE DISEÑO

Este modelo describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales tienen impacto en el sistema a considerar. Es utilizado como entrada fundamental de las actividades de implementación. Para su desarrollo se identificarán subsistemas y paquetes de diseño, así como diagramas de clases del diseño, de secuencia y de clases persistentes.

2.6.2 Diagrama de paquetes o subsistemas de diseño

Un diagrama de paquetes muestra cómo un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Los Paquetes están organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los mismos (Jacobson, 2000). Para consultar la descripción de cada uno de los paquetes especificados en este diagrama, examinar el documento Modelo de diseño, ver **Anexo 5**.

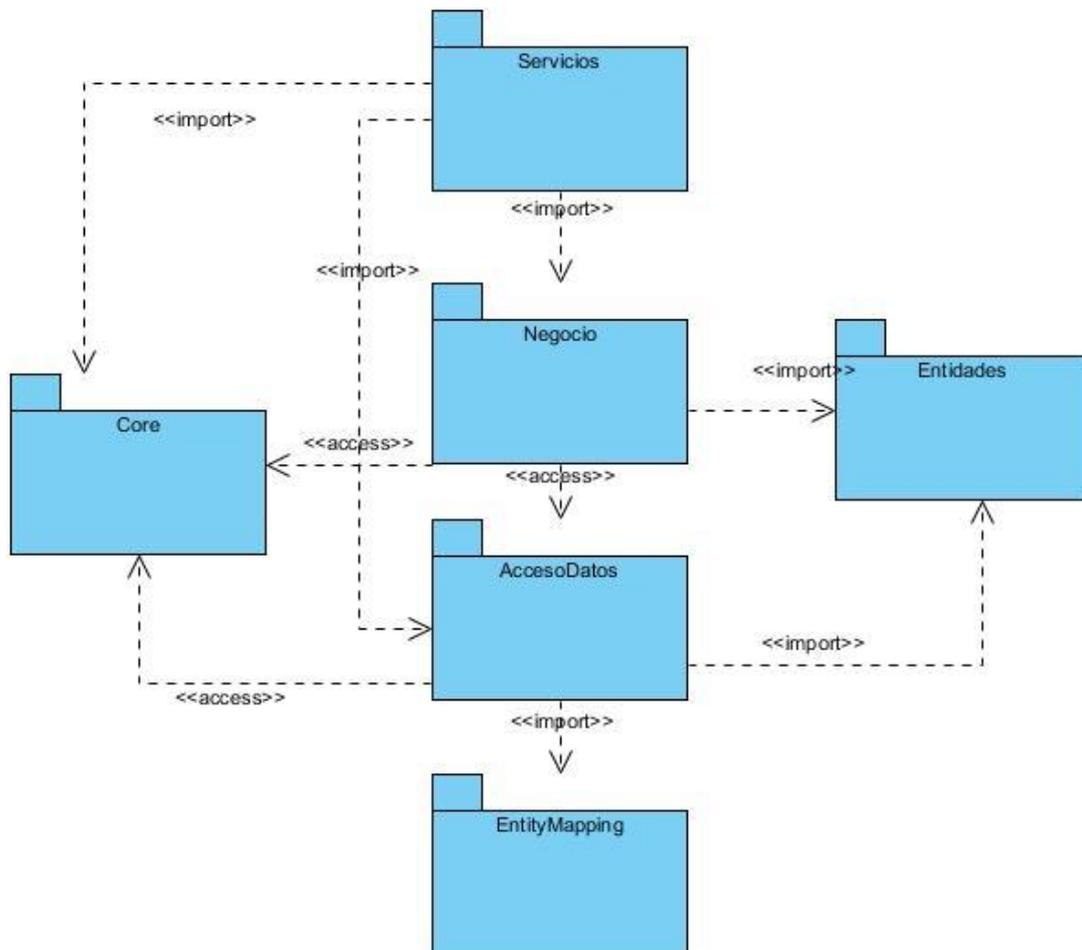


Figura 10: Diagrama de paquetes.

CAPÍTULO 2: REQUISITOS Y DISEÑO

2.6.3 Diagrama de clases del diseño

Los diagramas de clases del diseño describen la realización de los casos de uso y a la vez constituyen una abstracción del modelo de implementación y del código fuente. Son una entrada esencial a las actividades de implementación (Jacobson, 2000). Como el sistema registral del proceso Afiliación de Empresas Cubanas es una aplicación web, en los diagramas de clases de diseño elaborados se hizo uso de los estereotipos web, para ilustrar la colaboración real y la representación de los elementos que interactúan en la ejecución de las funcionalidades que debe brindar el sistema. Para consultar todos los diagramas de clases del diseño examinar el documento Modelo de diseño, ver **Anexo 5**.

