

Universidad de las Ciencias Informáticas

Facultad 3



**Título: Módulo de Reportes del Sistema de Informatización Registral de la
Cámara de Comercio de la República de Cuba.**

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

Autor(es): Ernesto Pastor González Hernández

Allens Torres Ruíz

Tutor(es): MSc. Danaysa Macías Hernández

Ing. Julio Cesar Prieto Álvarez

La Habana, Cuba

Junio, 2014



CHE

“El mundo camina hacia la era de la informática, la electrónica, las comunicaciones y todos sus derivados. Todo parece indicar que estas ciencias se concientizarán en algo como una medida del desarrollo, quien las domine será un país de vanguardia. Vamos a volcar nuestros esfuerzos en ese sentido con audacia revolucionaria”

Ernesto (Che) Guevara de la Serna



DECLARACIÓN DE AUTORÍA

Declaramos por este medio que nosotros, Ernesto Pastor González Hernández y Allens Torres Ruíz somos los autores del trabajo “Módulo de Reportes del Sistema de Informatización Registral de la Cámara de Comercio” y autorizamos a la Universidad de las Ciencias Informáticas hacer uso del mismo en su beneficio, así como los derechos patrimoniales, con carácter exclusivo.

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

Ernesto Pastor González Hernández
(Autor)

Allens Torres Ruíz
(Autor)

MSc. Danaysa Macías Hernández
(Tutora)

Ing. Julio Cesar Prieto Álvarez
(Tutor)

AGRADECIMIENTOS Y DEDICATORIAS

A nuestros tutores, Danaysa y Julio. A ti Danaysa, no creo que tengamos las palabras para describir tu ayuda, porque has sido para nosotros un ejemplo, porque te debemos los resultados alcanzado durante los últimos tres cursos, en las buenas y en las malas has estado ahí para nosotros, y porque nos has enseñado y convertido en mejores profesionales, porque no has vacilado ni un instante cuando se trata de tus estudiantes Allens y Ernesto. A Julio porque a pesar de las dificultades has sabido formar un equipo de desarrollo que hoy es una gran familia, gracias por el apoyo brindado en la realización de esta tesis, por tu amistad, preocupación y por los consejos dados.

A nuestros profesores de la carrera, a todos y cada uno, en especial a los que han marcado la diferencia, Raynel y Odette. A los profesores del proyecto Ana Cecilia, Reinier, Orlando, Alain, Juan David, Francisco y Abel. Al tribunal de tesis y la oponente por sus críticas constructivas sobre el presente trabajo.

A nuestros compañeros de aula, por compartir 5 años en la universidad y convivir con nosotros.

Allens

A mi mamá, que no bastarían las palabras, ni el mundo, para agradecerle todo lo que ha hecho por mí.

A Estelita por la dedicación y entrega.

A mi compañero de tesis Ernesto.

Al grupo de los de Moa y de los que no, a los que están y a los que se fueron: Rodolfo, Saname, Alejandro, Victor, Alberto, Alberto Enrique, Oscar, Joaquín, Dayron, Yosi, Arturo, Jasiel, Osmani, a los hijos Adrián y Alejandro.

Le dedico la presente tesis:

A mi madre, esa persona especial que siempre ha estado a mi lado en cada momento difícil de mi vida.

A mis hermanos porque no pasa un momento sin que piense en ellos y por ellos.

A Eulicer por todo su apoyo y preocupación.

A mi tía Yadis a quien a pesar de la distancia no le ha impedido apoyarme ni estar conmigo.

A mi padre, que le llegue esta dedicatoria con todo el cariño del mundo.

A mi esposa Estela que siempre ha estado junto a mí cada instante haciéndome la vida más fácil.

A Yannebis, Kirenia, Martha, Odel, Papito y Angelo, esa familia especial que me ha acogido como un igual.

A todos ustedes, de corazón.

Ernesto

El primer agradecimiento y el más especial, por tantas razones que necesitaría un libro para escribirlas. Por darme la vida, por ser la persona más importante del mundo para mí, porque tu vida es mi vida, por tus consejos, atenciones, por enseñarme que no hay nada más grande que el amor de una madre por su hijo, por los momentos duros que hemos pasado juntos, por dedicarme cada minuto de tu vida. Ese agradecimiento es para ti mamá, porque mi vida es tuya y trabajaré cada día para devolverte lo que me has dado.

A mi hermana pequeña Roxi, porque aunque crezcas siempre serás mi hermanita, y agradezco a la vida por darme a una hermana como tú, porque te quiero y eres especial, porque no se puede querer tanto a un hermano como me quieres tú a mí, y te digo una cosa, eres lo mejor que me ha pasado en la vida.

A mi abuela Coralia, por ser la mejor abuela del mundo, por nunca decirme que no a nada, por conocerme y quererme como nieto y como hijo, porque sin ti abuela, la vida no sería igual, por criarme y educarme para ser un hombre mejor, por hablarme con sinceridad y nunca esconderme nada, porque de tu parte no existen reproches, te agradezco y dedico este trabajo por porque fuiste la única persona que me apoyó cuando decidí cursar la carrera y me has ayudado cada minuto a terminarla. Sabes que eres el amor de mi vida.

A mi tía Zeida por estar siempre conmigo, por brindarme tu apoyo incondicional, porque eres mi segunda mamá, y nadie tiene el privilegio de tener dos madres, porque me has tratado como tu propio hijo, porque me has corregido lo malo y congratulado lo bueno.

A mi prima Yeilyn, por su ayuda, por facilitarme la vida en la universidad y confiar en mí para todo, por tenerme presente siempre, te quiero mangote.

A mi abuela Silvia porque soy su nieto preferido y a pesar del divorcio de mis padres has sabido quererme y estar pendiente de mí, te quiero.

A mis tíos y tías, Vladimir, Zenia, Zulema, Anita, Leticia, Justico, Elsitá, porque con nuestras locuras somos una gran familia, la mejor familia del mundo y no creo que exista otra mejor.

A mis primos Vladito, Jesús Daniel, el mangote de Elianys y mi negrita linda Lisi, Thalía, Daniela, somos el piquete perfecto, “the dream team”, porque son más que primos, son mis hermanos menores.

A mi primas de Remedios, Lue y Lidi, por su cariño, atención y apoyo.

A mi padrastro Fernando por criarme como a su propio hijo.

A mi papá.

A mis tres hermanos en la UCI: Crisvel, Allens y Carlos. De ellos, agradecer a Allens mi compañero de tesis, chama, tú sabes que eres un hermano para mí, gracias por compartir los resultados, ayudarme a alcanzar todas las metas propuestas y porque hemos obtenidos los más altos premios en investigación y producción durante los últimos 4 años. A mi amigo y hermano Crisvel, tú sabes el aprecio que te tengo y sabes lo que pienso, porque hay personas en la vida que aparecen y sin darse cuenta se quedan, tú sabes que aquí tienes otro hermano más. A ti Carlos no te voy a decir nada, porque creo que lo he demostrado con creces, sabes dónde estoy para cuando me necesites. Es una pena que estos 5 años acaben ya, pero tengan algo presente, en Villa Clara tienen un amigo.

A Yessi la novia de Crisvel, por tu cariño y por tu amistad y por cuidar de mi hermano tan bien. A mis amigos Anchel y Aliesky. A Estela la novia de mi compañero de tesis, por su amistad.

A mi piquete, ellos sabes quienes son, porque hemos hecho las locuras más grandes de la UCI y somos incomparables, porque redefinimos juntos el concepto de diversión UCI.

Por último, le dedico esta tesis a mis abuelos Jesús y Tata, porque aunque han fallecido y no están conmigo siempre serán un ejemplo para mí.

De forma general, a todas las personas que colaboraron de una forma u otra con la realización de la tesis o que han aportado en nuestra formación como profesionales.

RESUMEN

La Cámara de Comercio de la República de Cuba tiene como misión promover las ofertas exportables de productos y servicios, así como explorar las posibilidades de negocio a escala mundial. La entidad es la encargada de llevar a cabo los procesos de negocio de Registros de Empresas Cubanas, Sucursales Extranjeras, Agencias de Viajes, Importadores y Exportadores y Trámites Migratorios. Debido al proceso de informatización que se encuentra efectuando la entidad, se decidió el desarrollo de la aplicación Sistema de Informatización Registral de la Cámara de Comercio (SIRECC), aplicación de escritorio estructurada por cinco módulos para la gestión de los procesos y un módulo de administración. En consecuencia a que actualmente la Cámara almacena la información para la gestión de los registros en algunas bases de datos en Microsoft Access y hojas de cálculo en Microsoft Excel, se tomó la decisión de desarrollar e incorporar a la aplicación un módulo para la gestión de reportes, que debe contribuir a la solución de los problemas de tiempo de generación de reportes. En el presente trabajo se expone un estudio de varios sistemas generadores de reportes, se muestra la selección de las herramientas y tecnologías a utilizar y se brinda una descripción detallada de la solución de software. Son expresadas las actividades realizadas por cada una de las fases del modelo de desarrollo seleccionado para obtener una solución consistente. Por último se realizó la validación del módulo mediante métricas y pruebas de software.

PALABRAS CLAVE

Informes, Reportadores, Reportes.

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN.....	1
FUNDAMENTACIÓN TEÓRICA	9
1.1. Introducción.....	9
1.2. Descripción de los procesos de negocio informatizados por el proyecto SIRECC	9
1.2.1. Registro de Empresas Cubanas	10
1.2.2. Registro de Sucursales Extranjeras.....	10
1.2.3. Registro de Agencia de Viajes	10
1.2.4. Registro de Importadores y Exportadores.....	10
1.2.5. Registro de Trámites Migratorios	11
1.3. Sistemas Generadores de Reportes	11
1.3.1. Jasper Reports	12
1.3.2. Microsoft SQL Server 2005 Reporting Services (SSRS)	13
1.3.3. Crystal Reports	13
1.3.4. Active Reports	14
1.3.5. PHP Report Maker.....	15
1.3.5. Generador Dinámico de Reportes (GDR)	15
1.3.6. Valoración de los sistemas estudiados	16
1.4. Ingeniería de software	19
1.4.1. Modelo de desarrollo de software: Programa de Mejora	19
1.4.2. Modelado del Negocio y Requisitos	21
1.4.3. Análisis, Diseño e Implementación	22
1.4.4. Pruebas Internas y Pruebas de Liberación	23
1.5. Herramientas CASE.....	24
1.5.1. UML 2.0.....	25
1.5.2. Visual Paradigm for UML 8.0 Enterprise Edition	25
1.6. Tecnologías.....	26
1.6.1. Marco de Trabajo GEFORT	26
1.6.2. Servidor de Aplicaciones GlassFish 3.1.2.....	27
1.6.3. Sistema Gestor de Bases de Datos PostgreSQL 9.3	27
1.6.4. Entorno de desarrollo integrado.....	28
1.7. Validación de las variables	29
1.7.1. Variable independiente	29
1.7.2. Variable dependiente.....	31
1.8. Conclusiones del capítulo	31

SOLUCIÓN DE SOFTWARE	32
2.1. Introducción	32
2.2. Modelación del negocio	32
2.3. Levantamiento de Requisitos	33
2.3.1. Requisitos funcionales	34
2.3.2. Requisitos no funcionales	38
2.4. Análisis y Diseño	40
2.4.1. Diseño Arquitectónico	41
2.4.2. Diseño de Clases	42
2.4.3. Patrones de diseño empleados en la solución	43
2.4.4. Modelo de Datos	44
2.5. Implementación	45
2.5.1. Diseño de Componentes	45
2.5.2. Modelo de Despliegue	47
2.5.3. Estándares de Codificación	48
2.5.4. Implementación del componente	48
2.5.5. Seguridad del sistema	50
2.6. Pruebas	51
2.7. Conclusiones del capítulo	52
ANÁLISIS DE RESULTADOS	53
3.1. Introducción	53
3.2. Validación de las fases de desarrollo	53
3.2.1. Validación de la modelación del negocio y los requisitos de software	53
3.2.2. Validación del diseño y la implementación	54
3.3. Validación de las variables de investigación	57
3.3.1. Alcance	57
3.3.2. Calidad	58
3.3.3. Tiempo	64
3.4. Conclusiones del capítulo	65
CONCLUSIONES	66
RECOMENDACIONES	67
BIBLIOGRAFÍA	68
ANEXOS	71
GLOSARIO	91

ÍNDICE DE IMAGENES

Imagen 1: Fragmento del Modelo de Datos del Módulo de Reportes	44
Imagen 2: Fragmento del Diagrama de Componentes (RF-1 Generar Argumentación)	46
Imagen 3: Diagrama de Despliegue del Módulo de Reportes.....	47
Imagen 4: Estructura del Proyecto SIRECC y el Módulo de Reportes.....	49
Imagen 5: Gráfica de Índice de Requisitos Implementados.....	58
Imagen 6: Gráfica de Porcentaje de Requisitos Implementados	58
Imagen 7: Grafo de flujo asociado al método llenarMapaReporte().....	59
Imagen 8: Vista Lógica del Marco de Trabajo GEFORT.....	82
Imagen 9: Vista física de los estilos arquitectónicos empleados en la solución	82
Imagen 10: Cálculo del TOC para los atributos Responsabilidad y Complejidad de Implementación.....	83
Imagen 11: Cálculo del TOC para el atributo Reutilización	83
Imagen 12: Modelo de Dominio del Módulo de Reportes	84
Imagen 13: Diagrama de Clases del Diseño	85
Imagen 14: Generar documento Criterios de la junta ejecutiva sobre admisión como asociado.....	86
Imagen 15: Diagrama de Componentes General Módulo de Reportes	87
Imagen 16: Diagrama de Componentes del Módulo de Reportes (RF-1 Generar Argumentación)	88
Imagen 17: Modelo de Datos General Módulo de Reportes.....	89
Imagen 18: Evaluación de los atributos de calidad por los clientes.	90

ÍNDICE DE TABLAS

Tabla 1: Operacionalización de las variables de investigación.	5
Tabla 2: Resumen de la descripción del requisito funcional (RF-2).....	37
Tabla 3: Resumen de descripción del diseño de clases del Módulo de Reportes.....	42
Tabla 4: Cálculo del TOC para los atributos Responsabilidad y Complejidad de Implementación.....	55
Tabla 5: Cálculo del TOC para el atributo Reutilización.	56
Tabla 6: Caminos básicos del flujo.....	61
Tabla 7: Resultados de la Lista de Chequeo de Usabilidad de Desktop.....	63
Tabla 8: Tabla comparativa de los sistemas estudiados.	76
Tabla 9: Cálculo de dominio del TOC.....	76
Tabla 10: Modelo de calidad definido por la ISO.....	78
Tabla 11: Descripción de las clases implementadas.	81

INTRODUCCIÓN

La Cámara de Comercio de la República de Cuba fue creada bajo la ley No. 1091 del 1ro de febrero de 1963 y tiene como objetivo potenciar e intercambiar información valiosa en torno a las posibilidades de negocio a escala mundial, 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 asociados, a las sucursales de compañías extranjeras acreditadas en el país, así como a empresarios extranjeros interesados en negociar con Cuba. Según el compromiso social como institución cubana de responder al crecimiento de la sociedad, con amplio intercambio y desarrollo de todos los componentes asociados, la Cámara tiene como misión promover el desarrollo de la empresa cubana asociada, donde el país obtenga los mejores resultados socio-económicos. (Cámara de Comercio de la República de Cuba, 2009)

La Cámara de Comercio de la República de Cuba gestiona los Registros de Empresas Cubanas afiliadas, Sucursales Extranjeras, Agencias de Viajes, Exportadores e Importadores y Trámites Migratorios, que son servicios comerciales sobre mercados externos a empresas cubanas asociadas a la Cámara de Comercio y viceversa, además representan cada una de las áreas de la entidad. Estos servicios son especializados en temas vinculados al comercio tales como: servicios de información comercial sobre el mercado cubano, tanto a usuarios nacionales como a extranjeros. De los procesos anteriormente mencionados se dificulta la obtención de la información en tiempo, como los documentos certificadores referentes a cada uno de los procesos, estadísticas, listados, entre otros.