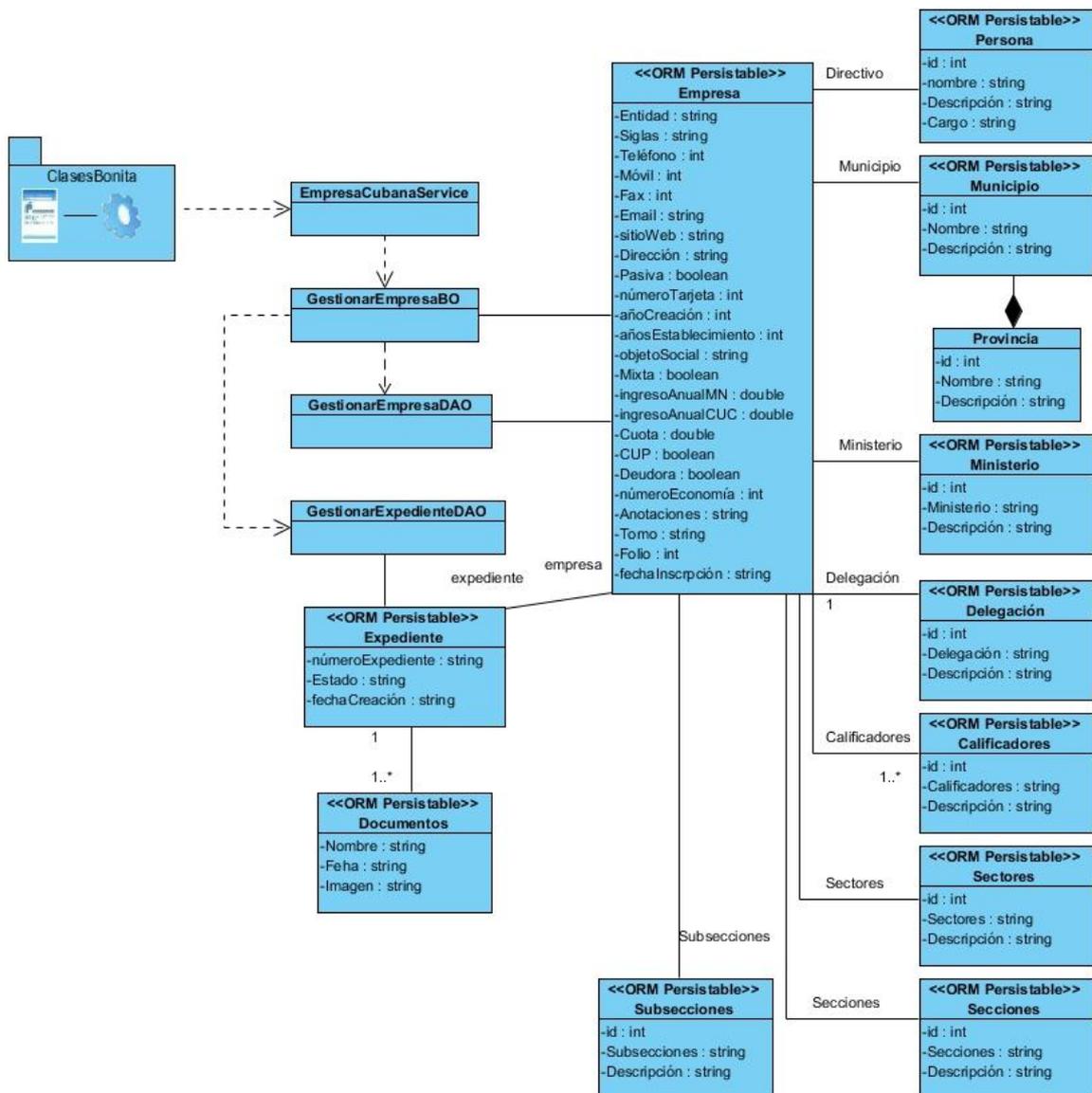


Figura 11: Diagrama de clases del diseño, CU Gestionar empresa.

CAPÍTULO 2: REQUISITOS Y DISEÑO

2.6.4 Diagramas de secuencia

Los diagramas de secuencia muestran una interacción ordenada según la secuencia temporal de eventos y, en particular, los objetos participantes en la interacción y los mensajes que intercambian, ordenados según su secuencia en el tiempo (Jacobson, 2000). A los casos de uso que tienen varias secciones se les realizó un diagrama de secuencia por cada flujo para un mejor entendimiento. Para consultar el diagrama de secuencia del caso de uso “Gestionar empresa”, de la sección “Adicionar”, ver **Anexo 6**. Para más información de la descripción detallada de cada uno de los diagramas de secuencia, consultar el documento Modelo de diseño, ver **Anexo 5**.

2.7 CONCLUSIONES

En este capítulo se llegó a las siguientes conclusiones:

- Se aplicaron técnicas de obtención de información que facilitaron la identificación de los requisitos del sistema.
- Se identificaron los actores y los casos de uso del sistema, y se generó el diagrama de casos de uso del sistema lo cual permitió poder describir el modelo de casos de uso del sistema, a partir de esta descripción se obtuvieron las clases del diseño y los diagramas de secuencia de cada caso de uso.
- Los artefactos obtenidos ayudaron a alcanzar un mayor entendimiento entre los involucrados en el proceso Afiliación de Empresas Cubanas del sistema SIRECC, los cuales servirán de entrada para la posterior implementación del sistema.
- La utilización de patrones de modelado y de casos de uso permitió una mejor modelación de los artefactos correspondientes a estas fases, así también la aplicación de patrones de diseño evitó cometer errores tradicionales en el diseño del proceso Afiliación de Empresas Cubanas del sistema SIRECC.
- Se describió brevemente la arquitectura propuesta y las capas que la componen lo que permitió entender la estructura del sistema para comenzar la implementación del mismo.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

3.1 INTRODUCCIÓN

En el presente capítulo se muestra el diagrama de componentes y además se hace un análisis de los resultados obtenidos durante el desarrollo de la solución propuesta. Para la validación de los artefactos modelo de procesos del negocio, modelo de casos de uso del sistema y descripción de requisitos, se muestran las listas de chequeo que les fueron aplicadas, además del acta de aceptación emitida por el cliente para los artefactos que les fueron presentados. También se aplican métricas para medir la calidad de la funcionalidad del diagrama de casos de uso del sistema y la especificación de los requisitos obtenidos.

3.2 DIAGRAMA DE COMPONENTES

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes. Uno de los usos principales es que puede servir para ver qué componentes pueden compartirse entre sistemas o entre diferentes partes de un sistema (Jacobson, 2000).

Para consultar el diagrama de componentes, ver **Anexo 7** y para consultar la descripción de cada uno de los componentes examinar el documento Modelo de diseño, ver **Anexo 5**.

3.3 VERIFICACIÓN

A continuación se explican detalladamente las métricas empleadas y los resultados logrados en la verificación de los diagramas de casos de uso y los requisitos obtenidos. Antes se debe conocer que la medición de los resultados es uno de los objetivos primordiales en la ingeniería de software y para lograrlo los ingenieros deben emplear métodos efectivos que les permitan evaluar la calidad de forma objetiva.

Las métricas del software proveen información para la toma de decisiones, ayudan a definir con mayor exactitud el desarrollo y calidad del producto, evalúan la productividad de las personas y los beneficios por el uso de nuevos métodos y herramientas. (Estevez, 2002)

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Al implantarse el proceso de medición, los resultados deben ser mostrados para mejorar las áreas en las cuales se identificaron los problemas. Las métricas permiten mejorar el proceso de software, planificando y evaluando la calidad del producto. Estas suministran una visión estratégica de la efectividad del proceso de software.

Además se aplican otras técnicas de verificación como la revisión por pares y las listas de chequeo ambas aplicadas a los artefactos.

3.4.1 Métricas para la calidad de la especificación de los requisitos de software

La métrica para la calidad de la especificación de los requisitos de software se basa en la consistencia de las interpretaciones del grupo de revisión para cada uno de los requisitos especificados. (Pressman, 2005)

Esta métrica consiste en realizar una división entre el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas (n_{ui}) y la cantidad de requisitos de software (n_r), siendo (n_r) la suma de los requisitos funcionales (n_f) más los requisitos no funcionales (n_{nf}). Cuanto más cerca de uno esté el valor de Q_1 menor será la ambigüedad de la especificación. (Pressman, 2005)

$$Q_1 = \frac{n_{ui}}{n_r}; n_r = n_f + n_{nf}$$

Para aplicar dicha métrica se seleccionó un grupo de revisores conformado por especialistas del proyecto y de la Cámara de Comercio los cuales se encargaron de verificar la especificidad de dichos requisitos.

Nombre y Apellidos	Rol desempeñado
Jorge Yuniel Jorrín Perdomo	Jefe de Proyecto
Ana Cecilia Labrador Valdés	Analista Principal
Daniel Varona Cordero	Administrador de Calidad
Yoenia María Martínez Díaz	Analista
Milena Pérez Cepero	Directora de Formación y Membrecía

Tabla 3: Descripción del grupo de revisores

A continuación se muestra un resumen de los resultados obtenidos según las interpretaciones dadas por los revisores:

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Atributo de Calidad	Tipo de Requisito	Interpretaciones	
		Iguales	Desiguales
Especificidad	Funcionales	33	0
	No funcionales	24	1
	Total	57	1

Tabla 4: Resultado de las interpretaciones de requisitos

$$Q_1 = \frac{n_{ui}}{n_r}; n_r = n_f + n_{nf} \quad n_r = 33 + 25 = 58 \quad Q_1 = \frac{57}{58} = 0.98$$

Luego de la aplicación de la métrica Q_1 resultó ser 0.98, lo cual es un valor cercano a 1 quedando demostrado que existe un bajo nivel de ambigüedad en la especificación de los requisitos de software.

3.4.2 Métricas para la calidad de la funcionalidad del diagrama de casos de uso de sistema

Las métricas para la calidad de la funcionalidad del diagrama de casos de uso de sistema definen cuatro atributos: (Bustamante, 2009)

- **Compleitud:** Permite determinar el grado en que se han incluido de forma clara y concisa todos los elementos necesarios para la descripción del aspecto.
- **Consistencia:** Permite definir el grado en que los elementos del artefacto representan en forma única y no contradictoria un aspecto del problema.
- **Correctitud:** Permite establecer el grado de adecuación del artefacto para satisfacer los requisitos establecidos.
- **Complejidad:** Permite medir el grado de claridad y reusabilidad del artefacto.