Para la obtención de la información de los documentos se realizan procedimientos manuales y se utilizan herramientas de ofimática y sistemas informáticos independientes como los software privativos Microsoft Access, hojas de cálculo en Microsoft Excel y formatos modificables como DOC (Document, por su traducción al español Documento), certificado en (Jorrín Perdomo, 2013). Estas formas de gestión de los informes en la Cámara dificulta la elaboración de la documentación en tiempo real.

Durante la entrevista inicial realizada, se comprobó que la información que se obtiene de los procedimientos anteriormente mencionados se encuentra dispersa, incompleta, repetida y/o con errores

en bases de datos y en archivos físicos, por lo que obtener un tiempo pequeño de consulta accediendo estos datos se hace extremadamente engorroso. Por tanto se dificulta el análisis de los datos que se generan, la elaboración de informes que apoyen las decisiones comerciales y la representación visual de información bajo determinadas condiciones y diferentes formatos en momentos imprescindibles para la empresa. (Ver Anexo 1)

Existen datos como el logotipo de la entidad, formato correcto de la fecha, departamento y el registro del proceso, que no se tienen en cuenta para el registro de la información correspondiente a los procesos antes mencionados, como consecuencia, obtener en tiempo un documento estandarizado puede convertirse en una actividad engorrosa. Los reportes correspondientes a cada una de las áreas son realizados de manera manual, dificultando el control de los datos en tiempos determinados. Realizar una revisión completa de la información puede tornarse una actividad ardua, corriendo el riesgo de duplicación, dificultando la creación, edición y traslado; provocando así su atraso, o el carecer de la información adecuada llegando a no satisfacer los requerimientos de la entidad. Estos informes que son generados en diferentes áreas de la entidad como los documentos de aprobación de solicitudes de visados aprobados por el Departamento Jurídico, los Criterios de la junta ejecutiva sobre admisión como asociado, dictaminado por el Departamento de Formación y Membresía, llegan a tener carácter inmediato en su aprobación, por lo que el tiempo es un factor de vital importancia para la entidad. Otro factor influyente es la inexistencia de una vía directa de intercambio de la información donde las partes involucradas puedan consultar los documentos.

En todos los niveles de la organización se comprobó mediante las entrevistas realizadas a los especialistas principales de cada una de las áreas, que la fluidez de la información reportada o estadística no era lo suficientemente rápida y su representación en formato de tablas presenta una alta complejidad. Se pudo apreciar que la documentación viaja por canales inseguros al trasladarla personalmente de un departamento a otro para su consulta y aprobación en diferentes dispositivos de almacenamiento, pudiendo incurrir esto en pérdida o modificación de datos, poco control y atraso en el proceso. Las salidas de cada uno de estos procesos registrales son de valor agregado al negocio de la Cámara; donde luego de procesados los datos, la institución necesita la redacción de complejos informes, tablas estadísticas y datos generados por las correspondientes áreas. En consecuencia no se cuenta con la información centralizada, con el objetivo de administrarla y entregarla en tiempo real. Se

pudo establecer que no existía uniformidad en los informes, ni contaban con un estándar establecido para los documentos generados, en consecuencia, lograrlo como tarea diaria, mensual o anual puede demorar más del tiempo deseado. Tampoco es posible explotar al máximo las características y potencialidades de estos informes finales, su diseño no permite la reestructuración de la información en forma de encabezado, cuerpo del documento y pie de firma, presentada en el Manual de Identidad de la Cámara de Comercio.

Estas deficiencias encontradas en el proceso de estudio preliminar y las encontradas en las entrevistas, evidencian que las necesidades de información y elaboración de informes en tiempo no son completas o no brindan la información suficiente. Esta situación impide a la Cámara garantizar la creación y modificación de reportes de los procesos registrales.

En correspondencia con el auge de las Tecnologías de la Información y las Comunicaciones (TIC) a nivel mundial, la entidad se encuentra efectuando un proceso de informatización, abarcando la mejora de sus procesos de negocios y digitalización de sus datos, con el objetivo de mejorar los tiempos de respuesta y el manejo correcto de la información. La necesidad de existencia de una aplicación que se encargue de los procesos de negocio referente a cada una de las áreas de la empresa, fue constatada durante la fase de estudio preliminar. Durante las tareas a realizar en cada uno de estos procesos se pretende la generación de la documentación e informes correspondientes a la entidad, en consecuencia, surge la necesidad de informatización de reportes de información por el sistema a desarrollar.

Por lo que el **problema a resolver** se centró en ¿Cómo disminuir el tiempo de generación de reportes de información de los procesos registrales Afiliación de Empresas Cubanas, Sucursales Extranjeras, Importadores y Exportadores, Agencias de Viaje y Trámites Migratorios en la Cámara de Comercio de la República de Cuba?

Se define por lo tanto como **objeto de estudio**: Generación de reportes.

Para dar solución al problema, se propone como **objetivo general** desarrollar un Módulo de Reportes para el Sistema de Informatización Registral de la Cámara de Comercio, que permita disminuir el tiempo de generación de reportes de información de los procesos registrales referentes a la Afiliación de Empresas Cubanas, Sucursales Extranjeras, Importadores y Exportadores, Agencias de Viaje y Trámites Migratorios.

Inciendo en el **campo de acción**: Herramientas para la generación de reportes en plataforma Java.

Al logro del objetivo general tributan los siguientes **objetivos específicos**:

- ✓ Realizar el marco teórico de la investigación que permita conocer las principales tendencias en cuanto al desarrollo de reportes en los proyectos de software.
- ✓ Desarrollar el Módulo de Reportes para el Sistema de Informatización Registral de la Cámara de Comercio que permita resolver los problemas de tiempo en la generación de reportes de información.
- ✓ Validar el Módulo de Reportes a partir de la ejecución de técnicas y métricas aplicables al desarrollo de la solución que permitan evaluar los resultados.

Para cumplir con los objetivos específicos se planificaron las siguientes **tareas de la investigación**:

1. Estado del arte de las tendencias del desarrollo de reportes en proyectos de software.
2. Descripción de los procesos de negocio que generan reportes en la Cámara y su relación funcional con el resto de los procesos de la entidad, destacando su importancia.
3. Resumen de buenas prácticas en el desarrollo de software teniendo en cuenta las fases de: modelación del negocio, requisitos, análisis y diseño, implementación, pruebas internas y pruebas de liberación, tomando como guía para el desarrollo de estas fases el modelo de desarrollo propuesto por el centro CEGEL (Centro de Gobierno Electrónico).
4. Definición de los requisitos funcionales y no funcionales de los Procesos Registrales de la Cámara.
5. Desarrollo de los diagramas establecidos por el modelo propuesto.
6. Implementación de los requisitos funcionales de los Procesos Registrales.
7. Realización de pruebas de software de los Procesos Registrales.
8. Análisis de las variables de la investigación.

Planteándose la siguiente **hipótesis**: Con la utilización de un módulo para la generación de reportes se contribuirá a disminuir el tiempo de obtención de reportes de información de los procesos registrales referentes a la Afiliación de Empresas Cubanas, Sucursales Extranjeras, Importadores y Exportadores, Agencias de Viaje y Trámites Migratorios.

Operacionalización de las variables:

Variab Independientes	Variab Dependientes	Dimensiones	Indicadores	Unidad de Medida
Módulo de Reportes		Alcance	Cumplimiento de los requisitos	Índice de requisitos implementados = (Cantidad de requisitos implementados) / (Cantidad total de requisitos)
		Calidad	Calidad Externa (Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Facilidad de Mantenimiento, Portabilidad)	Cálculo del valor de calidad según el modelo definido por la ISO
			Calidad Interna	Índice de no conformidades detectadas en las Pruebas de Liberación
	Tiempo de generación de reportes	Tiempo	Tiempo de generación de reportes	Índice de no conformidades detectadas con el uso de las listas de chequeo de usabilidad
				Tiempo de generación de los documentos de salida < 2 segundos
				Tiempo de generación de las tablas del sistema < 3 segundos

Tabla 1: Operacionalización de las variables de investigación.

Para dar cumplimiento a las tareas de investigación propuestas se utilizan los métodos científicos:

Métodos teóricos

- ✓ **Analítico – Sintético:** Después de hacer un análisis de los referentes teóricos y la bibliografía en cuestión se hizo una extracción y síntesis de los elementos más significativos relacionados con los diferentes componentes de reportes y las herramientas de generación de reportes. Este método se utiliza también para tomar posiciones en cuanto a los resultados arrojados en las entrevistas realizadas. Sirvió como base teórica para el desarrollo de la tesis.
- ✓ **Histórico – Lógico:** Este método se utiliza para resolver las deficiencias en la Cámara; no se pueden soportar todos los reportes necesarios en la entidad, de los cuales dependen procesos sustanciales. Se sustentará a partir del estudio de la evolución que han tenido los conceptos más importantes que se tratan en la tesis como es el caso de los reportes, reportadores dinámicos, componentes, entre otros.
- ✓ **Método Hipotético - Deductivo:** Teniendo como primicia la hipótesis planteada y siguiendo reglas de deducción se generarán nuevos conocimientos y predicciones, las que posteriormente son sometidas a verificaciones empíricas.
- ✓ **Modelación:** Permite crear abstracciones del producto final para validar los requisitos con el cliente y obtener un versión del sistema original.

Métodos empíricos

- ✓ **Medición:** Con la utilización de este método se obtiene información cuantitativa acerca de las propiedades o cualidades de los elementos de estudio, permite comparar magnitudes medibles y conocidas como en los cálculos de la Calidad. Permite obtener los resultados de los tiempos de generación de reportes del sistema.

Métodos particulares

- ✓ **Entrevistas:** Este método facilita obtener información del proceso de generación de reportes, intercambiando directamente con las personas involucradas en el desarrollo de los procesos registrales de Empresas Cubanas, Sucursales Extranjeras, Agencias de Viajes, Exportadores e Importadores y Trámites Migratorios. Además permite la recopilación de la información para el almacenamiento de requisitos del componente.
- ✓ **Encuesta:** Mediante cuestionarios pre-elaborados se realizan preguntas a varios especialistas de la entidad, con el propósito de obtener la información necesaria para comprender los procesos de negocio de la entidad. La información se obtendrá de forma escrita y firmada por el objeto de encuesta para obtener la certificación de la información.

Se define la siguiente **estructura del documento**, quedando conformado en tres capítulos:

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Se describe el estudio de la teoría sobre la investigación y el marco de los conocimientos acerca de las herramientas a trabajar. Se realiza una síntesis en relación al lenguaje de programación, el Gestor de Base de Datos, la metodología a utilizar así como la arquitectura del componente. Se analizan las investigaciones necesarias para mostrar el proceso de reportes, realizando un estudio del estado el arte. Se hará un estudio del modelo de desarrollo propuesto por el Centro de Gobierno Electrónico (CEGEL) y se conceptualizarán elementos importantes para la captura y la validación de requisitos; así como los distintos patrones de diseño, las herramientas y tecnologías a utilizar para el diseño e implementación del componente.

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

En este capítulo se describe la propuesta de solución del componente a desarrollar y el proceso que será informatizado. Se presentarán los requisitos funcionales y no funcionales con los que debe cumplir el componente propuesto. Se hará referencia a las técnicas empleadas tanto en la captura como en la validación de los requisitos. Se mostrarán el modelo de datos del sistema y el diagrama de clase del diseño que será implementado. Se explicará la utilización de un conjunto de patrones dentro del diseño del componente. Se realizará una breve descripción del Marco de Trabajo utilizado, así como los estándares a utilizar durante la implementación del componente, describiendo las clases y funcionalidades. En consecuencia, se obtendrá un conjunto de artefactos establecidos por la metodología de desarrollo propuesta en el capítulo 1. Por último se abordaran las actividades realizadas durante la fase de pruebas.

CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

En este capítulo se presenta la validación de las variables para fundamentar el funcionamiento correcto del módulo. Se validará el diseño de la solución a través de métricas, obteniendo los resultados arrojados de las variables dependiente e independiente.

1

FUNDAMENTACIÓN TEÓRICA

“Si las computadoras y máquinas automáticas pueden obrar milagros en la creación de bienes materiales y servicios, ¿Por qué no podríamos servirnos todos de la ciencia que ha creado el hombre con su inteligencia para el bienestar humano?”

Fidel Castro Ruz

1.1. Introducción

El presente capítulo se enfoca en la realización de un análisis de las fuentes bibliográficas para obtener información actualizada acerca de la teoría necesaria para la investigación. Se describe el marco teórico de los principales reportadores existentes a escala mundial y nacional, así como la teoría referente a la Ingeniería de Software (ISW) y sus disciplinas. Se esclarecerán los principales aspectos del modelo de desarrollo por el que se dirige esta investigación, teniendo en cuenta el Marco de Trabajo seleccionado y las especificidades propias del proyecto. Se realizará un análisis de las tecnologías, lenguajes de programación y herramientas a utilizar. Por último se abordará sobre la validación de las variables de la investigación.

1.2. Descripción de los procesos de negocio informatizados por el proyecto SIRECC

Como parte de la gestión de registros de Empresas Cubanas, Sucursales Extranjeras, Agencias de Viaje, Exportadores e Importadores y Trámites Migratorios que se realizan en la Cámara de Comercio de la República de Cuba se establece la necesidad de elaboración de reportes de información.

Actualmente, la aplicación SIRECC gestiona la información de los procesos de negocio de la Cámara, informatizando la anterior forma de gestión de la información, realizada de forma manual y/o con la utilización del software privativo: Microsoft Access y hojas de cálculo en Microsoft Excel. Atendiendo a esto se identificaron las salidas del sistema con el objetivo de resolver los problemas que en materia de información se presentan en la gestión de los procesos.

1.2.1. Registro de Empresas Cubanas

El módulo Registro de Empresas Cubanas permite gestionar la información de los procesos: afiliación de la empresa, actualización de los datos, cancelación de la empresa y la gestión de los pagos de las cuotas anuales que por concepto de afiliados deben pagar. Además de reportes y documentación como: solicitud de admisión como asociado, criterios de la junta ejecutiva sobre admisión como asociado, carta de aprobación, así como documentos de certificación, observaciones, entre otros.

1.2.2. Registro de Sucursales Extranjeras

El módulo Registro de Sucursales Extranjeras permite gestionar la información de las empresas o compañías extranjera interesadas en negociar con Cuba, desde el registro de sus datos con el expediente y los documentos asociados. Permite además gestionar los datos de las oficinas secundarias que estas pueden abrir, así como los pagos que realizan las sucursales de las cuotas correspondientes. También se encarga de brindar reportes como licencias de inscripción y anexos.

1.2.3. Registro de Agencia de Viajes

El módulo Registro de Agencias de Viajes, permite inscribir las agencias de viajes nacionales, extranjeras y extranjeras representadas por nacionales, realizar la renovación de la inscripción y cancelar la inscripción. Brinda reportes como la licencia de inscripción y renovación o renovación de agencias de viajes, licencias, referencias de modificación de resolución, referencias de notificación.

1.2.4. Registro de Importadores y Exportadores

El módulo Registro de Importadores y Exportadores permite registrar la información de las empresas importadoras y exportadoras, además de generar los documentos correspondientes a su registro como anexo de inscripción, documentos de certificación de importadora y exportadora, entre otros.

1.2.5. Registro de Trámites Migratorios

El módulo Registro de Trámites Migratorios permite gestionar la información de los procesos de trámites, entre los que se encuentran: registro de funcionarios, registro de trámites migratorios tales como: tramitación de visados comerciales, cambio de clasificación migratoria, cambio de dirección, prórroga de residencia temporal, solicitud de visados múltiples, prórroga de visado, actualización de permiso de trabajo, renovación de permiso de trabajo, solicitar duplicado de permiso de trabajo, solicitudes de permiso de trabajo. Además permite generar reportes como la plantilla AO-3 (solicitudes migratorias), AO-1 (solicitud de trámites), entre otros.

En el siguiente epígrafe se analizarán diferentes sistemas generadores de reportes enmarcados a nivel nacional e internacional, donde se describirán sus principales características y funcionalidades.

1.3. Sistemas Generadores de Reportes

Los generadores de reportes son herramientas complementarias de los sistemas de información. Utilizan un lenguaje transparente para el usuario por medio del cual realiza consultas y obtiene información de ella en forma de reporte. Partiendo de que la Cámara de Comercio de la República de Cuba, tiene características especiales en la elaboración de reportes y necesitan de un sistema que brinde estos informes para permitir la correcta evaluación de sus procesos de negocio, el estudio del estado del arte se centró fundamentalmente en estudiar y comprender el tratamiento que recibe este tema en sistemas ya existentes, debido a las implicaciones que posee. (Comercio, 2012)

Para comprender el ámbito de la solución propuesta se analizaron sistemas de reportes, tanto nacionales como foráneos con el objetivo de evaluar el comportamiento de las características en cuanto a: arquitectura, tipo de software, multiplataforma, generación de reportes dinámicos, personalización al detalle, escalabilidad, formato de las salidas y generación de subreportes. Dentro de los sistemas analizados se encuentran: Dynamic Report Generator, Jasper Report, PHP Reports Maker, Crystal Reports, Active Reports, Microsoft SQL. Estos sistemas son los más reconocidos en cuanto a la generación de reportes y su utilización se extiende en diferentes tipos de aplicaciones (incluyendo aplicaciones del tipo registral). Existe experiencia en la utilización de estos reporteadores en la

universidad, además de contar con abundante documentación y soporte en línea. Seguidamente se realiza la caracterización de cada uno de ellos. Los parámetros de selección de estas herramientas de reportes y la justificación de su selección puede encontrarla en (Suárez del Villar, y otros, 2013).

1.3.1. Jasper Reports

Jasper Reports es una librería de código abierto poderosa y flexible para la generación y gestión de reportes permitiendo la creación de informes en Java. Permite la creación de reportes que pueden ser presentados en la web o se pueden crear ficheros en diversos formatos (Ver Anexo 3), para que los datos puedan ser procesados con otras aplicaciones.

Admite la creación de reportes dentro de reportes (subreportes), permite añadir enlaces en los reportes, lo que haría más fácil y rápida la navegación entre las diferentes secciones del informe cuando este es muy largo. (JasperSoft, 2011)

Las principales características de Jasper Reports son:

- ✓ Permite una diagramación flexible de los reportes: Los reportes se pueden dividir en secciones opcionales que son: título del reporte, el encabezado de página, una sección para los detalles del reporte, el pie de página y una sección de resumen que aparece al final del reporte.
- ✓ Permite que los desarrolladores le surtan datos en varias formas: esto es que los desarrolladores pueden pasar datos a los reportes por medio del paso de parámetros. Estos parámetros de reportes pueden ser instancia de cualquier clase de Java.
- ✓ Los reportes son capaces de presentar los datos de manera textual o a través de gráficos: no sólo son capaces de mostrar los datos que le son pasados sino que pueden generar o calcular con esos datos otros datos de forma dinámica y mostrarlos.

No es una herramienta por sí sola, por lo que no se puede instalar. Para utilizar Jasper Reports es necesario añadirlo a las aplicaciones Java por medio de la inclusión de su librería al classpath (por su traducción al español, ruta de la clase) de la aplicación. En cuanto a las especificaciones de sistemas operativos se puede utilizar en cualquier entorno, siempre y cuando exista una implementación de la

máquina virtual de Java. Su licencia se distribuye bajo los términos de la Licencia Pública para Librerías GNU (por su traducción al inglés GNU Library Public License), por lo que es software libre y está respaldado por una gran comunidad internacional de desarrollo. (JasperSoft, 2011)

1.3.2. Microsoft SQL Server 2005 Reporting Services (SSRS)

Reporting Services es un componente clave de SQL Server 2005. Es una plataforma de elaboración de informes basada en servidor que se puede utilizar para crear y administrar informes tabulares, matriciales, de gráficos y de formato libre con datos extraídos de orígenes de datos relacionales y multidimensionales. Se pueden visualizar y administrar mediante una conexión basada en la web. (Microsoft, 2012)

Contiene un servidor de informes que aloja y procesa informes en diversos formatos (Ver Anexo 3). Los informes incluyen características interactivas basadas en la web API (Application Programming Interface o por su traducción al español interfaz de programación de aplicaciones) que permite a los programadores integrar o extender procesamiento de datos e informes en aplicaciones personalizadas. En SSRS, los usuarios pueden enrutar directamente los trabajos de impresión, sin necesidad de exportarlos antes. Sistema Operativo: Microsoft Windows, preferiblemente en sus versiones para servidores. Licencia: se distribuye bajo licencia privativa perteneciente a Microsoft Corporation. (Microsoft, 2012)

Reporting Services permite que los informes incluyan características interactivas basadas en la web: informes de varios niveles de detalle, que permiten la navegación por distintas capas de datos; los informes con parámetros, que admiten el filtro de contenido en tiempo de ejecución; o los informes de formato libre, con diseños verticales, anidados o adyacentes.

1.3.3. Crystal Reports

Crystal Reports es un producto de alta tecnología, además de ser una herramienta potente es fácil de usar para el diseño y generación de reportes a partir de datos almacenados en una base de datos u otra fuente de información.

Permite transformar rápidamente cualquier fuente de datos en contenido interactivo, integrar estrechamente capacidades de diseño, modificación y visualización en aplicaciones .NET (derivado del inglés "Network"), Java o COM (Component Object Model o por su traducción al español Modelo de Objeto Componente), y además permite a los usuarios finales acceder e interactuar con los reportes a través de portales web, dispositivos móviles y documentos de Microsoft Office. (Business Objects, 2012)

Se puede utilizar en sistemas operativos como, Microsoft Windows XP con Service Pack (SP) 2, Windows Server 2003 con SP 1 o posterior y se distribuye bajo los términos de la EULA (End User Licensing Agreement o por su traducción al español Licencia de Usuario Final), por lo que es software privativo y de uso restringido mediante el pago de patente.

Los visores web avanzados habilitan a los usuarios finales para realizar búsquedas dentro de los datos de un reporte y exportarlas posteriormente a Microsoft Excel, Word y páginas HTML con el vínculo dinámico al reporte original. Adicionalmente, el reporte completo puede ser exportado a una variedad de formatos incluyendo XML (Extensible Markup Language o por su traducción al español lenguaje de marcas extensible), PDF (Portable Document Format o por su traducción al español Formato de Documento Portátil), HTML (HyperText Markup Language o por su traducción al español Lenguajes de Marcas de Hipertexto) y Microsoft Excel. (Business Objects, 2012)

1.3.4. Active Reports

Active Reports es una herramienta de generación de reportes que brinda posibilidades de personalización, alto rendimiento e interfaz intuitiva. Se distribuye bajo licencia privativa perteneciente a Grape City. Es un componente de informes .NET, entre las características claves de este componente figuran la personalización, rendimiento alto, alta calidad y numerosas prestaciones. Admite exportaciones de datos a todos los formatos de archivo habituales (Ver Anexo 3).

Dentro de las características principales se tiene que los informes se crean dentro de Visual Studio .NET y se compilan directamente en el ejecutable y dado que Active Reports es totalmente administrado, no presenta dependencias de aplicaciones de terceros. (Grape City, 1997)

1.3.5. PHP Report Maker

Es una poderosa herramienta de informes que se pueden generar informes Web PHP dinámicos de MySQL, PostgreSQL, Microsoft Acces, Microsoft SQL Server y base de datos Oracle. Las páginas Web generadas son PHP puro, está diseñado para la alta flexibilidad y permite generar reportes que se adapten a las necesidades. Los códigos generados son limpios y fáciles de personalizar. PHP Report Maker puede ahorrar tiempo y su uso es adecuado tanto para principiantes como para desarrolladores experimentados.

Con PHP Report Maker, los usuarios pueden exportar los informes a formatos como PDF, Word o Excel. Dentro de sus características cuenta con: informes de resumen y detalle, propiedades avanzadas de gráfico, plantilla personalizable y sincroniza los ajustes de proyectos con los cambios en la base de datos.

1.3.5 Generador Dinámico de Reportes (GDR)

Constituye una de las herramientas que posibilitan la inteligencia de negocios ya que genera vistas agregadas de datos para mantener a la gerencia informada sobre el estado de su negocio. Proporciona soporte a todo el ciclo de vida de los reportes, provee accesibilidad a la información emitiendo reportes a través de un navegador web u otros formatos estándares y está orientada al usuario final, brindando herramientas como el Diseñador de reportes que permite la confección de informes de forma interactiva en la web. (Abreu Medina, 2012)

Este diseñador permite personalizar al detalle la salida de los reportes y la forma en que la información será visualizada. Presenta una arquitectura modular y las posibilidades de escalabilidad por medio de extensiones y complementos que se integrarán a los procesadores principales que componen el servidor de reportes, esta información puede ser consultada en (Abreu Medina, 2012). Seguidamente, un análisis comparativo entre estas tecnologías.

1.3.6 Valoración de los sistemas estudiados

A partir del estudio de los sistemas generadores de reportes se implantó un análisis comparativo de los principales parámetros que deben tener para satisfacer las necesidades de la Cámara de Comercio (Ver Anexo 3).

Para realizar este análisis comparativo entre los sistemas generadores de reportes se estableció un grupo de propiedades ideales a cumplir: arquitectura, tipo de software, multiplataforma, generación de reportes dinámicos, personalización al detalle, escalabilidad, formato de las salidas y generación de subreportes. Estas propiedades son necesarias para el manejo correcto de los reportes en el módulo.

Como conclusión del análisis se dedujo, que los diseñadores de reportes analizados poseen propiedades similares; una de ellas es la generación de subreportes, la cual el 100% de estos diseñadores la tienen de manera implícita. En cuanto al diseño personalizado, que es una característica necesaria para los reportes del módulo a desarrollar se evidenció que 83,33% de estos generadores de reportes permiten la personalización al detalle, permitiendo el enriquecimiento visual de la información. Por otra parte, a estos datos se suma que el 100% de los objetos estudiados generan reportes dinámicos y el 66,67% de estos son escalables permitiendo que la solución puede adaptarse y reaccionar a las necesidades que se presenten sin perder calidad.

Dos propiedades esenciales que se enmarcaron en el estudio fueron el tipo de software y su capacidad de adaptación a plataformas distintas (multiplataforma). La primera de estas propiedades a cumplir arrojó una equivalencia del 50% perteneciente a software libre y la misma cantidad a software privativo. En cuanto a la segunda característica antes mencionada se estableció que el 66,67% son multiplataforma y el 33,33% restante pertenece a un sistema operativo específico. En cuanto a la arquitectura de estos sistemas se pudo apreciar que el 50% de ellos son librerías que se pueden agregar a aplicaciones, mientras que el 16,66% de diseñadores de reportes presentan una arquitectura modular, con igual cifra las extensiones y los de arquitectura servidor.

Las herramientas estudiadas permiten extraer datos de varias fuentes, así como generar y exportar los informes en varios formatos dentro los que se destaca PDF y DOC, cumpliendo con los requerimientos de los clientes.

Luego de los resultados arrojados en el análisis anterior se impuso la elección de la herramienta más conveniente en dependencia de las necesidades de los clientes así como las funcionalidades que ofrece cada una como base para el soporte de una aplicación que responda a las necesidades.

Por lo que se concluye que:

- ✓ Dynamic Reports Generator es una aplicación modular pero no permite el trabajo en múltiples plataformas siendo esto una necesidad de los clientes. Este reportador favorece las aplicaciones web, siendo esto otra desventaja para el Sistema Registral como aplicación de escritorio.
- ✓ PHP Reports Maker, pese a ser una herramienta de software libre, es una extensión para aplicaciones web y no ofrece dentro de sus atributos la escalabilidad para asumir los cambios que se presenten en cuanto a las necesidades de reportes.
- ✓ Cristal Reports tiene como principal deficiencia que es una librería de software privativo por lo que no se ajusta a las políticas del centro de desarrollo CEGEL y por lo tanto a las políticas de la universidad. Solo permite el trabajo con la plataforma Windows, incidiendo esto de manera contraria a lo necesitado por los clientes y presenta restricciones en cuanto a la personalización al detalle de estos informes. De igual forma, en estas características incurre Microsoft SQL Server Reporting Services.
- ✓ Active Reports es un sistema que cumple con algunos de los requerimientos, pero una de las desventajas contrasta con los principales intereses de la Cámara de Comercio, y es el hecho de ser software privativo, donde tampoco se presenta como librería multiplataforma. Al igual que PHP Reports no es escalable.
- ✓ En cambio Jasper Reports es una herramienta que se distribuye bajo licencia pública, es multiplataforma, que maneja muy fácilmente los subreportes y que satisface las necesidades de los clientes, genera reportes dinámicos, permite la personalización de cada uno de los atributos, es escalable y es una librería que permite generar en diferentes formatos (incluyendo el PDF).

Después de lo anteriormente planteado se propone la utilización de la librería Jasper Reports para la creación del Módulo de Reportes del proyecto SIRECC, por las principales ventajas que brinda como tecnología de código abierto flexible para generar reportes. Permite la estandarización de reportes en formato PDF y concurre en la integridad de la información, donde se establece también un formato de impresión fácil y estable. En cuanto a la creación de tablas dinámicas, es una herramienta que facilita diseño y configuración de tablas dinámicamente. Agregando a esto la personalización independiente de las diferentes secciones de los reportes como el título del reporte, el encabezado de página, una sección para los detalles del reporte, el pie de página y una sección de resumen, facilitando así el cumplimiento de lo establecido en el Manual de Identidad Visual de la Cámara de Comercio bajo la existencia de las normas, modelos y patrones ahí expresados.

Luego de seleccionado el sistema de generación de reportes, se dio paso al establecimiento de las buenas prácticas de la ingeniería de software. Teniendo como premisa que el desarrollo del software es un proceso complejo y muchas veces se torna engorroso llegar a satisfacer las necesidades de los clientes. Debido a que con la selección de la herramienta generadora de reportes no era suficiente, se tomaron tareas y actividades que facilitaran la construcción y desarrollo del Módulo de Reportes, apoyados en herramientas y procedimientos sobre los que se apoya la Ingeniería de Software.

La importancia de la Ingeniería de Software (ISW) como ciencia viene dado al permitir analizar, diseñar, programar y aplicar un software de manera correcta y organizada, cumpliendo con todas las especificaciones del cliente y el usuario final. Como beneficios, permite mejorar la calidad de los productos de software, aumentar la productividad y trabajo de los ingenieros del software, facilitar el control del proceso de desarrollo de software y suministrar a los desarrolladores las bases para construir software de alta calidad en una forma eficiente. Seguidamente se definirán los principales elementos tomados de la Ingeniería de Software.