La tabla de las métricas correspondientes a cada uno de los atributos especificados anteriormente se muestra en el **Anexo 8**.

Al aplicar dicha métrica se obtuvo una tabla para la representación de los porcentajes de error de acuerdo a las métricas aplicadas para cada atributo y el resultado obtenido:

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Atributo	Métrica	Cantidad de casos de uso analizados	Resultado	% de error
Complejidad	1	22	Todos los casos de uso poseen una descripción resumida.	0%
	2	22	Se detectaron 4 casos de uso que tienen requisitos omitidos.	18.18 %
	3	22	Se detectaron 20 casos de uso que no presentaban una descripción extendida.	90.90 %
	4	22	Se detectaron 19 casos de uso que no están redactados en función del responsable.	86.36 %
	5	22	No se detectó ningún caso de uso que no describiera condiciones de excepción relevantes.	0%
	6	22	Todos los casos de uso han sido clasificados según los diferentes tipos de prioridad.	0%
Consistencia	7	22	Ningún caso de uso presentó problemas en cuanto al nombre dado, ya que todos los nombres proporcionados a cada uno de los casos de uso representan una expresión verbal en infinitivo describiendo alguna funcionalidad relevante y significativa para el usuario.	0%
	8	22	Todos los casos de uso representan correctamente la interacción con un actor.	0%
	9	22	Ningún caso de uso presentó	0%

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

			acciones del flujo de eventos que estuvieran asignados a un responsable que no le corresponde.	
	10	22	Todos los casos de uso fueron aceptados.	0%
	11	22	Todos los casos de uso poseen sus descripciones del flujo básico y de flujos alternos separadas.	0%
	12	22	Todos los casos de uso poseen un usuario responsable.	0%
Correctitud	13	22	Todos los casos de uso en que los requisitos fueron representados son comprensibles por el usuario.	0%
	14	22	Ningún caso de uso tuvo que ser modificado para adecuarlo a la funcionalidad del sistema.	0%
	15	22	Ningún caso de uso tuvo que ser modificado para mejorar el proceso actual.	0%
Complejidad	16	22	Ninguno de los elementos que conforman el diagrama de casos de uso requiere reubicación.	0%

Tabla 5: Resultado de las métricas correspondientes a los atributos completitud, consistencia, correctitud y complejidad.

La Descripción de casos de uso es uno de los artefactos más importantes del flujo de trabajo de requisitos, por ello se realizaron dos revisiones en aras de asegurar que el sistema tenga la calidad requerida. En la primera revisión se comprobó que el diagrama construido no cumplía con las métricas 2, 3 y 4 referentes al atributo completitud, representando un 12.21% de error para un 87.79% de calidad. Las métricas referentes a este atributo se aplicaron por segunda ocasión, una vez realizadas las modificaciones en el diagrama de casos de uso del sistema, y se obtuvo un 100% de calidad en todos los indicadores.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

A continuación se muestran gráficamente los resultados obtenidos de la evaluación de los atributos de calidad en sus dos iteraciones:

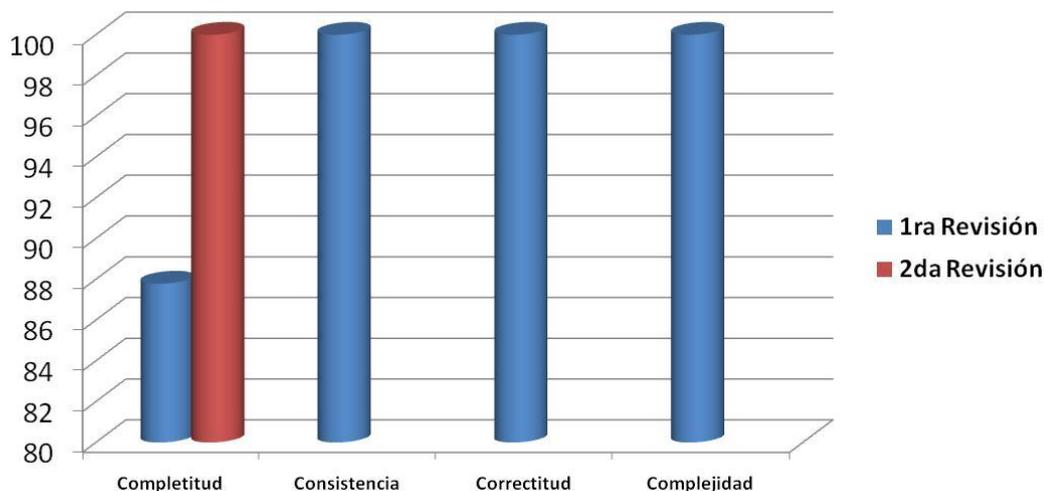


Figura 12: Gráfica de resultados de la métrica para la calidad de la funcionalidad de los DCU.

3.4.3 Revisión por pares

La revisión por pares es una técnica utilizada para validar los documentos que se generan en cada uno de los procesos que se llevan a cabo para el desarrollo del sistema registral para la Cámara de Comercio, con el fin de medir su calidad y factibilidad. Este método deja abierto el trabajo a la comprobación y examen de estos documentos, por un número de integrantes del equipo de desarrollo de igual rango al autor.

Siguiendo esta propuesta los artefactos elaborados durante el desarrollo del proceso Afiliación de Empresas Cubanas fueron intercambiados entre los analistas principales, para que fueran revisados con el fin de medir su calidad, para en el menor tiempo posible corregir los errores encontrados. Esta revisión arrojó varias no conformidades que fueron resueltas en tres iteraciones realizadas. A continuación se muestra una breve descripción de estos resultados:

Documentos	Cantidad de No conformidades	Resultados por iteraciones		
		Iteración 1	Iteración 2	Iteración 3
Descripción de los procesos de negocio	5	4 resueltas	1 resuelta	-
Especificación de requisitos de software	4	4 resueltas	-	-

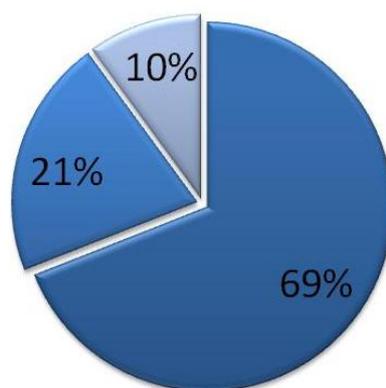
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Descripción de casos de uso del sistema	14	7 resueltas	5 resueltas	2 resueltas
Prototipos de interfaz de usuario	6	5 resueltas	-	1 resuelta
TOTAL	29	20	6	3
% de No conformidades resueltas	100%	69%	21%	10%

Tabla 6: Resultados de la revisión por pares.

Para mayor entendimiento y visión se muestran los resultados de la revisión por pares de forma gráfica:

■ 1ra Revisión ■ 2da Revisión ■ 3ra Revisión



% de No conformidades resueltas

Figura 13: Resultados de la aplicación de la técnica revisión por pares.