1.4. Ingeniería de software

Tomando la definición brindada por Roger Pressman sobre la ISW, la intención de la Ingeniería de Software es proporcionar un marco general para construir software con una calidad mucho mayor. Este autor plantea un grupo de disciplinas para desarrollar las actividades de la ingeniería de software, como son: Ingeniería de Requisitos, Modelado del Análisis, Ingeniería del Diseño, Desarrollo, Pruebas. También Pressman plantea la utilización de un modelo de desarrollo de software como conjunto distinto de actividades, acciones, tareas, fundamentos, y productos de trabajo para desarrollar software de alta calidad. (S. Pressman, 2007)

En la actualidad son disímiles los modelos existentes y cada uno brinda aspectos específicos al desarrollo de software. Estos modelos no son excluyentes por lo que se pueden utilizar en conjunto para el desarrollo de grandes proyectos. En esta investigación se utilizará el modelo de desarrollo establecido en la UCI (Universidad de las Ciencias Informáticas), en el siguiente epígrafe se analizarán sus características.

1.4.1. Modelo de desarrollo de software: Programa de Mejora

Como parte del proceso de reestructuración de la producción en la universidad, quedó establecido el Programa de Mejora como conclusión de la certificación del nivel II del modelo CMMI, siendo este el nivel básico para la gestión de proyectos de software. Estos modelos son conjuntos de buenas prácticas que ayudan a las organizaciones a mejorar sus procesos y proporciona un conjunto completo e integrado de guías para desarrollar productos y servicios estableciendo un Programa de Mejora. Transformando de procesos de desarrollo a procesos institucionalizados (guiada por procesos).

Permitiendo de esta manera que los procesos de la organización sean definidos y administrados, los recursos sean los adecuados, las responsabilidades sean asignadas, la dirección tenga visibilidad objetiva de los procesos y que se definan las actividades básicas para la administración de los proyectos. (CALISOFT, 2010)

En consecuencia, el centro CEGEL y por correspondiente el proyecto SIRECC, utiliza este modelo de desarrollo que detalla el ciclo de vida de sus proyectos asintiendo en los procesos y subprocesos

dictados por el Nivel II de Modelo de Capacidad de Madurez del Software (CMMI, por sus siglas en inglés), este programa de mejora explota las características fundamentales de RUP (Rational Unified Process, por su traducción al español Proceso Unificado de Rational): dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental. Las fases, actividades, recursos y componentes a desarrollar son reflejadas en el Proyecto Técnico del proyecto SIRECC y pueden ser consultadas en (Prieto Álvarez, 2012).

Para el desarrollo de la aplicación se toma como referencia el ciclo de vida del proyecto definido como parte del Programa de Mejora, incluyéndose las fases e iteraciones que se describen a continuación: Estudio Preliminar, Modelación del negocio, Levantamiento de Requisitos, Análisis y Diseño, Implementación y Pruebas. Cada una de las fases dictadas por el Programa de Mejora corresponde a las descritas por Pressman en (S. Pressman, 2007) y su concepto de Ingeniería de Software.

Dentro de las características de este modelo de desarrollo se encuentra que: brinda la posibilidad de que el Proyecto SIRECC cuente con un modelo estandarizado que incluye las fases por las que se debe transitar y el conjunto de artefactos a generar en cada una de ellas. Una de las utilidades a explotar es que es orientado a componentes debido a que se desarrolla teniendo en cuenta la estructura de agrupar por componentes, es adaptable al cambio y a las características específicas de cada proyecto; siendo un modelo iterativo incremental que además está centrado en la arquitectura. Todas estas características implementadas a la vez por el Marco de Trabajo a utilizar en el proyecto.

Como parte de este programa se estructura un expediente de proyecto, como producto de un estudio realizado por especialistas destacados en cada una de las esferas del desarrollo de software, abarcando con este expediente un grupo de artefactos necesarios para las buenas prácticas de desarrollo de software. Por lo que quedaron plasmadas en él las actividades que eran consideradas necesarias hasta ese momento, constatado esto en la entrevista realizada a la especialista Ing. Dagmay Aveleira Quiñonez (Ver Anexo 2).

Se obtendrá un calendario y presupuesto predecibles en el proyecto, se podrá mejorar el ciclo de vida dentro del desarrollo de software, se elevará productividad, calidad en los productos y servicios que ofrece la universidad a sus clientes. Reorganización de la producción, disciplina, niveles de chequeo,

informes y reportes a la alta gerencia, poder evaluar cuáles son los estados en el proyecto, cuales son los hitos, las tareas que se han realizado y cuáles no, especificación de requisitos formales y un grupo de elementos que disciplinan el proceso de desarrollo de software, son otras de las ventajas que se obtendrán, constatado en la entrevista realizada a la directora de CALISOFT, Aylin Febles. La entrevista realizada puede ser consultada en (Febles, 2011).

Dentro de las fases definidas en este modelo de desarrollo se utilizarán herramientas para el modelado del sistema, aunque estas herramientas sirven de apoyo en todo el ciclo de desarrollo del software. Las herramientas CASE (Computer Aided Software Engineering, por su traducción al español, Ingeniería de Software Asistida por Computadora) se utilizarán para aumentar la productividad en el desarrollo de software reduciendo el costo en términos de tiempo y de dinero. Seguidamente, la caracterización de las actividades a desarrollar en la segunda y tercera fase del Programa de Mejora.

1.4.2. Modelado del Negocio y Requisitos

El Modelado del Negocio es la fase destinada a comprender los procesos de negocio de una organización. Se comprende cómo funciona el negocio que se desea automatizar para tener garantías de que el software desarrollado va a cumplir su propósito. Para la descripción y modelado de negocio pueden ser utilizadas diferentes técnicas como el Modelado de Casos de Uso del Negocio y Notación de Modelado de Procesos de Negocio¹. (Programa de Mejora, 2011)

Posteriormente se comenzó con el levantamiento de requisitos. El esfuerzo principal en la fase es desarrollar un modelo del sistema que se va a construir. Incluye un conjunto de casos de uso, servicios que describen todas las interacciones que tendrán los usuarios con el software, estos responden a los requisitos funcionales del sistema.

En esta fase se realiza la especificación de requisitos, que incluye requisitos funcionales y no funcionales. Esta especificación es un documento que, clara y precisamente, describe cada uno de los requisitos esenciales del software (funciones, desempeño, restricciones de diseño y atributos de calidad) y de sus interfaces externas. (Programa de Mejora, 2011)

¹ Business Process Modeling Notation (BPMN)

Para el levantamiento de requisitos son utilizadas técnicas como las entrevistas y cuestionarios. Luego estos son validados con técnicas de validación de requerimientos como las Revisiones Técnicas Formales y Prototipos.

1.4.3. Análisis, Diseño e Implementación

Durante esta fase se modelará el sistema y su forma (incluida su arquitectura) para que soporte todos los requisitos, incluyendo los requisitos no funcionales. Esto contribuirá a una arquitectura sólida y estable que se convierta en un plano para la implementación. Los modelos a desarrollar en esta etapa son más formales y específicos de una implementación. Durante esta fase son desarrollados, diagramas de clases, diagramas de entidad relación, entre otros. (Programa de Mejora, 2011)

Con las salidas obtenidas de la fase de Análisis y Diseño se comenzará la implementación, donde a partir de los resultados que obtenidos anteriormente se implementará el sistema en términos de componentes, ficheros de código fuente, scripts, ejecutables y similares. Con el objetivo de certificar estas fases se realizará una validación de los resultados obtenidos con el uso de métricas. Seguidamente se explica el funcionamiento de la métrica Tamaño Operacional de la Clase (TOC).

Validación del Análisis, Diseño e Implementación

Para cuantificar la complejidad de implementación se estableció como indicador la Complejidad de Implementación de las clases según la cantidad de métodos y atributos de las clases. Utilizando como unidad de medida el cálculo del Tamaño Operacional de la Clase como métrica para el modelado del diseño, para su posterior clasificación en baja, media y alta.

Según (S. Pressman, 2007) las métricas para el modelado del diseño cuantifican los atributos de diseño de manera tal que le permiten al ingeniero de software evaluar la calidad del diseño. Estas métricas de modelado del diseño incluyen: métricas arquitectónicas, métricas al nivel de componentes, métricas de diseño de la interfaz y métricas especializadas en diseño orientado a objetos.

El TOC es una métrica especializada en diseño orientado a objetos, midiendo características de clases, además de las correspondientes a comunicación y colaboración. Está dado por el número de métodos y atributos asignados a una clase para evaluar los siguientes atributos de calidad:

- ✓ **Responsabilidad:** Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
- ✓ **Complejidad de implementación:** Un aumento del TOC implica un aumento de la complejidad de implementación de la clase.
- ✓ **Reutilización:** Un aumento del TOC implica una disminución del grado de reutilización de la clase.

Grandes valores de TOC representa gran responsabilidad de la clase. Esto implica la reducción de la reutilización de la clase y complica la implementación y las pruebas. De forma general, operaciones y atributos deben ser ponderados al determinar el tamaño de la clase. Para valores pequeños de TOC para una clase existe mayor posibilidad de que la clase pueda ser reutilizada (S. Pressman, 2007).

La importancia y facilidad de aplicación de esta métrica es: que las clases constituyen la unidad básica y fundamental de un sistema orientado a objetos. Por tanto, y dado el caso de que este software se realizó bajo la POO, es indiscutible que la validación se centró en la aplicación de métricas dirigidas a sus clases de forma individual, sus jerarquías y colaboraciones, haciendo uso de los atributos de calidad descritos anteriormente. (Lorenz, y otros, 1994)

Las medidas del cálculo del dominio del TOC pueden ser observadas en (Ver Anexo 4).

1.4.4. Pruebas Internas y Pruebas de Liberación

Durante la fase de Pruebas Internas el proyecto verifica el resultado de la implementación probando según sea necesario cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones finales a ser liberadas. Durante esta fase se deben desarrollar artefactos de prueba como: Diseños de casos de prueba, Listas de chequeo y de ser posibles componentes de prueba ejecutables para automatizar las pruebas. (Programa de Mejora, 2011)

Las Pruebas de Liberación son diseñadas e implementadas por el Laboratorio Industrial de Pruebas de Software a todos los entregables de los proyectos antes de ser entregados. (Programa de Mejora, 2011)

En la actualidad CALISOFT realiza las pruebas clasificadas en: Basadas en Riesgos, de Confiabilidad, Eficiencia, Estáticas, Funcionalidad, Mantenibilidad (en desarrollo), Portabilidad y Usabilidad. De las anteriormente mencionadas será aplicadas al Módulo de Reportes las Pruebas de Funcionalidad y Unitarias, estas últimas pertenecientes al grupo de Pruebas Estáticas.

Durante todas las fases del modelo de desarrollo se utilizaron herramientas y tecnologías como apoyo al desarrollo de la propuesta de solución. Seguidamente se muestra el estudio y selección de estas herramientas.

1.5. Herramientas CASE

Las herramientas CASE comprenden un amplio grupo de programas que se utilizan para ayudar a las actividades del proceso del software, como el análisis de requerimientos, el modelado de sistemas, la depuración y las pruebas (Sommerville, 2005).

Dentro de ellas, el modelado del sistema es una de las actividades más importantes del proceso de software y su función es permitir la comprensión sobre las necesidades de conocimiento, lo que se necesita obtener y cómo debe quedar el producto. Es definido también en (Sommerville, 2005) como modelo de un conjunto de componentes y de relaciones entre estos componentes. En este nivel de detalle, el sistema se descompone en un conjunto de subsistemas que interactúan y deben ser representados de forma similar hasta que el sistema esté dividido en componentes funcionales.

Para realizar el modelado del sistema existen diversos lenguajes como UML (Unified Modeling Language o por su traducción al español Lenguajes de Modelado Unificado), estos se encargan del diseño de modelos. En el proyecto técnico presentado en (Jorrín Perdomo, 2013), quedó establecido el lenguaje de modelado UML para el modelado del sistema, en consecuencia de ser un lenguaje entendible y dominado por todos los recursos humanos del proyecto. En el epígrafe que a continuación aparece se detallan algunas de las características de este lenguaje de modelado.

1.5.1. UML 2.0

UML está definido como un lenguaje para visualizar y documentar los artefactos de un sistema de software orientado a objetos. Permite transformar en un lenguaje estándar los componentes del proceso de desarrollo de aplicaciones. Tiene como objetivo principal, entregar un material de apoyo que permita definir diagramas propios, como también poder entender el modelamiento de diagramas ya existentes. Es un lenguaje estándar de modelado para software, para la visualización, especificación, construcción y documentación de los artefactos de sistemas en los que el software juega un papel importante. Permite a los desarrolladores visualizar los resultados de su trabajo en esquemas o diagramas estandarizados. (Jacobson, y otros, 2000)

Para facilitar el uso de este lenguaje de modelado fue establecida la herramienta Visual Paradigm por parte del proyecto SIRECC debido a las facilidades de uso y la amplia gama de utilidades que brinda, a continuación una descripción de esta herramienta.

1.5.2. Visual Paradigm for UML 8.0 Enterprise Edition

Es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software. Permite representar diagramas de clases, generar código desde diagramas y generar documentación. Tiene soporte multiplataforma y proporciona excelentes facilidades de interoperabilidad con otras aplicaciones. Tiene apoyo adicional en cuanto a generación de artefactos automáticamente y generación de documentación, ya que brinda la posibilidad de documentar todo el trabajo sin necesidad de utilizar herramientas externas. (Visual Paradigm, 2013)

Dentro de sus características principales cuenta con, diagramas de Procesos de Negocio y Modelado colaborativo con el repositorio Subversion (nueva característica). Generación de bases de datos y transformación de diagramas de Entidad-Relación en tablas de base de datos, así como un generador de informes para generación de documentación.

La selección de esta herramienta viene dado por las especificidades del modelo de desarrollo, donde es propuesta para el trabajo del Análisis y Diseño.

Luego del uso de esta herramienta CASE para la representación visual de los principales componentes del desarrollo del software se dio paso a la selección de las tecnologías de desarrollo, estableciendo el uso de un Marco de Trabajo, la plataforma de desarrollo, etc. A continuación se ofrecen las caracterizaciones de cada una.

1.6. Tecnologías

La solución de software se desarrollará sobre el Marco de Trabajo GEFORT, implementado en plataforma Java Enterprise Edition (por su traducción al español Edición Empresarial Java), basando la arquitectura en dos características fundamentales. La primera es que toda la organización estará soportada en un modelo de desarrollo basado en procesos. La segunda se refiere a que todo el ciclo de elaboración del sistema se ajustará a un modelo iterativo e incremental. El uso de este Marco de Trabajo, permitirá manejar correctamente el proyecto haciendo especial énfasis en la generación de artefactos bien documentados que faciliten la capacitación y transferencia del producto. (Jorrín Perdomo, y otros, 2011)

1.6.1. Marco de Trabajo GEFORT

El término “Marco de Trabajo” se refiere a una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Se puede considerar como una aplicación genérica incompleta y configurable a la que puede añadirse las últimas piezas para construir una aplicación concreta. (Carrasco Puebla, y otros, 2009)

Para el proyecto SIRECC fue establecido el Marco de Trabajo GEFORT desarrollado anteriormente para el Sistema de Gestión de Antecedentes Penales de la República Bolivariana de Venezuela. Por las similitudes entre ambos sistemas y los componentes de configuración que posee se considera que contribuirá a disminuir el tiempo de desarrollo de SIRECC. Para comprender el diseño lógico del Marco de Trabajo (Ver Anexo 7).

1.6.2. Servidor de Aplicaciones GlassFish 3.1.2

Un servidor de aplicaciones es una tecnología que proporciona la infraestructura y servicios clave a las aplicaciones alojadas en un sistema. Entre los servicios habituales de un servidor de aplicaciones se incluyen los siguientes: agrupación de recursos (por ejemplo, agrupación de conexiones de base de datos y agrupación de objetos), administración de transacciones distribuida, servicios de detección de errores y estado de las aplicaciones, así como seguridad integrada.

Glassfish es un servidor de aplicaciones que implementa las tecnologías definidas en la plataforma JavaEE y permite ejecutar aplicaciones que siguen esta especificación. Entre las características más notables de GlassFish se puede destacar su velocidad, alta escalabilidad, manejo centralizado de clúster e instancias, bajo consumo de memoria, interoperabilidad y un excelente panel de administración. (ORACLE, 2013)

1.6.3. Sistema Gestor de Bases de Datos PostgreSQL 9.3

PostgreSQL es el gestor de bases de datos de código abierto más avanzado hoy en día, ofreciendo control de concurrencia multi-versión, soportando casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, y tipos y funciones definidas por el usuario), contando también con un amplio conjunto de enlaces con lenguajes de programación incluyendo Java. La necesidad de almacenar grandes volúmenes de información referente a los procesos, así como las unidades documentales asociadas, que se ejecutan en la Cámara de Comercio de la República de Cuba requiere el uso de gestores de bases de datos que permitan gestionar de manera ágil la información necesaria. La selección debe estar acorde con las políticas que se establecen en el Programa de Mejora, por lo que se utiliza PostgreSQL como gestor de base de datos Open Source. Útil cuando se trata de la gestión de grandes volúmenes de datos. Además es usado en las soluciones informáticas que se desarrollan en la Universidad de las Ciencias Informáticas para las instituciones del estado, lo que facilita la transferencia y la mantenibilidad por parte de los clientes.

Cuenta, dentro de sus principales características con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario. Su administración se basa en usuarios y privilegios y es altamente confiable en cuanto a estabilidad se

refiere. Sus ventajas se resumen en que como gestor de bases de datos es una herramienta código abierto, multiplataforma y sin restricciones de uso, requisitos solicitados por el cliente para el desarrollo de la aplicación. Además la universidad cuenta con la información necesaria gracias a la existencia de una comunidad de PostgreSQL, que realiza actividades conjuntamente con la comunidad nacional e internacional del gestor de base de datos. (The PostgreSQL Global Development Group, 1996)

1.6.4. Entorno de desarrollo integrado

Un entorno de desarrollo integrado (en inglés Integrated Development Environment o IDE, por sus siglas en igual idioma) es un programa compuesto por una serie de herramientas que utilizan los programadores para desarrollar código. Esta herramienta puede estar pensada para su utilización con un único lenguaje de programación o bien puede dar cabida a varios de estos. (Oracle Corporation, 2013)

Para la implementación de la Solución de Software se utilizará NetBeans 7.2 como IDE de desarrollo, con un conjunto de complementos que amplían sus funcionalidades como ambiente de desarrollo integrado de aplicaciones de escritorio. Los complementos estarán en correspondencia con el Marco de Trabajo que se utilizará. A continuación se abordará acerca de la herramienta utilizada en el proyecto SIRECC.

NetBeans 7.2

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. Es un producto libre y gratuito sin restricciones de uso. Permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos.

Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software. (Oracle Corporation, 2013)

La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están: administración de las interfaces de usuario (ej. menús y barras de herramientas), administración de las configuraciones del usuario y administración del almacenamiento. (Oracle Corporation, 2013)

Luego de seleccionadas las herramientas a trabajar para el desarrollo del Módulo de Reportes y todo su proceso de desarrollo, se dio paso a la selección de las métricas para llevar a cabo el proceso de validación de las variables del problema en cuestión. Evaluando con estos resultados la validación de la propuesta de solución. Seguidamente, la teoría referente a estas métricas.

1.7. Validación de las variables

Para la validación de las variables de la investigación, se estudiarán las teorías relacionadas a la medición y evaluación de la variable dependiente y la variable independiente.

1.7.1. Variable independiente

Alcance

La revista especializada en artículos científicos Líder de Proyecto en (Project Management Institute, 2012) describe el alcance del producto como características y funciones que caracterizan a un producto o servicio. Éstas incluyen tanto características de tipo técnico como características del producto relacionadas con el plazo de terminación y coste de producto final.

Por lo que se medirá el Alcance del Módulo de Reporte como las características y funciones que definen un producto, servicio o resultado. En esta investigación se estableció para la dimensión Alcance, el indicador: cumplimiento de los requisitos terminados y como unidad de medida se realizará una valoración del índice de requisitos terminados. Para la consulta de información sobre los requerimientos tanto funcionales como no funcionales puede consultar en este documento el Capítulo 2: Solución de Software, epígrafe 2.3. El cálculo se establecerá mediante la fórmula:

Índice de requisitos implementados = (Cantidad de requisitos implementados) / (Cantidad total de requisitos)

Calidad

Múltiples han sido los conceptos brindados sobre la calidad del software. Uno de ellos es el planteado por modelo CMMI, que presenta la calidad de un producto o de un sistema es en su mayor parte consecuencia de la calidad de los procesos empleados en su desarrollo y mantenimiento. El aseguramiento de la calidad es un medio planeado y sistemático para asegurarle a la dirección que los estándares, prácticas, procedimientos, y métodos de un proceso se aplican correctamente. (CALISOFT, 2011)

Dentro de las áreas de procesos del nivel 2 se encuentra el Aseguramiento de la Calidad al Proceso y al Producto (PPQA, por sus siglas en inglés). El propósito del Aseguramiento de la calidad del proceso y el producto es proporcionar a los equipos de trabajo y a la dirección visibilidad objetiva de los procesos y de sus productos asociados. Esto se logra mediante las políticas de PPQA como evaluar objetivamente adherencia a procesos y productos de trabajo, registrar y comunicar los resultados de las evaluaciones y el análisis de tendencias, así como asegurar la resolución de las no conformidades detectadas. (CALISOFT, 2011)

Por lo antes expuesto se decidió caracterizar la dimensión Calidad Externa e Interna por el modelo de basamento de CMMI en cuanto a calidad, propuesto por la ISO. Los indicadores a tener en cuenta: Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Facilidad de Mantenimiento, Portabilidad, y como unidad de medida el Cálculo del valor de calidad. Estos datos serán los resultados de las pruebas de software realizadas al Módulo de Reportes. (ISO, 2006)

La Calidad Interna será medida mediante las Pruebas de Caja Blanca y Caja Negra. La Prueba de Caja Blanca a utilizar es: Prueba del Camino Básico, mientras que como Prueba de Caja Negra se realizarán Pruebas Internas mediante la elaboración de Casos de prueba y su posterior aplicación.

Como indicador de Calidad Externa se utilizarán las medidas establecidas por la norma ISO, esto permitirá obtener una evaluación por parte de los clientes en cuanto a calidad del producto. Para consultar estos criterios de evaluación, puede consultar (Anexo 5).

1.7.2. Variable dependiente

Para cuantificar la variable: Tiempo en la obtención de reportes, se obtuvo como dimensión el Tiempo y como indicador de esta dimensión el Tiempo de generación de reportes. Con el objetivo de realizar una correcta medición se estableció dos unidades de medida:

- ✓ Tiempo de generación de los documentos de salida < 2 segundos
- ✓ Tiempo de generación de las tablas del sistema < 3 segundos

Estos fueron establecidos por la necesidad de la obtención rápida de reportes de información, evidenciados en los requisitos no funcionales como el RnF-04 que plantea que el tiempo de respuesta brindado por el sistema será menor de 3 segundos. Para ello se medirá el tiempo de salida de cada uno de los reportes en un ambiente de trabajo real.

1.8. Conclusiones del capítulo

Se realizó un análisis de los sistemas generadores de reportes a nivel internacional y nacional, obteniendo como resultado la propuesta de creación de un Módulo de Reportes con la utilización de Jasper Report. Se analizaron herramientas, tecnologías y lenguajes propuestos por políticas del Centro de Gobierno Electrónico (CEGEL) de la universidad, utilizando para este trabajo el modelo de desarrollo Programa de Mejora y el Marco de Trabajo GEFORT, como herramienta CASE Visual Paradigm 8.0 y como lenguaje de modelado UML. Como servidor de bases de datos PostgreSQL en su versión 9.3, el entorno de desarrollo integrado NetBeans 7.2 y servidor de aplicaciones GlassFish 3.1.2.

2 SOLUCIÓN DE SOFTWARE

“El socialismo va a ser muy difícil de construir plenamente sin la computación”

Fidel Castro Ruz

2.1. Introducción

Debido a las dificultades existentes para realizar los reportes en la Cámara de Comercio, se toma la decisión de desarrollar un Módulo de Reportes que permita solucionar los inconvenientes de la gestión de informes y documentos generados en los procesos de negocio. Este módulo posteriormente sería incorporado a la aplicación SIRECC. La generación de reportes, es una operación que surge como consecuencia directa de la necesidad de interrelación y operacionalización de los procesos de negocio entre las áreas de la Cámara. Esta operación pretende resolver los problemas entre los formatos de los documentos guardados en la Cámara, la accesibilidad a la información generada de estos informes y los estándares presentados en su manual de identidad. En este capítulo se describe la propuesta de solución del módulo según las fases propuestas por el Programa de Mejora y las caracterizaciones de su utilización en el desarrollo del sistema.

2.2. Modelación del negocio

Esta fase estuvo destinada a comprender los procesos de negocio de la organización. Permitted la visión del funcionamiento de las actividades de negocio de la Cámara de Comercio que posteriormente se informatizarían para tener garantías de que el software desarrollado daría cumplimiento a su propósito. Se realizó el modelado del negocio mediante el diagrama de Modelo de Dominio, donde quedaron establecidas las principales definiciones a tratar. El modelo de dominio sirvió como mapa explicativo para esclarecer los conceptos en el dominio del problema y los atributos pertenecientes a cada uno de los conceptos.

La finalidad de esta actividad fue estipular una especificación del dominio del problema y los requisitos desde la perspectiva de la clasificación por objetos y desde el punto de vista de entender los términos empleados en el dominio. Para ver el diagrama Modelo de Dominio (Ver Anexo 11).

Las salidas de la Modelación del Negocio se convirtieron en entradas a la fase de Levantamiento de Requerimientos, facilitando las actividades correspondientes a esta fase. A continuación se ofrece una descripción de las actividades realizadas en el levantamiento de los requerimientos.

2.3. Levantamiento de Requisitos

En correspondencia a lo planteado por el Plan de Mejora en el Libro de Procesos para la Administración de Requisitos, se generaron los artefactos correspondientes al modelo de desarrollo.

Como parte de las actividades realizadas en el Modelado del Sistema se generó el artefacto: Salidas del Sistema. Este documento es especificado por el programa de mejora para obtener las salidas de la aplicación y los reportes correspondientes a cada uno de los procesos de negocio, permitiendo el nivel de detalle necesario en estos documentos.

Se realizó el levantamiento de requisitos, tanto funcionales como no funcionales, generándose el artefacto Especificación de Requisitos del Software para el Módulo de Reportes, que constituye un elemento clave para el diseño y la posterior implementación. Este artefacto se caracteriza por contener para cada uno de los requerimientos el nombre, una descripción, establece la complejidad y prioridad para el cliente, facilitando el orden y criticidad de implementación. En esta plantilla además se establecen los campos necesarios, tipo de datos y regla o restricciones.

Luego de realizada la Especificación de Requisitos se dio paso a la Descripción de Requisitos del Software para el Módulo de Reportes. Caracterizado fundamentalmente por presentar los requisitos de forma completa, definiéndose todas las responsabilidades del módulo. Este artefacto presenta una adecuada organización y documentación donde los términos, las tablas y los prototipos están correctamente descritos y referenciados. Cada requisito que comprende, tiene una única interpretación evitando la ambigüedad en las definiciones y funcionalidades.

Según lo referido anteriormente, la descripción de requisitos puede ser tomada como artefacto de entrada al diseño para facilitar la comprensión del desarrollo del módulo. A continuación se presentan los requisitos funcionales y no funcionales del Módulo de Reportes.

2.3.1. Requisitos funcionales

Para realizar el proceso de captura de los requisitos, se hizo necesaria la utilización de algunas técnicas que sirvieron para verificar la veracidad de los objetivos propuestos para el desarrollo de este módulo, permitiendo que dieran satisfacción a las necesidades de los clientes. Dentro de las técnicas utilizadas se encuentran: las entrevistas, aplicada a los especialistas de la Cámara que fueron considerados proveedores válidos para los cuales va a ser desarrollado el componente. Estas entrevistas posibilitaron los debates, donde cada cliente (especialista de cada área) explicó de forma clara sus necesidades.

Listado de los requisitos funcionales (RF):

- RF-1 Generar documento Argumentación
- RF-2 Generar documento Criterios de la junta ejecutiva sobre admisión como asociado
- RF-3 Generar documento Carta al Presidente
- RF-4 Generar documento Certificación
- RF-5 Visualizar reporte de asamblea general
- RF-6 Visualizar reporte de cantidad de empresas por provincia
- RF-7 Visualizar reporte de cantidad de empresas por sectores
- RF-8 Visualizar reporte de cantidad de empresas por secciones
- RF-9 Visualizar reporte de cuotas de asociados activos
- RF-10 Visualizar reporte de datos de sitio web
- RF-11 Visualizar reporte de empresas para invitaciones de FIHAV
- RF-12 Generar observaciones
- RF-13 Generar licencia de inscripción
- RF-14 Mostrar reporte de licencias a renovar
- RF-15 Mostrar reporte por clasificadores
- RF-16 Mostrar reporte por actividad autorizada

- RF-17 Mostrar reporte por objeto social
- RF-18 Mostrar reporte de pagos
- RF-19 Generar licencia de renovación
- RF-20 Generar anexos
- RF-21 Generar licencia de agencia de viajes nacional
- RF-22 Generar licencia de agencia de viajes extranjera
- RF-23 Generar licencia de agencia de viajes extranjera representada
- RF-24 Generar licencia de renovación de agencia de viajes nacional
- RF-25 Generar licencia de renovación de agencia de viajes extranjera
- RF-26 Generar licencia de renovación de agencia de viajes extranjera representada
- RF-27 Visualizar inscripciones de agencias de viajes nacionales y extranjeras próximas a renovar
- RF-28 Visualizar inscripciones de agencias de viajes representadas próximas a renovar
- RF-29 Generar certificado de inscripción de Importadora y Exportadora
- RF-30 Generar certificado de ratificación o modificación de facultades
- RF-31 Generar certificado de cancelación de las facultades
- RF-32 Generar certificación a tercero de existencia
- RF-33 Generar certificación a terceros de no existencia
- RF-34 Visualizar empresas por diferentes parámetros
- RF-35 Visualizar empresas por actividad económica
- RF-36 Generar planilla AO-3 para solicitud de entrada de extranjeros
- RF-37 Generar planilla CEC-4
- RF-38 Generar planilla Registro de Trámites
- RF-39 Generar planilla Permisos de Trabajo
- RF-40 Generar Acta
- RF-41 Generar Anexo 1
- RF-42 Adicionar documento generado al expediente
- RF-43 Visualizar documento
- RF-44 Exportar documento

Para el desarrollo de la solución final del componente se tuvo en cuenta agregar los siguientes requisitos funcionales de cara a la aplicación:

RF-45 Definir fuente de datos (del inglés data-source)

RF-46 Buscar información en forma dinámica

RF-47 Crear tablas según criterios de búsquedas

RF-48 Desarrollar Gestor de Acceso a Datos

RF-49 Desarrollar Enumerador de variables estáticas

RF-50 Generar cuerpo de reporte

Aunque no fueron solicitados por el cliente, se consideraron importantes ya que el requisito RF-39 permitirá la generación y configuración de reportes a partir de plantillas Jasper y mapas de configuración. Por otra parte el requisito RF-40 ejecuta una búsqueda completa de la información en forma dinámica mediante los parámetros que desee el cliente como nombre de la empresa, etc. Con el objetivo de crear reportes que incluyan tablas con información según criterios de búsquedas (ejemplo: cantidades) y facilitar los datos como listas se realizó el RF-41. En cuanto a RF-42 es usado para el acceso a la información de la base de datos del sistema y mantener la consistencia de los datos a extraer, por último y para enumerar variables de contenido estático se implementó el RF-43. Una parte importante de estos reportes es el cuerpo del documento que contendrá las variables de información el cuerpo de texto, por lo que se decidió implementar el RF-44.