3.4.4 Listas de chequeo

La identificación de los requisitos, la especificación de los casos de uso, y la descripción de los modelos de procesos de negocio, constituyen los principales artefactos obtenidos durante el desarrollo del proceso Afiliación de Empresas Cubanas del sistema SIRECC. Con el objetivo de garantizar su factibilidad, fueron sometidos a varias revisiones por parte del equipo de calidad a nivel de proyecto. Quienes para verificar los mismos aplicaron listas de chequeo. Durante la aplicación de las listas de chequeo a los artefactos obtenidos surgieron varias no conformidades por parte del equipo de calidad del proyecto. Todas estas no

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

conformidades encontradas fueron solucionadas por el equipo de análisis del proyecto. A continuación se muestran los resultados obtenidos:

Artefacto	Cantidad de no conformidades	No conformidades resueltas
Descripción de los procesos de negocio	10	10
Especificación de requisitos de software	5	5
Descripción de casos de uso del sistema	6	6
TOTAL	21	21
% de No conformidades resueltas	100%	100%

Tabla 7: Resultado de la aplicación de las listas de chequeo.

Para consultar las listas de chequeo correspondientes a cada artefacto ver los siguientes anexos:

- Lista de chequeo del modelo de procesos de negocio, ver **Anexo 9**.
- Lista de chequeo de requisitos, ver **Anexo 10**.
- Lista de chequeo de casos de uso, ver **Anexo 11**.

3.4 VALIDACIÓN

A continuación se muestran los resultados obtenidos durante la validación del modelado de los procesos, de los requisitos y del sistema. Además de la carta de aceptación de los artefactos emitida por el cliente.

3.4.1 Validación del modelado de proceso

Existen distintos métodos para la validación de los modelos de procesos teniendo en cuenta las características del producto a desarrollar se decidió utilizar para la validación de estos modelos, talleres con el cliente. Estos talleres se llevaron a cabo con el objetivo de lograr la aceptación y confianza de los clientes que harán uso del sistema en el modelo.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Se llevaron a cabo 3 talleres, los cuales arrojaron resultados positivos pues se logró el objetivo perseguido, obtener la aceptación del cliente con respecto a los procesos modelados.

3.4.2 Validación de requisitos

Permite demostrar que los requisitos definidos son los que realmente quiere el cliente y que cumplen con las necesidades del mismo, “examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencia, sin omisiones, que los errores detectados han sido corregidos y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto”. (Pressman, 2005)

Validación de requisitos mediante prototipos no funcionales: Un prototipo es una versión inicial de un sistema de software, es la presentación de la interfaz del producto de manera que permita que el usuario verifique que el sistema va a satisfacer sus necesidades. Esta técnica provee a los analistas de una importante retroalimentación para conocer si el sistema diseñado sobre la base de los requisitos capturados, les permite a los usuarios realizar su trabajo de forma efectiva y eficiente.

Una de las técnicas seleccionadas para realizar la validación de los requisitos del sistema fue la de prototipos. Se crearon uno o varios prototipos de interfaz de usuario por cada requisito, son estos un conjunto de interfaces que muestran una simulación de lo que sería el futuro sistema a implementar. Después de confeccionados, se realizaron varias reuniones con los clientes que tuvieron como objetivo y resultado la aceptación por parte de los mismos de la propuesta realizada.

Los prototipos asociados a cada requisito pueden encontrarse en el documento de Especificación de requisitos de software, ver **Anexo 2**.

3.4.3 Pruebas de caja negra

Las pruebas de caja negra, también denominadas pruebas de comportamiento, se centran en los requisitos funcionales del software. O sea, la prueba de caja negra permite al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Intentan encontrar errores de las siguientes categorías: funciones incorrectas o ausentes, errores de interfaz, errores en estructuras de datos o en accesos a bases de datos externas, errores de rendimiento y errores de inicialización y de terminación. (Pressman, 2005)

Para desarrollar las pruebas de caja negra existen varias técnicas, entre ellas están:

- Los métodos de pruebas basados en grafos: Exploran las relaciones entre los objetos del programa y su comportamiento.
- La partición equivalente: Divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- El análisis de valores límites: Prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- La prueba de la tabla ortogonal: Suministra un método sistemático y eficiente para probar sistemas con un número reducido de parámetros de entrada.

Se utilizará el método de caja negra, aplicando la técnica de particiones equivalentes y análisis de valores límites que está definido en la estrategia de pruebas trazada por el proyecto.

Para llevar a cabo la aplicación de estas pruebas se confeccionaron varios casos de pruebas basados en requisitos con diferentes escenarios. Estos pudieron ser probados a través de funcionalidades o propiedades presentes en el gestor de procesos Bonita, tales como la definición de tipos de datos para cada uno de los parámetros de entrada, definición de campos nulos u obligatorios, así como el límite de caracteres de los parámetros.

A continuación se muestra solo el caso de prueba para el requisito Registrar empresa, para más información de la descripción detallada de cada uno de ellos, consultar el documento Casos de pruebas basados en requisitos, ver **Anexo 12**.

Escenario	Resultado esperado	Resultado de la prueba
EC1. Introduce los campos de forma correcta. Ejemplo: Entidad: CIMEX	El sistema realiza la validación de los datos correctamente y lleva a cabo el registro de la	Satisfactorio

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Directivo: Santiago Cargo: Administrador	empresa.	
EC2. Introduce los campos de forma incorrecta. Ejemplo: Entidad: CIMEX156 Directivo: Santiago145 Cargo: Administrador155	El sistema muestra un mensaje de error alertando que los campos son incorrectos.	Satisfactorio
EC3. Deja campos vacíos. Ejemplo: Entidad: Directivo: Cargo:	El sistema muestra un mensaje de error alertando que los campos son incorrectos.	Satisfactorio
EC4. Introduce los campos con caracteres no válidos. Ejemplo: Entidad: CIMEX\$\$\$ Directivo: Santiago///-~ Cargo: Administrador%%	El sistema muestra un mensaje de error alertando que los campos son incorrectos.	Satisfactorio

Tabla 8: Caso de prueba: Requisito Registrar empresa.

3.4.4 Carta de aceptación del cliente

Durante todo el ciclo de vida del producto y durante las diferentes fases se fueron generando varios artefactos los cuales fueron presentados a los funcionarios de la Cámara de Comercio, y a través de entrevistas y talleres se pudo lograr que el cliente aceptara las propuestas que se le llevaban y sugirieran cambios en algunos de ellos. Después de varias iteraciones de revisiones los artefactos Modelo de procesos de negocio y Especificación de requisitos de software están en su fase final y han sido firmados por el cliente, por lo que se generó el acta de aceptación para estos dos artefactos emitidas por dicha entidad. Para consultar el acta de aceptación del cliente, ver **Anexo 13**.

3.5 CONCLUSIONES

En este capítulo se arribó a las siguientes conclusiones:

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

- Al culminar el capítulo “Implementación y Prueba”, se le ha dado cumplimiento a las últimas tareas trazadas para el desarrollo de la aplicación, quedando totalmente implementada la herramienta web.
- La aplicación de la métrica para la calidad de la funcionalidad del diagrama de casos de uso de sistema demostró que se construyó un diagrama con un 100% de completitud, consistencia, correctitud y complejidad.
- La utilización de la métrica para la calidad de la especificación de los requisitos permitió conocer el nivel de ambigüedad que presentaban los requisitos especificados, arrojando que existe un bajo nivel de ambigüedad en la especificación de los mismos.
- Las técnicas aplicadas para la validación de los artefactos permitieron que cada uno de ellos presentaran descripciones entendibles y correctas.
- La creación de los casos de pruebas para aplicar las pruebas de caja negra arrojaron resultados satisfactorios, puesto que el sistema dio todas las respuestas que se esperaban ante las diferentes circunstancias.
- El acta de aceptación emitida por el cliente demostró que tanto la descripción de los procesos de negocio como la especificación de requisitos quedaron bien definidos, especificados y sin ambigüedad, abarcando las necesidades de los funcionarios de la cámara.