A continuación se muestra un resumen de la descripción textual del requisito *Generar documento de Criterios de la Junta ejecutiva sobre admisión como asociado*, su prototipo de interfaz de usuario funcional se puede consultar en el Anexo 13. En los documentos Descripción de Requisitos de Software y Especificación de Requisitos de Software, se muestran las especificaciones y descripciones de todos los requisitos funcionales del Módulo de Reportes, ubicados en (González Hernández, y otros, 2014).

Precondiciones	Debe existir el expediente de la empresa creado y la documentación de la entidad debe haber sido revisada.	
Flujo de eventos		
Flujo básico		
1.	El sistema permitirá generar un documento de solicitud de admisión como asociado.	
2.	Se selecciona el documento, del cual se quieren mostrar los datos.	
3.	El sistema muestra los documentos con los siguientes datos: <ul style="list-style-type: none"> • Nombre del Documento. • Fecha de Emisión • Destinatario • Remitente • Nombre de la empresa • Organismo al que se subordina. • Objeto Social. • Ingreso Anual. • Cuota. 	
4.	Concluye el requisito.	
Pos-condiciones		
1.	Se mostró el documento Criterio de la Junta Ejecutiva sobre solicitud de admisión como Asociado.	
Conceptos	Documento	Visibles en la interfaz: <ul style="list-style-type: none"> • Todos (Ver paso 3)

Tabla 2: Resumen de la descripción del requisito funcional Generar documento de Criterios de la Junta ejecutiva sobre admisión como asociado. (RF-2)

2.3.2. Requisitos no funcionales

Los requerimientos no funcionales cumplen con las especificaciones establecidas en el documento Especificación de Requisitos de Software NF del Módulo de Reportes. Este documento puede ser consultado en (González Hernández, y otros, 2014). Seguidamente, listado de los requisitos NF:

Usabilidad

- ✓ Tipo de usuario final: RnF-01 El sistema podrá ser utilizado por los usuarios que se describen a continuación: Director de Formación y Membrecía, Director Jurídico, Asesor Jurídico, Especialista en Desarrollo Empresarial, Junta Ejecutiva, Abogado.
- ✓ Ambiente: RnF-02. El sistema deberá presentar una interfaz de usuario fácil de entender y usar.

RnF-03 Agrupar vínculos y botones por grupos funcionales. La consistencia de la interacción entre usuario y sistema estará determinada por el diseño de la interfaz de usuario que mantendrá los elementos como menús, banners y zona de trabajo, en posiciones fijas, además de la mayor uniformidad posible entre cuadros de texto y botones.

El sistema debe ser de uso intuitivo, de tal forma que se reduzca los tiempos de entrenamiento, soporte y prueba por parte del usuario. La agrupación de los botones por funcionalidad determinará además la capacidad de componer la interfaz de acuerdo a las funciones requeridas por un rol determinado.

RnF-04 El tiempo de respuesta brindado por el sistema será menor de 3 segundos. Teniendo en cuenta el nivel de concurrencia que pueda existir, debe ser capaz de prestar servicio sin que se deterioren los tiempos de respuestas en función de los requisitos de hardware necesarios.

Disponibilidad

RnF-05 El tiempo máximo de inactividad es de 5 min., y una vez que el sistema quede inactivo el usuario deberá autenticarse nuevamente y el sistema mantendrá el estado que tenía antes de pasar a la inactividad.

RnF-06 El sistema deberá estar disponible las 24 horas de los días laborales para su funcionamiento y realización de las salvas que se deben ejecutar después de la jornada laboral (8 horas).

Eficiencia

RnF-07 El sistema debe ser capaz de dar respuestas a las peticiones con un nivel aceptable de desempeño. Teniendo en cuenta el nivel de concurrencia que pueda existir, debe ser capaz de prestar servicio sin que se deterioren los tiempos de respuestas.

RnF-08 El sistema debe garantizar el acceso concurrente para todos los usuarios del sistema durante la jornada laboral establecida.

Restricciones de diseño

RnF-09 Para el montaje del sistema se requerirá del sistema gestor de bases de datos PostgreSQL 9.1 y del servidor de aplicaciones Glassfish 3.1.2.

RnF-10 Diseñar el sistema a través subsistemas: el sistema debe garantizar que cada subsistema tenga fronteras claramente definidas y funciones relacionadas. Cada módulo será independiente del resto.

Interfaz

RnF-11 El sistema presentará una interfaz legible, simple de usar e interactiva.

RnF-12 El sistema deberá tener un diseño de interfaz de usuario en funciones de los estándares definidos en la UCI para las Interfaces de usuario.

Estándares Aplicables

RnF-13 Se aplicarán los estándares de java en función de facilitar el mantenimiento de la aplicación.

Seguridad

RnF-14 El sistema podrá ser utilizado solamente por usuarios autenticados en el sistema.

RnF-15 El sistema brindará la posibilidad de establecer permisos sobre acciones, garantizando que solo acceda a la información quien esté autorizado.

RnF-16 El sistema mostrará las funcionalidades de acuerdo a quien esté autenticado en el sistema.

RnF-17 El sistema debe asegurar el almacenamiento de las credenciales de los usuarios utilizando algoritmos criptográficos que oculten la identidad verdadera de los usuarios.

RnF-18 El sistema debe permitir almacenar todas las acciones de los usuarios sobre el sistema como constancia de las acciones realizadas.

Al concluir el levantamiento de los requisitos, tanto funcionales como no funcionales, se aplicaron técnicas para validar los requisitos recolectados. Estos resultados alcanzados en la validación de los requerimientos pueden ser consultadas en el Capítulo 3: Análisis de los resultados, epígrafe 3.2 Validación de las fases de desarrollo, sub-epígrafe 3.2.1.

2.4. Análisis y Diseño

Durante esta fase se modeló el sistema y su forma (incluida la arquitectura) para que soporte todos los requisitos, incluyendo los no funcionales. Esto contribuyó a una arquitectura sólida y estable que se convierte en un plano para la implementación. Los modelos desarrollados en esta etapa son más formales y específicos de una implementación. Durante esta fase se desarrollaron: documento de arquitectura, diagramas de clases, diagramas de entidad relación, diagrama de despliegue entre otros.

2.4.1. Diseño Arquitectónico

La Arquitectura de Software de la Solución de Software, como módulo de aplicación de escritorio, será desarrollado sobre la plataforma Java Enterprise Edition (JEE), basando la arquitectura en dos características fundamentales. La primera es que toda la organización estará soportada en un modelo de desarrollo basado en procesos. La segunda se refiere a que todo el ciclo de elaboración del sistema se ajustará a un modelo iterativo e incremental.

Para el diseño arquitectónico se utilizará un estilo cliente-servidor con el objetivo de proporcionar un nivel de mantenibilidad adecuado para la solución. Este estilo permitirá la comunicación contractual entre la aplicación cliente y la aplicación servidor que permitirán dar solución a los requerimientos de los usuarios, abstrayéndolos de la distribución física del sistema. El objetivo fundamental de la aplicación cliente será la interacción entre la solución y los usuarios finales. Por su parte, el servidor contendrá la implementación de todas las reglas de negocio, además de la gestión de la información que se maneja, siendo su objetivo fundamental dar respuesta a las diferentes peticiones de la aplicación servidor.

Otro estilo a utilizar es el n-capas para organizar mejor el diseño teniendo en cuenta que permite establecer una organización jerárquica entre cada una de las capas, garantizando que cada capa proporcione servicios a la capa inmediatamente superior. Fueron identificadas 4 capas: Presentación, que contendrá los componentes de interfaz de usuario; Negocio del Cliente, que tendrá como responsabilidad fundamental llevar a cabo procesos de validación y gestión de la información entre la aplicación cliente y la aplicación servidor; Negocio del Servidor, que gestionará los componentes que implementarán la lógica de negocio del sistema; Acceso a Datos, que manejará la información entre la aplicación servidor y la base de datos; esta vista física de los estilos arquitectónicos empleados en la solución puede ser consultada en el Anexo 8.

Otro estilo empleado es el desarrollo basado en componentes, con el fin de reducir el ciclo de desarrollo y proporcionar un diseño más escalable y refinado. Este estilo permitirá reutilizar componentes desarrollados previamente en el Marco de Trabajo, sumándole un incremento en la mantenibilidad del sistema considerable, debido a que la actualización de estos será transparente para el sistema, siempre y cuando no se modifique la interfaz de interacción de estos con el resto del sistema.

2.4.2. Diseño de Clases

Los diagramas de clases permitieron obtener de forma estática la representación de los requisitos que se lleva a cabo a través de las clases del sistema y sus relaciones. Su principal utilidad radicó en mostrar a través de sus atributos y métodos la estructura de las clases que después serían escritas en el lenguaje de programación Java.

A continuación se muestra un fragmento de la descripción de las clases del diseño del proceso que se informatizó con el desarrollo del módulo, el diagrama de clases del Módulo de Reportes puede ser consultado en (Ver Anexo 12). Para visualizar la tabla completa de las descripciones de clases (Ver Anexo 6).

Descripción del diseño de clases del Módulo de Reportes y sus funcionalidades	
Nombre	Descripción
EDReporte	Contiene toda la información referente a los reportes como la plantilla Jasper, etc.
DSGenerico	Se encarga de la fabricación de los reportes.
FactoraEDReporte	Se encarga de convertir de entidad dominio a entidades persistentes y viceversa.
GestorADReporte	Permite el acceso a datos de reportes.
AcciónReporteArgumentación	Permite la creación en PDF del reporte Argumentación con los datos de la empresa y permite la visualización e impresión correspondiente.

Tabla 3: Resumen de descripción del diseño de clases del Módulo de Reportes.

2.4.3. Patrones de diseño empleados en la solución

SIRECC está implementado bajo el Marco de Trabajo de GEFORT. El diseño, a través del correcto uso de patrones de diseño en la generación de los artefactos necesarios para el desarrollo, facilitó crear una entrada apropiada como punto de partida a las actividades de implementación, con el fin de lograr una mayor calidad del producto y la satisfacción del cliente. La selección de los patrones de diseño estuvo basada en el Marco de Trabajo y se emplearon:

Patrones GRASP (Patrones de Principios Generales para Asignar Responsabilidades)

Experto: Este patrón se evidencia en la clase que cuenta con la información necesaria para cumplir las funcionalidades a realizar a partir de la información manejada dentro del componente. Es evidenciado en la clase DSGenerico (Data Source - Genérico), la cual es la encargada de generar los reportes y así obtener toda la lógica para cada una de estas funcionalidades.

Creador: Este patrón es evidenciado además en las clases Acciones (Ejemplo: AcciónReporteArgumentación) que son las encargadas de visualizar e imprimir el reporte en formato PDF, cada una especializada en su propio reporte.

Alta cohesión: GEFORT presenta entre sus características la organización del trabajo en cuanto a estructura y responsabilidades bien definidas, esto permite el trabajo de las clases con una alta cohesión. Evidenciándose en la clase DSGenerico, encargada solo de la conformación del reporte.

Bajo Acoplamiento: Mantiene las relaciones mínimas entre las clases, respetando sus relaciones y prioridades, de forma tal que las modificaciones posteriores que se deseen realizar no afecten el resto de las clases. Por lo que un cambio en el módulo no implicaría un impacto de relevancia.

Para observar la utilización de estos patrones GRASP consultar la Imagen 2: Diagrama de Componentes del Módulo de Reportes (RF-1 Generar Argumentación)

Patrones GOF (Gang-of-Four, por su traducción al español “pandilla de los cuatro”)

Factoría (Factory): Se encargó en convertir de entidades de dominio a entidades persistentes y viceversa, con el objetivo de ocultar detalles de cómo son creados los objetos.

Fachada (Facade): Permitió presentar una única interfaz responsable de colaborar con los componentes del subsistema.

2.4.4. Modelo de Datos

Para representar la estructura de la Base de Datos del Módulo de Reportes se realizó el Modelo de Datos, permitiendo la representación lógica de los datos procesados por el sistema. Este modelo fue construido utilizando la técnica de modelado de datos Entidad-Relación-Atributo (Modelado ERA), mostrando las entidades de datos, sus atributos asociados y las relaciones entre las entidades. La imagen que se muestra a continuación, refleja un fragmento del modelo de datos del Módulo de Reportes y representa las principales tablas que guardan información referente a los reportes y quienes pueden acceder a consultarlos y generarlos. Para consultar el Modelo de Datos General del Módulo de Reportes (Ver Anexo 15).

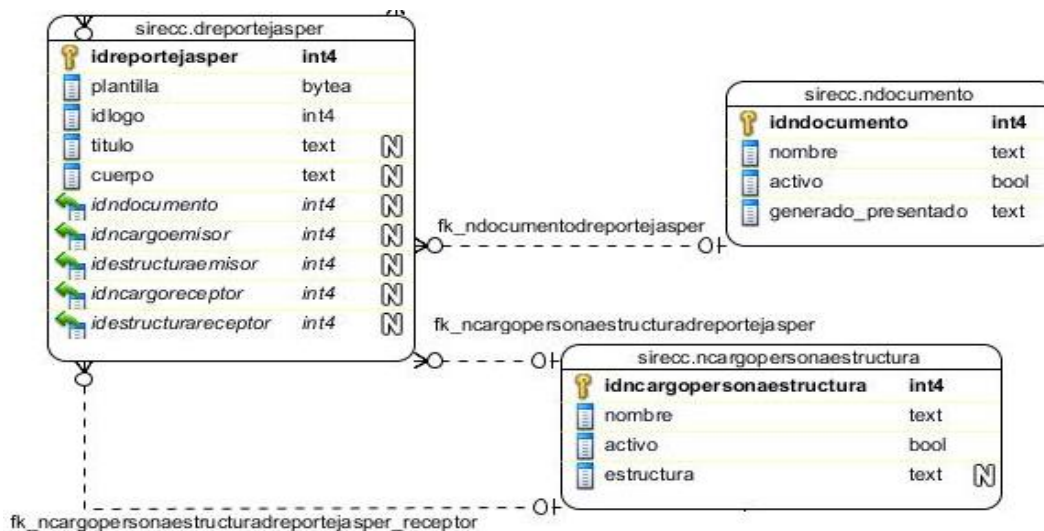


Imagen 1: Fragmento del Modelo de Datos del Módulo de Reportes

Las actividades de Análisis y Diseño fueron esenciales en el proceso de desarrollo de software. Se realizaron con el objetivo de permitir la implementación del sistema de manera precisa. La descripción de las actividades realizadas durante la implementación se presenta en el epígrafe que aparece a continuación.