CONCLUSIONES GENERALES

Una vez concluido el desarrollo del presente trabajo de diploma se puede llegar a las siguientes conclusiones:

- El estudio de las herramientas y tecnologías a partir de los criterios de selección facilitaron la obtención de las mismas para aplicarlas en la solución del sistema.
- El análisis y modelado de los procesos de negocio, junto a la interacción directa con los clientes del sistema, permitió que se obtuvieran resultados satisfactorios en la identificación de los requisitos que debe cumplir el sistema; identificándose un total de 33 requisitos funcionales, de los cuales 12 tienen complejidad alta, 8 complejidad media y 13 complejidad baja, definiéndose todos de prioridad alta para el cliente.
- La aplicación de métodos y métricas permitió identificar las deficiencias en la elaboración de los artefactos, posibilitando la corrección de las mismas para obtener resultados positivos con alto grado de calidad. Donde la especificación de requisitos de software contiene un 0.98 de carencia de ambigüedad y la especificación de casos de uso del sistema presenta los atributos de completitud, consistencia, correctitud y complejidad a un 100% de calidad.
- Se diseñó una aplicación web que implementa el proceso de Afiliación de Empresas Cubanas. Dándole cumplimiento con lo anterior al objetivo general de la investigación, facilitando una estandarización en la forma de llevar a cabo el proceso, donde las funcionalidades que responden a este se encuentran centralizadas en una sola aplicación y accesible para todos los usuarios de la misma.

RECOMENDACIONES

Luego de haber dado cumplimiento a los objetivos de este trabajo de diploma y teniendo en cuenta las experiencias adquiridas durante el desarrollo del mismo, se recomienda:

- Que el trabajo sea analizado como material de estudio para aquellas personas que deseen desarrollar un sistema con similares características.
- Continuar con el mejoramiento del sistema web a partir de los nuevos cambios que puedan surgir en fases superiores, de acuerdo a nuevas necesidades de los clientes.
- Proponer las herramientas y tecnologías utilizadas a otros proyectos del centro que tengan la necesidad de trabajar con un enfoque orientado a procesos.

BIBLIOGRAFÍA

- ABCdatos. 2011.** abcdatos. [En línea] 2011. [Citado el: 18 de Febrero de 2012.] <http://www.abcdatos.com/programas/programa/v586.html>.
- Apache Foundation. 2011.** Apache Tomcat Wiki. [En línea] 2011. [Citado el: 20 de Febrero de 2012.] <http://wiki.apache.org/tomcat/>.
- Bizagi. 2009.** Patrones de modelado. [En línea] 2009. [Citado el: 15 de Febrero de 2012.] http://wiki.bizagi.com/es/index.php?title=Main_Page.
- Buschmann, Frank, y otros. 1996.** *Pattern – Oriented Software Architecture. A System of Patterns*. Inglaterra : John Wiley & Sons, Inc., 1996. 978-0471958697.
- Bustamante, Jomar. 2009.** Métricas de Software. 2009.
- Camacho, Erika, Cardeso, Fabio y Núñez, Gabriel. 2004.** *Arquitecturas de Software*. 2004.
- Cámara de Comercio - Lima. 2007.** Cámara de Comercio - Lima. [En línea] 2007. [Citado el: 12 de Diciembre de 2011.] <http://www.e-camara.net>.
- Cámara de Comercio de Cuba. 2009.** Cámara de Comercio de Cuba. [En línea] 2009. [Citado el: 20 de 10 de 2011.] <http://www.camaracuba.cu/>.
- Cámara de Comercio de Florencia para el Caquetá. 2011.** CAE- Centro de Atención Empresarial. [En línea] 2011. [Citado el: 15 de Diciembre de 2011.] <http://www.ccflorencia.org.co/servicio-al-cliente/cae-centro-de-atencion-empresarial.html>.
- Cámara de Comercio de Medellín para Antioquía. 2011.** Cámara de Comercio de Medellín. [En línea] 2011. [Citado el: 11 de Diciembre de 2011.] <http://www.camaramedellin.com.co/site/Tramites-Virtuales/Inscripciones-virtuales.aspx>.
- CC de Bogotá. 2011.** Cámara de Comercio de Bogotá. [En línea] 2011. [Citado el: 15 de Diciembre de 2011.] <http://serviciosonline.ccb.org.co/proponentes/defaultproponente.aspx>.
- Colectivo de Autores. 2011.** Buscar Portal. *Buscar Portal*. [En línea] 2011. [Citado el: 2 de Febrero de 2012.] http://www.buscarportal.com/articulos/iso_9001_enfoque_procesos.html.

BIBLIOGRAFÍA

- . 1999. Departamento de Ciencias de la Computación Universidad de Chile. [En línea] 8 de Diciembre de 1999. [Citado el: 16 de febrero de 2012.] <http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>.
- Compujuy. 2010.** Compujuy. [En línea] 2010. [Citado el: 18 de Febrero de 2012.] <http://www.compujuy.com.ar/index.php>.
- Corporación Oracle. 2012.** NetBeans. [En línea] Oracle Corporation, 2012. [Citado el: 4 de Diciembre de 2011.] http://netbeans.org/community/releases/71/index_pt_BR.html.
- Definición.de. 2011.** Definición de. [En línea] 2011. [Citado el: 14 de 11 de 2011.] <http://definicion.de/comercio/>.
- Delgado, Andrea. 2007.** Grupo Alarcos. *Grupo Alarcos*. [En línea] 2007. <http://alarcos.inf-cr.uclm.es/pnis/articulos/pnis-07-delgado-dsen.pdf>.
- Departamento de Sistemas Informáticos y Computación. 2002.** *Introducción a Herramientas CASE y System Architect*. Valencia : s.n., 2002.
- Diccionario Manual de la Lengua Española Vox. 2012.** The Free Dictionary. [En línea] Farlex, Inc., 2012. [Citado el: 15 de Febrero de 2012.] <http://es.thefreedictionary.com/patr%C3%B3n>.
- Director de Formación y Membrecía. 2009.** *Procedimiento de Afiliación de Empresas Cubanas*. Habana : s.n., 2009.
- E, Gamma, y otros. 2003.** *Design Patterns. Elements of Reusable Object-Oriented*. s.l. : Addison Wesley, 2003.
- . 2003. *Patrones de Diseño. Elements of Reusable Object-Oriented*. s.l. : Addison Wesley, 2003.
- Equipo de desarrollo BOS. 2011.** BonitaSoft. [En línea] BonitaSoft, 2011. [Citado el: 14 de 02 de 2012.] <http://es.bonitasoft.com/productos/dibuje-sus-procesos-connectese-sus-sistemas-ejecute-sus-procesos-con-un-solo-clic>.
- Equipo de Desarrollo de VP. 2012.** Visual Paradigm. [En línea] 2012. [Citado el: 15 de febrero de 2012.] <http://www.visual-paradigm.com/>.
- Estevez, Isabel Pérez. 2002.** *Métricas para el control de proyectos de software*. Ciudad de la Habana : s.n., 2002.
- Experiencia en transformación de modelos de procesos.* **Beatriz Mora, Francisco Ruiz, Félix García, Mario Piattini;** 2007. Universidad de Castilla-La Mancha, Escuela Superior de Informática, España : s.n., 2007.

BIBLIOGRAFÍA

- Grupo Kaizen Calidad. 2005.** GestioPolis. *GestioPolis*. [En línea] Septiembre de 2005. [Citado el: 21 de Enero de 2012.] <http://www.gestiopolis.com/canales5/ger/gksa/61.htm>.
- Gunnar Övergaard, Karin Palmkvist. 2004.** *Use Cases Patterns and Blueprints*. s.l. : Addison Wesley, 2004.
- ISO 9126. 2005.** “Software product evaluation – Quality characteristics and guidelines for their use”. 2005.
- ISO/TC 176. 2009.** Iram. *Iram*. [En línea] 2009. [Citado el: 16 de Enero de 2012.] http://www.iram.com.ar/Documentos/Certificacion/Sistemas/ISO9000_2000/processos.pdf. ISO/TC 176/SC 2/N 544R.
- Jacobson, Ivar, Booch, Grady and Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000.
- JAVA J2EE. 2009.** JAVA J2EE. [En línea] 2009. [Citado el: 18 de Febrero de 2012.] <http://luchorondon.blogspot.com/2009/04/jpa-java-persistence-api.html>.
- Java. 2012.** Java. [En línea] 2012. [Citado el: 15 de febrero de 2012.] http://www.java.com/es/download/faq/whatis_java.xml.
- MasterMagazine. 2004.** MasterMagazine. [En línea] 2004. [Citado el: 12 de Enero de 2012.] <http://www.mastermagazine.info/termino/4720.php>.
- OpenXava. 2011.** OpenXava. [En línea] 2011. [Citado el: 18 de Febrero de 2012.] <http://www.openxava.org/web/guest/jpa>.
- Oracle. 2011.** Oracle. [En línea] 2011. [Citado el: 22 de Febrero de 2012.] <http://www.oracle.com/technetwork/java/javasee/overview/index.html>.
- Pressman, Rogen S. 2005.** *Ingeniería de Software. “Un enfoque práctico”*. s.l. : Addison Wesley, 2005.
- Rational Software Corporation. 2003.** Rational Software Corporation. *Rational Software Corporation*. [En línea] 2003. [Citado el: 12 de Diciembre de 2012.] <http://www.rational.com>.
- SOA Agenda. 2011.** SOA Agenda. [En línea] 25 de Septiembre de 2011. [Citado el: 20 de Febrero de 2012.] <http://www.soaagenda.com/journal/articulos/que-son-los-frameworks/>.
- Spring Source. 2011.** Get Started With Spring. [En línea] 2011. [Citado el: 21 de Febrero de 2012.] <http://www.springsource.org/get-started>.

BIBLIOGRAFÍA

—. **2011**. Introduction to Spring Framework. [En línea] 2011. [Citado el: 21 de Febrero de 2012.] <http://static.springsource.org/spring/docs/3.0.x/spring-framework-reference/html/overview.html>.

UTN. 2012. Universidad Tecnológica Nacional. [En línea] 2012. [Citado el: 15 de febrero de 2012.] <http://www.frt.utn.edu.ar/sistemas/paradigmas/lenguajes.htm>.

Veronica de la Morena. 2008. Connexions. [En línea] 27 de Noviembre de 2008. [Citado el: 13 de Enero de 2012.] <http://cnx.org/content/m17457/latest/>.

ANEXOS

ANEXO 1. DOCUMENTO MODELO DE PROCESOS DE NEGOCIO

Este documento contiene la definición del Modelo de procesos de negocio del proceso Afiliación de Empresas Cubanas.

(Consultar documento con el nombre "[0115_Modelo de Procesos de Negocio Afiliación de empresas cubanas.doc](#)" en la carpeta Artefactos).

ANEXO 2. DOCUMENTO DE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

Este documento contiene las condiciones o capacidades y las propiedades o cualidades que el sistema deberá tener.

(Consultar documento con el nombre "[0113_Especificación de Requisitos de Software.doc](#)" en la carpeta Artefactos).

ANEXO 3. DIAGRAMAS DE CASOS DE USO POR PAQUETES



Ilustración 1: Diagrama de casos de uso del paquete Reportes.

ANEXOS

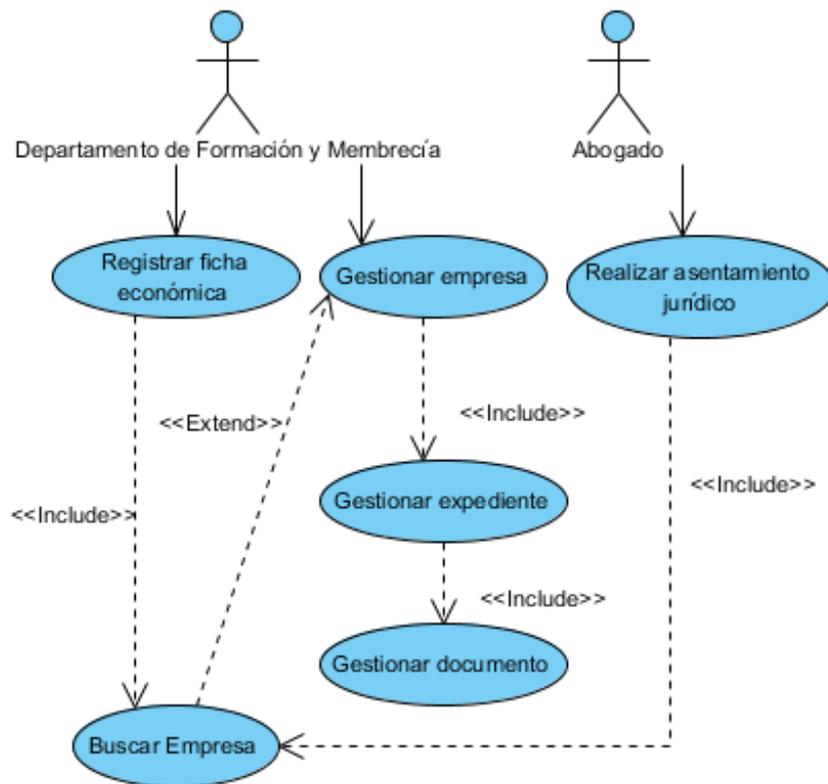


Ilustración 2: Diagrama de casos de uso del paquete Registrar datos de la empresa.