2.5. Implementación

En la implementación y a partir de los resultados del análisis y diseño se implementó el sistema en términos de componentes, como: ficheros de código fuente, scripts, ejecutables y similares.

2.5.1. Diseño de Componentes

El diseño de componentes permitió traducir el modelo de diseño en un software operacional. En este trabajo de tesis el componente reporte perteneciente a SIRECC, se encargó de informatizar todo el proceso de generación de informes, documentos y tablas dinámicas. Este modelo de componentes representa los componentes, sus interfaces y las relaciones de los componentes con las interfaces que utilizan, es una definición de los estándares para el establecimiento de componentes, documentación y despliegue. Estas relaciones (Ver Anexo 14), fueron definidas para dar solución arquitectónica al componente entre los cuales se encuentran: Empresa Cubana, Sucursal Extranjera, Agencia de Viaje, Importador-Exportador, Trámite, Solicitante y el componente a desarrollar Reportes.

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

Empresa Cubana: Es el encargado de registrar los datos del contacto, datos del registro y la actividad económica que enmarca a la entidad en cuestión.

Sucursal Extranjera: Es el componente encargado de registrar los datos de contacto de la sucursal, datos del contacto en Cuba, de registro y de la actividad económica de la sucursal extranjera.

Agencia de Viaje: Es el componente encargado de inscribir una agencia de viajes con sus datos de contacto, expediente, datos de registro y sus representaciones.

Importador-Exportador: Es el componente encargado de inscribir una empresa importadora-exportadora con sus datos de contacto, datos de actividad económica, expediente, datos de registro.

Trámite: Es el componente encargado de definir los datos de los trámites.

Solicitante: Es el componente encargado de definir los datos de las solicitudes.

Reporte: Es el componente a partir del cual se generarán los documentos de salidas del sistema, permitirá la consulta de la información en forma tabular y apoyará los procesos de negocios restantes.

A continuación se muestra el Diagrama de Componentes para el RF-1 Generar Argumentación, con el objetivo de facilitar el entendimiento entre los componentes y sus relaciones. Para visualizar el modelo de componentes general del Módulo de Reportes (Ver Anexo 14).

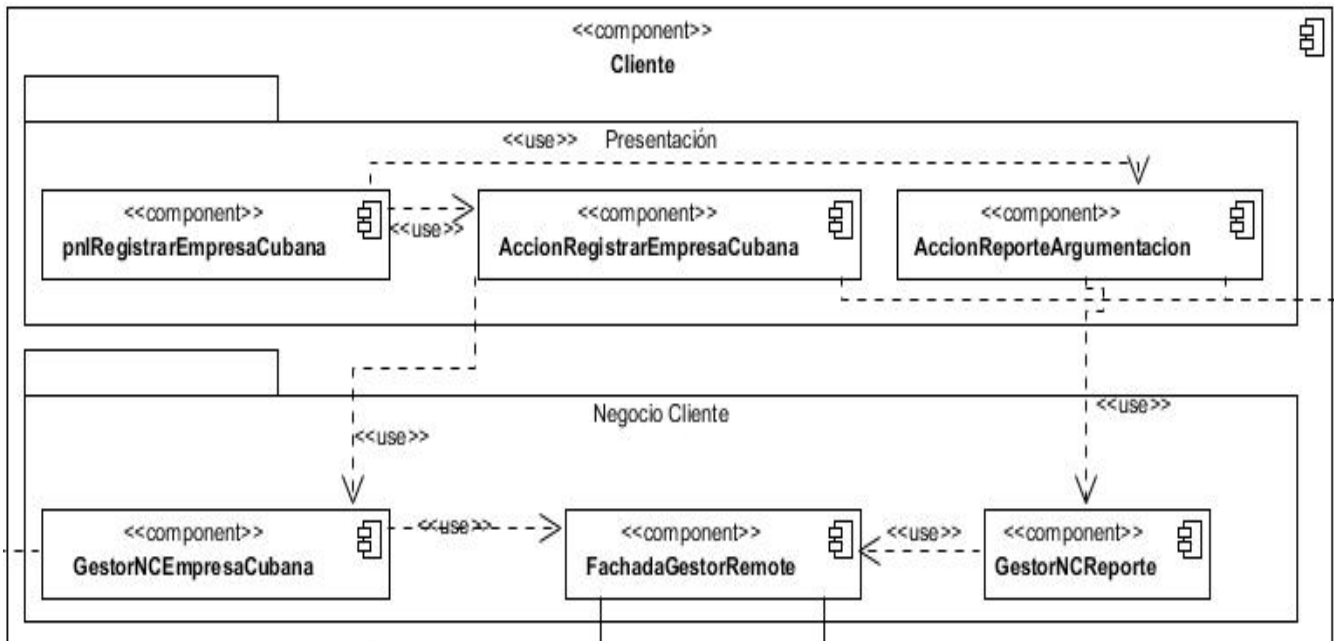


Imagen 2: Fragmento del Diagrama de Componentes (RF-1 Generar Argumentación)

2.5.2. Modelo de Despliegue

El Modelo de Despliegue se realizó como parte de la implementación y se representó con forma de descriptor, representando las principales funcionalidades del sistema dentro del contexto del entorno que las albergará. Permitió establecer la correspondencia entre la arquitectura lógica y los procesos y nodos. Este diagrama UML se utilizó para ilustrar los detalles de implementación, ilustrando los componentes y procesos en los nodos de proceso. Se reflejó en este artefacto los protocolos de comunicación mediante los cuales se comunicaran los nodos respectivos. Para la realización de este diagrama se tuvo en cuenta la distribución cliente-servidor con que cuenta la aplicación.

El despliegue físico de los procesos y componentes sobre los nodos de proceso, y la configuración de la red física entre los nodos fue visualizado con los diagramas de despliegue de UML. Para consultar el diagrama de despliegue (Ver Imagen 3).

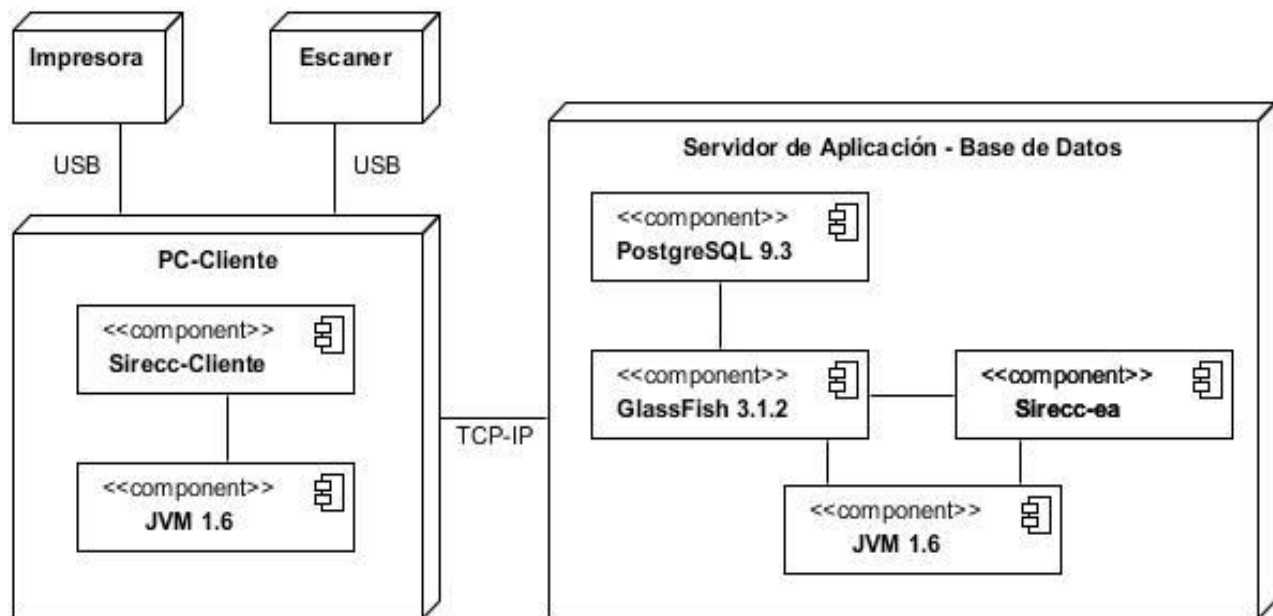


Imagen 3: Diagrama de Despliegue del Módulo de Reportes

2.5.3. Estándares de Codificación

En el caso del Proyecto SIRECC quedó establecido el Estándar de codificación para Java definido por la Dirección Técnica de la Producción basado en Convenciones de Código para el lenguaje de programación JAVA™ por Scott Hommel Sun Microsystems Inc. Estos estándares de codificación pueden ser consultados en el Expediente del Proyecto SIRECC.

Estos estándares de codificación establecen las características a cumplir para la implementación del proyecto incluyendo el Módulo de Reporte. Dentro de las normativas presentadas se encuentran las extensiones de los ficheros que pueden ser para Fuente Java (.java) y Bytecode de Java (.class). Otras de estas reglas son la organización de los ficheros que está compuesto por secciones que deben estar separadas por líneas en blanco y comentarios opcionales que identifican cada sección, deben ser evitados los ficheros de más de 2000 líneas. Las demás especificidades pueden ser consultadas en (Producción, 2011).

Luego de establecido los Estándares de Codificación se dio paso a la implementación. A continuación se brinda una descripción de como fue implementado el Módulo.

2.5.4. Implementación del componente

Estructura del Proyecto SIRECC e inclusión del Módulo de Reportes

Seguidamente se presenta la estructura del Proyecto SIRECC según el Marco de Trabajo GEFORT, mostrando cómo se organizará la implementación del módulo a desarrollar, de manera que facilite la organización y claridad durante el desarrollo. En esta sección se explicará el contenido de las carpetas utilizadas o creadas que complementan la solución propuesta. (Ver Imagen 4)

Contenido dentro de la carpeta “Reporte”: Dentro de esta carpeta se encuentran definidas las clases necesarias para la elaboración de los reportes como el Data Source Genérico. Se encuentran además los paquetes por módulos y los correspondientes reportes a cada uno de estos. Puede encontrarse además el cuerpo de cada reporte y todos los detalles del Módulo.

Contenido dentro de la carpeta “comun”: Se establecen las clases para la creación del cuerpo de reporte y su posterior visualización en formato PDF.

Contenido dentro de la carpeta “plantillas”: Contiene las plantillas de los reportes con sus variables y textos.

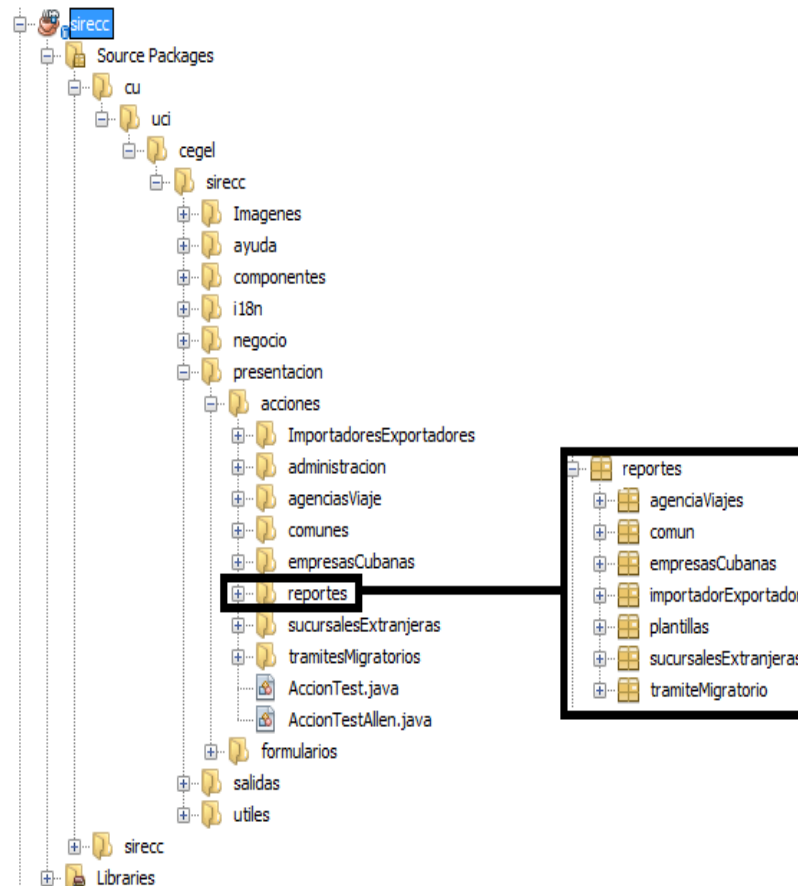


Imagen 4: Estructura del Proyecto SIRECC y el Módulo de Reportes

Contenido dentro de la carpeta “empresasCubanas”: Contiene las acciones correspondientes a la generación de reportes sobre Empresas Cubanas como documentos de salida durante la ejecución de los subprocessos de negocio y las acciones para visualizar los reportes en forma de tablas de este proceso.

Contenido dentro de la carpeta “sucursalExtranjeras”: Contiene las acciones correspondientes a la generación de reportes sobre Sucursales Extranjeras como documentos de salida durante la ejecución de los subprocesos de negocio y las acciones para visualizar los reportes en forma de tablas de este proceso.

Contenido dentro de la carpeta “agenciaViaje”: Contiene las acciones correspondientes a la generación de reportes sobre Agencias de Viajes como documentos de salida durante la ejecución de los subprocesos de negocio y las acciones para visualizar los reportes en forma de tablas de este proceso.

Contenido dentro de la carpeta “importadorExportador”: Contiene las acciones correspondientes a la generación de reportes sobre empresas Importadores-Exportadores como documentos de salida durante la ejecución de los subprocesos de negocio y las acciones para visualizar los reportes en forma de tablas de este proceso.

Contenido dentro de la carpeta “tramiteMigratorio”: Contiene las acciones correspondientes a la generación de reportes sobre Trámites Migratorios como documentos de salida durante la ejecución de los subprocesos de negocio y las acciones para visualizar los reportes en forma de tablas de este proceso.

2.5.5. Seguridad del sistema

La seguridad en el Módulo de Reportes cumple con las especificaciones planteadas para el proyecto SIRECC, especificadas en el documento Arquitectura Vista de Seguridad que es definido por el programa de mejora establecido en la UCI. Este documento puede ser consultado en el Expediente del Proyecto SIRECC, se encuentra además en (González Hernández, y otros, 2014).

En general la seguridad fue establecida en profundidad y por niveles siguiendo un conjunto de buenas prácticas para el desarrollo de aplicaciones seguras como definir las metas de seguridad del producto desarrollado y considerar la seguridad como una funcionalidad del producto. Se estableció el principio del menor privilegio y usar la defensa en profundidad, asumiendo que los sistemas externos son inseguros. Para complementar las medidas de seguridad en la aplicación y en el Módulo de Reportes se

definieron los siguientes requisitos no funcionales en el rango de RnF-14 hasta RnF-18 que pueden ser consultados en el Capítulo 2: Solución de Software, epígrafe 2.3, sub-epígrafe 2.3.1.

Al concluir el diseño y la implementación comenzaron las actividades para validar estas fases. Los resultados alcanzados en la validación puede ser consultada en el Capítulo 3: Análisis de los resultados, epígrafe 3.2 Validación de las fases de desarrollo, sub-epígrafe 3.2.2.

Al terminar la validación de la implementación comenzó la validación del Módulo de Reportes mediante la realización de Pruebas de Software. A continuación se describe el proceso de pruebas aplicadas al sistema.