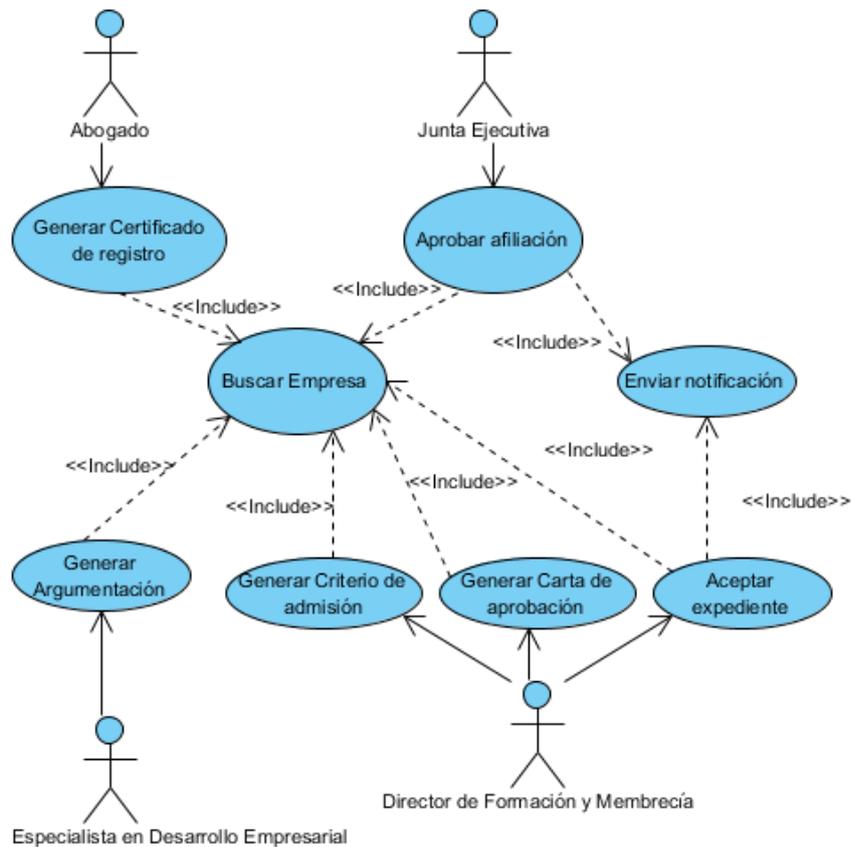


Ilustración 3: Diagrama de casos de uso del paquete Aprobaciones.

ANEXOS

ANEXO 4. DOCUMENTO ESPECIFICACIÓN DE CASOS DE USO

Este documento contiene todos los actores del sistema y su relación con los casos de uso del sistema, además de la descripción de cada caso de uso.

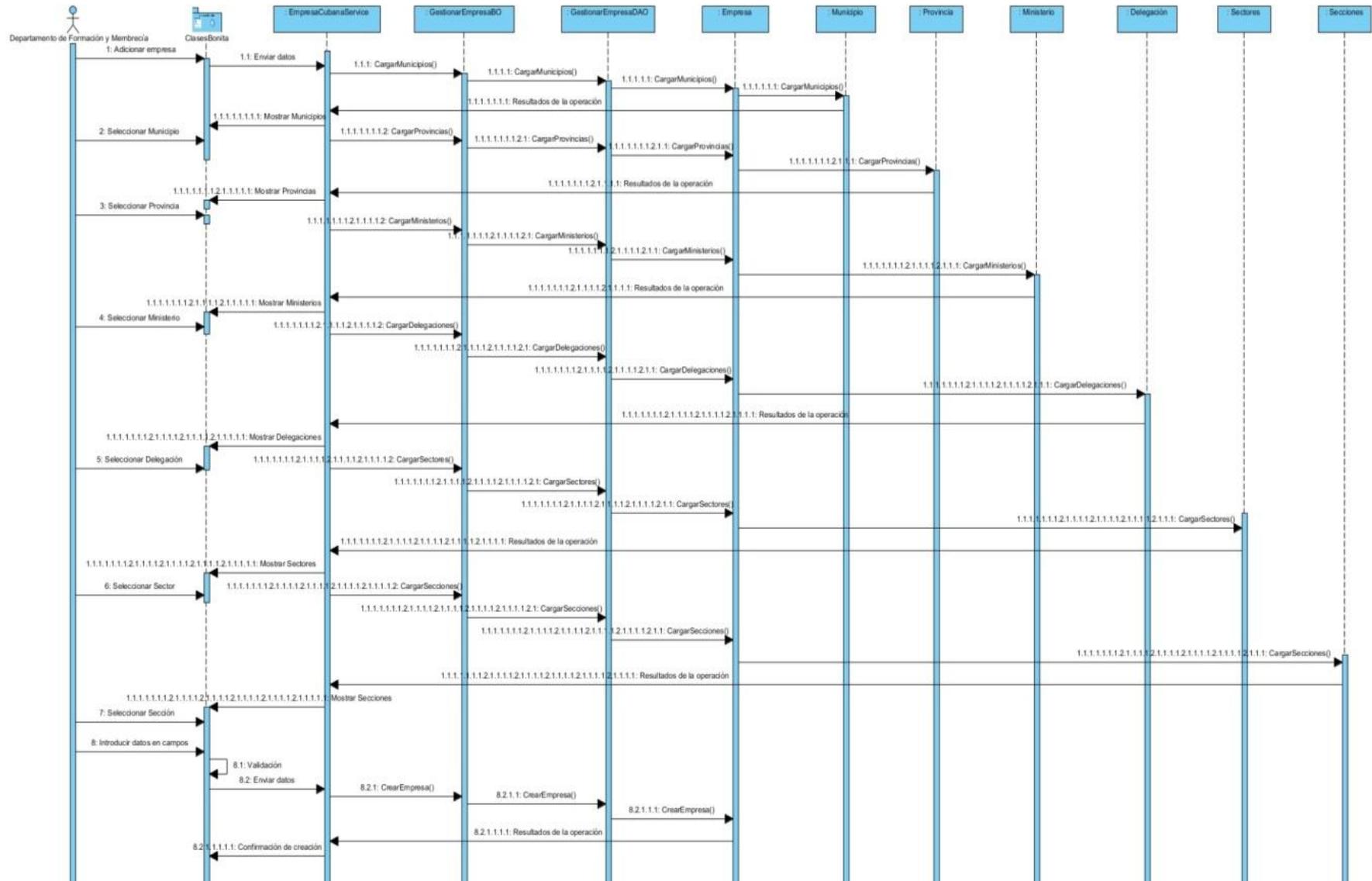
(Consultar documento con el nombre "[0114 Especificación de Casos de Uso.doc](#)" en la carpeta Artefactos).

ANEXO 5. DOCUMENTO MODELO DE DISEÑO

Este documento contiene la realización física de los casos de usos centrándose en cómo los requisitos funcionales y no funcionales tienen impacto en el sistema a considerar. (Consultar documento con el nombre "[Modelo de diseño.doc](#)" en la carpeta Artefactos).

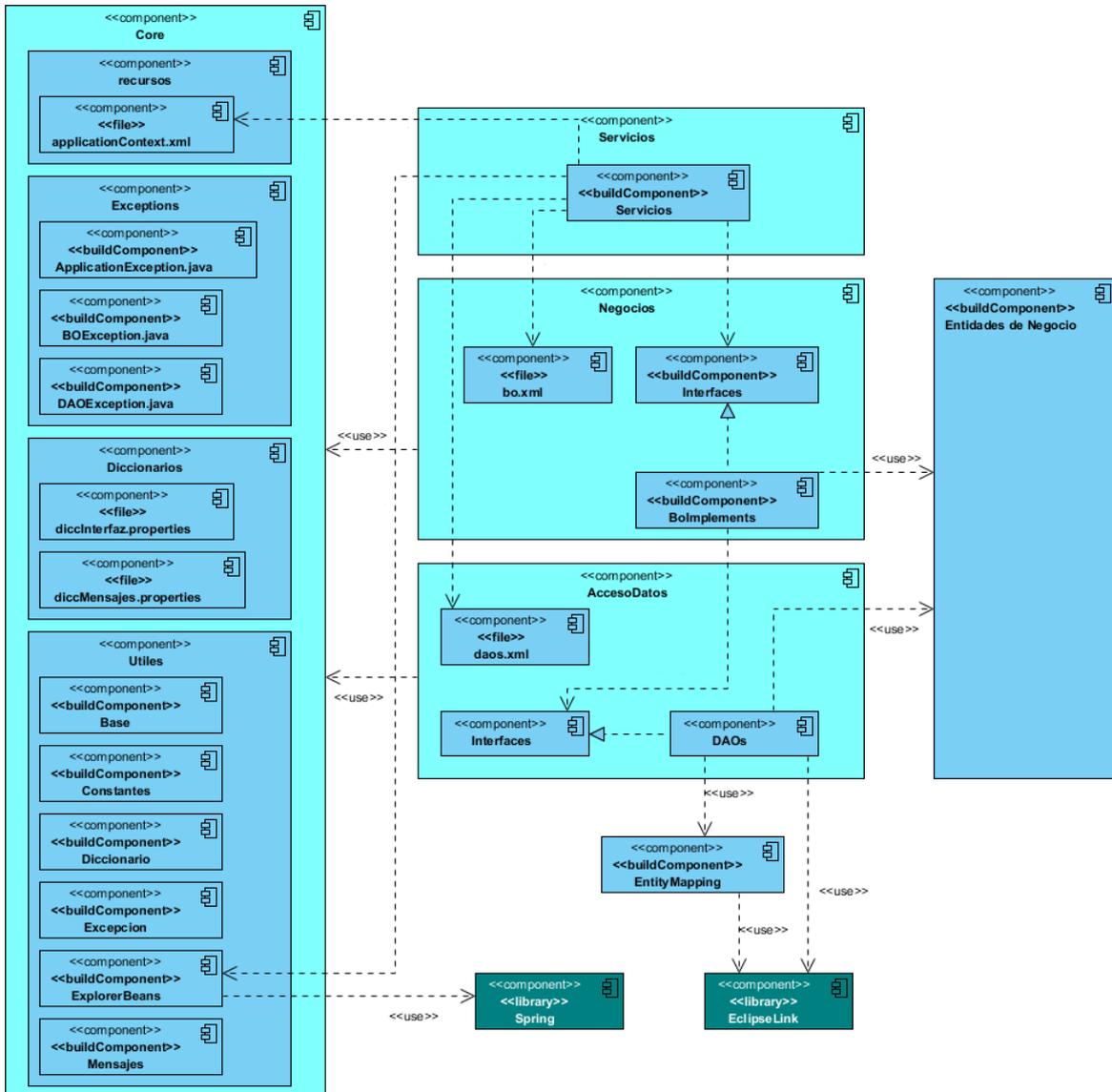
ANEXO 6. DIAGRAMA DE SECUENCIA DEL CASO DE USO "GESTIONAR EMPRESA", SECCIÓN "ADICIONAR".

ANEXOS



ANEXOS

ANEXO 7. DIAGRAMA DE COMPONENTES



ANEXO 8: MÉTRICAS PARA LA CALIDAD DE LA ESPECIFICACIÓN DEL DIAGRAMA DE CASOS DE USO.

Atributo	No.	Métricas
Completitud	1	Número de casos de uso que no tienen descripción resumida.
	2	Número de casos de uso que tienen requisitos omitidos.
	3	Número de casos de uso que no poseen una descripción extendida.

ANEXOS

	4	Número de casos de uso que tienen acciones del flujo de eventos no redactados en función del responsable.
	5	Número de casos de uso que no describen condiciones de excepción relevantes.
	6	Número de casos de uso que no han sido clasificados.
Consistencia	7	Número de casos de uso que tienen un nombre incorrecto.
	8	Número de casos de uso que no representan una interacción observable por un actor.
	9	Número de casos de uso que tienen acciones del flujo de eventos asignados a un responsable que no le corresponde.
	10	Número de casos de uso no aceptados.
	11	Número de casos de uso complejos que no tienen separación del flujo básico y de flujos alternos.
	12	Número de casos de uso que no tienen un usuario responsable.
Correctitud	13	Número de casos de uso en que los requisitos representados no son comprensibles por el usuario.
	14	Número de casos de uso que deben ser modificados para adecuarlos a la funcionalidad del sistema.
	15	Número de casos de uso que deben ser modificados para mejorar el proceso actual.
Complejidad	16	Número de elementos del diagrama que requieren reubicación.

ANEXO 9. DOCUMENTO LISTA DE CHEQUEO PARA EL ARTEFACTO MODELO DE PROCESOS

Este documento contiene una lista de criterios por los cuales se evalúa el modelado de procesos. (Consultar documento con el nombre "[Lista de chequeo para el artefacto modelo de procesos](#)" en la carpeta Artefactos).

ANEXOS

ANEXO 10. DOCUMENTO LISTA DE CHEQUEO PARA DETECTAR INCONSISTENCIAS EN REQM

Este documento contiene una lista de criterios por los cuales se evalúan los requisitos. (Consultar documento con el nombre "[Lista de chequeo para detectar inconsistencias en REQM](#)" en la carpeta Artefactos).

ANEXO 11. DOCUMENTO LISTA DE CHEQUEO PARA EL MODELO DE SISTEMA

Este documento contiene una lista de criterios por los cuales se evalúan las descripciones de los casos de uso. (Consultar documento con el nombre "[Lista de chequeo para el modelo de sistema](#)" en la carpeta Artefactos).

ANEXO 12. DOCUMENTO DISEÑO DE CASOS DE PRUEBAS BASADO EN REQUISITOS

Este documento contiene una lista de criterios por los cuales se evalúan las descripciones de los casos de uso. (Consultar documento con el nombre "[0122_Diseño de Casos de Pruebas basado en Requisitos](#)" en la carpeta Artefactos).

ANEXOS

ANEXO 13. ACTA DE ACEPTACIÓN DEL CLIENTE

		Acta de aceptación	
ACTA DE ACEPTACIÓN			
En cumplimiento del Convenio de colaboración con la Cámara de Comercio y en función de la ejecución del Proyecto Sistema Informatización Registral de la Cámara de Comercio, se hace entrega del producto que se relaciona a continuación.			
Lista de productos que serán aceptados:			
Módulo Registro de Empresas Cubanas			
<ul style="list-style-type: none">• Modelo de proceso de negocio con BPMN.• Especificación de Requisitos de Software.			
Entrega		Recibe	
Proyecto Sistema Informatización Registral de la Cámara de Comercio.		Cámara de Comercio	
Nombre y Apellidos:		Nombre y Apellidos:	
<i>Jorge Manuel José Pedraza</i>		<i>Tónica Quiroga Cucheta</i>	
Cargo: Analista Principal		Cargo: Exp. Desarrollo Empresarial	
Firma: 		Firma: 	
Fecha: 31-5-2012			
Comentarios:			

GLOSARIO DE TÉRMINOS

BPMN: (Business Process Modeling Notation), Notación de modelado de procesos de negocio.

CASE: (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador).

Metodologías: Es una palabra compuesta por tres vocablos griegos: metà (“**más allá**”), odòs (“**camino**”) y logos (“**estudio**”). Son los métodos de investigación que permiten lograr ciertos objetivos en una ciencia. En otras palabras, la metodología es una etapa específica que procede de una posición teórica y epistemológica, para la selección de técnicas concretas de investigación.

Métrica: Medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado.

Proponentes: Dicho de una persona o de una entidad que propone algo o a alguien.

Requisitos: Condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo.

RF: Requisito funcional.

RNF: Requisito no funcional.

RUP (*Rational Unified Process*): Se trata de una metodología para describir el proceso de desarrollo de software.

SIRECC: Sistema de Informatización Registral para la Cámara de Comercio.

UML (*Unified Modeling Language*): Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Está respaldado por el OMG (*Object Management Group*). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.