2.6. Pruebas

Las pruebas de software realizadas estuvieron enmarcadas en detectar la mayor cantidad de defectos de software posibles para su debida corrección. Se realizaron una serie de actividades para encontrar posibles fallos de implementación, calidad o usabilidad del sistema, probando el comportamiento del módulo.

Para el Módulo de Reportes se realizaron las comprobaciones establecidas por el programa de mejora: Pruebas Internas y Pruebas de Liberación. Las Pruebas Internas se realizaron para probar desde funcionalidades específicas hasta la versión final y se desarrollaron artefactos de pruebas como los Casos de Pruebas (CP). Las pruebas de liberación fueron realizadas por el Centro Nacional de Calidad de Software (CALISOFT). Para consultar los resultados alcanzados en las pruebas de software ver Capítulo 3: Análisis de los resultados, epígrafe 3.3 Pruebas, sub-epígrafes 3.4 y 3.5 titulados Aplicación de las Pruebas de Caja Blanca y Aplicación de las Pruebas de Caja Negra respectivamente.

2.7. Conclusiones del capítulo

Al finalizar este capítulo se concluyó que:

- ✓ La modelación del negocio permitió comprender los procesos a informatizar de la institución en cuestión.
- ✓ Se realizó la definición, especificación y descripción de los requisitos del módulo que permitió cumplir con todas las funcionalidades requeridas para el proceso de generación de Reportes de Información.
- ✓ La utilización de los patrones de diseño y arquitectura durante el desarrollo del módulo, proporcionaron una mayor calidad del producto,
- ✓ El diseño del modelo de datos permitió conocer las relaciones existentes entre las diferentes tablas de la base de datos y la descripción del diagrama de clases de diseño permitió conocer la estructura y las relaciones entre las clases que se manejan en el componente.
- ✓ Los diagramas de componentes y de despliegue facilitaron la fase de implementación.
- ✓ La utilización de los estándares de implementación a utilizar, facilitaron el entendimiento del código por los programadores y el futuro mantenimiento del componente.
- ✓ La seguridad del sistema se estableció como funcionalidad del producto.
- ✓ La validación del diseño, a través de métricas arrojaron resultados satisfactorios sobre el diseño realizado, demostrando que este era simple y los atributos de calidad alcanzaban niveles favorables.
- ✓ La realización de pruebas de caja blanca y caja negra al componente arrojaron resultados satisfactorios, demostrándose con ello la calidad del componente realizado.

3

ANÁLISIS DE RESULTADOS

“Ahora existe Internet... Nada debe bloquear la obtención de conocimientos”

Fidel Castro Ruz

3.1. Introducción

En el presente capítulo se realizarán las validaciones correspondientes a las fases propuestas por el modelo de desarrollo como buenas prácticas del desarrollo de software. Se validarán las variables dependiente e independiente a través de métricas. Se cuantificarán los resultados obtenidos y se graficarán con el objetivo de su apreciación visual.

3.2. Validación de las fases de desarrollo

Con el objetivo de obtener un producto de elevada calidad, mantener un proceso estable y continuo de desarrollo, para cometer el menor número no conformidades posibles y con el propósito de lograr que las salidas de cada etapa de desarrollo que se conviertan en entradas a la siguiente etapa se realizaron las validaciones de cada una de las fases del ciclo de desarrollo. A continuación la validación de estas fases.

3.2.1. Validación de la modelación del negocio y los requisitos de software

Las técnicas que se utilizaron para la validación de los requisitos identificados fueron las siguientes:

Revisión Técnica Formal, una vez terminada la descripción de los requisitos se realizaron revisiones a cada una de estas descripciones. Ante los errores detectados se procedió a la corrección para comenzar una nueva revisión. Con esta técnica se pudo validar que la interpretación de cada una de las descripciones no fuera ambigua, ni tuviese omisiones o errores y además que cada uno de los requisitos cumplía con lo que necesitaba el cliente.

Prototipos de interfaz, es otra de las técnicas utilizadas donde se hicieron simulaciones del posible producto (reportes). Cada uno de los prototipos le permitió a los especialistas tener una idea de las

interfaces gráficas del módulo. Estos se realizaron de forma no funcional con la herramienta Visual Paradigm for UML con el objetivo de lograr mayor aprobación por parte del cliente.

Criterios para validar requisitos del cliente, artefacto definido por el Programa de Mejora para las validaciones de requisitos. Establecidos en el Libro de Proceso para la Administración de Requisitos del modelo seleccionado. Permite establecer una evaluación de los requisitos mediante los parámetros fijados en (SIRECC, 2012), permitiendo el entendimiento y compromiso con/para el cliente.

Para validación de los requisitos no funcionales se aplicó la encuesta que se muestra en el Anexo 5 (resultados de aplicación de la encuesta en Anexo 16), en esta tabla se establecieron los parámetros a satisfacer por la aplicación. Estos parámetros quedan establecidos en (González Hernández, y otros, 2014). En el documento de arquitectura del proyecto SIRECC se establecen las características del cumplimiento de estos requisitos no funcionales, teniendo en cuenta que fueron levantados con las especificidades de software, hardware y tráfico de red con que cuenta la entidad. Otro de los métodos utilizados para corroborar el funcionamiento correcto de los RnF fue la Lista de Chequeo Usabilidad de Desktop. Los resultados arrojados en estas pruebas pueden ser consultadas en el presente Capítulo 3: Análisis de los resultados, epígrafe 3.2 Validación de la variable independiente, sub-epígrafe 3.4.2 Calidad.

Partiendo de las especificaciones y descripciones de los requisitos se dio paso a la fase de análisis y diseño para su posterior implementación, estas documentaciones fueron utilizadas en conjunto a los diagramas generados para obtener un correcto diseño del sistema. Seguidamente las tareas realizadas en las validaciones de las fases Análisis, Diseño e Implementación.

3.2.2. Validación del diseño y la implementación

La validación de del diseño e implementación se realizó mediante la Métrica: Complejidad de Implementación, donde se cuantificó mediante el uso de la técnica: Tamaño Operacional de Clase (TOC) propuesta por Lorenz y Kidd. El tamaño general de las clases se determinó mediante las siguientes medidas:

- ✓ El total de operaciones (TO), tanto heredadas como privadas de la instancia, que se encapsulan dentro de la clase.
- ✓ El número de atributos (NA), tanto heredados como privados de la instancia, encapsulados por la clase.

Medidas para las principales clases de la solución

La métrica de TOC fue aplicada a un total de 10 clases para un total de 48 atributos y un promedio (P) de atributos de 4.8. Para un total de 90 operaciones y un promedio operacional de 9.

Responsabilidad y Complejidad de Implementación (Cálculo del umbral)

(NA) Baja: $P \leq 4.8$ $4.8 < Media \leq 9.6$ Alta > 9.6 **(TO)** Baja: $P \leq 9$ $9 < Media \leq 18$ Alta > 18

Atributo	Clases	NA	TO	Tamaño (NA)	Tamaño (TO)
Responsabilidad y Complejidad de Implementación	DSGenerico	5	13	Media	Media
	DrepoteJasper	7	17	Media	Media
	FactoriaEDReporte	1	2	Baja	Baja
	GestorADReporte	3	7	Baja	Baja
	EDReporte	10	18	Alta	Media
	DSReporteColumnasDinamicas	7	3	Media	Baja
	AccionReporteArgumentacion	5	8	Media	Baja
	AccionVisualizarAsambleaGeneral	1	6	Baja	Baja
	AccionReporteVisorPDF	5	10	Media	Media
	CuerpoReporte	4	6	Baja	Baja

Tabla 4: Cálculo del TOC para los atributos Responsabilidad y Complejidad de Implementación.

Reutilización (Cálculo del umbral)

(NA) Baja: $P > 9.6$ $4.8 < Media \leq 9.6$ Alta ≤ 4.8 (TO) Baja: $P > 18$ $9 < Media \leq 18$ Alta ≤ 9

Atributo	Clases	NA	TO	Tamaño (NA)	Tamaño (TO)
Reutilización	DSGenerico	5	13	Media	Media
	DrepoteJasper	7	17	Media	Media
	FactoriaEDReporte	1	2	Alta	Alta
	GestorADReporte	3	7	Alta	Alta
	EDReporte	10	18	Baja	Media
	DSReporteColumnasDinamicas	7	3	Media	Alta
	AccionReporteArgumentacion	5	8	Media	Alta
	AccionVisualizarAsambleaGeneral	1	6	Alta	Alta
	AccionReporteVisorPDF	5	10	Media	Media
	CuerpoReporte	4	6	Alta	Alta

Tabla 5: Cálculo del TOC para el atributo Reutilización.

Resultados obtenidos de la aplicación de la métrica al componente

Con la representación de los resultados obtenidos en un gráfico de por ciento (Ver Anexo 9 y 10) se obtuvo que: del total de clases analizadas en cuanto a los atributos Responsabilidad y Complejidad de Implementación, el 50% presenta un TOC (NA) mediano y el 40% un TOC (TO) en igual calificación. De tamaño bajo se encuentran el TOC (NA) con el 40% y TOC (TO) igual a 60%. En cuanto a calificación como alta, del total, el TOC (NA) cuenta con 10% y 0% el TOC (TO). Sobre el atributo Reutilización se tiene que la calificación media se comporta de igual manera e los atributos anteriores, mientras que la baja es TOC (NA) de 10% y nulo para TOC (TO). Por otra parte, alta es para los valores correspondientes a TOC (NA) igual a 40% y TOC (TO) de 60%.

El análisis de los datos anteriormente planteados permite concluir que el módulo presenta una reutilización alta, mientras que la Responsabilidad de las clases y Complejidad de Implementación se comportan de manera media. Estos datos favorecen el buen uso del diseño empleado y el resultado de la implementación.

3.3. Validación de las variables de investigación

Para la validación del **Módulo de Reportes (Variable independiente)** se realizó un estudio de las principales técnicas utilizadas para dar solución al problema inicial, se hizo uso de métricas para medir la reutilización, limpieza y buenas prácticas de diseño. Estas métricas sirvieron de instrumentos para cuantificar criterios sobre la calidad del producto.

Las métricas básicas estudiadas abarcan atributos de calidad que permitieron medir la calidad del diseño propuesto. Estos atributos pueden ser consultados en el Capítulo I: Fundamentación Teórica, epígrafe 1.6, sub-epígrafe 1.6.1. Estas métricas concebidas para evaluar la calidad del diseño se seleccionaron teniendo en cuenta las características propias del módulo. Estas son: Alcance y Calidad que permitirán la validación de la variable independiente.

3.3.1. Alcance

El alcance del producto a desarrollar fue determinado por el cumplimiento de los requisitos. Para ello se calculó del Índice de requisitos implementados. A continuación se desarrollan los cálculos pertinentes.

- ✓ Cantidad de requisitos implementados (CRI) = 44
- ✓ Cantidad total de requisitos (CTR) = 44
- ✓ Índice de requisitos implementados (IRI) = $CRI / CTR = 44 / 44 = 1.0$

A continuación se presenta el análisis gráfico correspondiente a los cálculos realizados.



Imagen 5: Gráfica de Índice de Requisitos Implementados

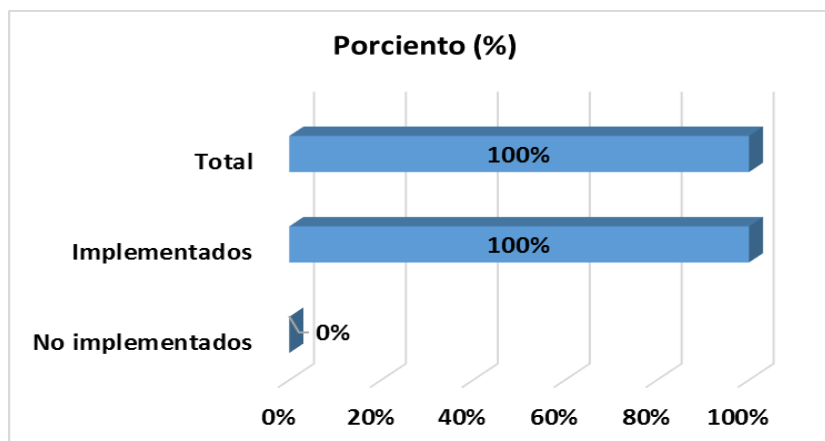


Imagen 6: Gráfica de Porciento de Requisitos Implementados

3.3.2. Calidad

Para evaluar la Calidad del producto desarrollado se tomaron como indicadores la Calidad Interna y Externa. La Calidad Interna será cuantificada con el Índice de no conformidades detectadas en las Pruebas Internas y por CALISOFT durante las pruebas de Liberación, mientras que para realizar las mediciones correspondientes a la Calidad Externa se realizará el cálculo del valor según el modelo definido por la ISO.

En el presente sub-epígrafe se abordarán los resultados obtenidos en la aplicación de las pruebas de software mencionadas en el Capítulo 2: Solución de Software, epígrafe 2.6 Pruebas.

Evaluación de la Calidad Interna (Aplicación de las Pruebas de Caja Blanca)

Se realizaron con conocimiento de la estructura interna del programa y su objetivo principal se enmarcó en probar que todos los caminos del código están correctos. La Prueba de Caja Blanca empleada, fue la Prueba del Camino Básico, la cual permitió obtener una medida de la complejidad lógica del diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. De esta manera se garantiza que durante la prueba se ejecute por lo menos una vez cada sentencia del programa.

Fue necesario calcular antes la complejidad ciclomática del algoritmo o fragmento de código a analizar. A continuación se muestra el grafo de flujo asociado al método y se enumeran las sentencias de código del procedimiento realizado sobre el método *llenarMapaReporte()*, ver (Imagen 5).

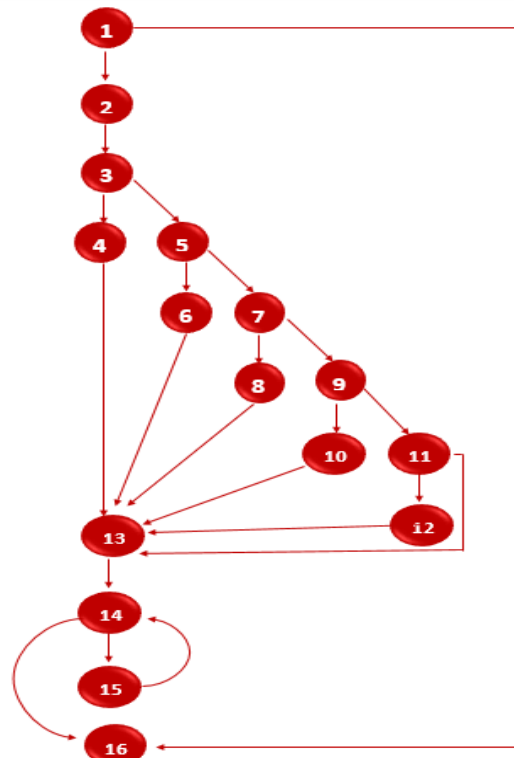


Imagen 7: Grafo de flujo asociado al método *llenarMapaReporte()*

```
private void llenarMapaReporte() throws Exception {
    if (!ListaVariableValor.isEmpty()) { //1
        String fax = ""; //2
        if (documento instanceof EDEmpresaCubana) { //3
            path = "cu.uci.cegel.sirecc.entidades.EDEmpresaCubana"; //4
        } else if (documento instanceof EDSucursal) { //5
            path = "cu.uci.cegel.sirecc.entidades.EDSucursal"; //6
        } else if (documento instanceof EDAgenciaViaje) { //7
            path = "cu.uci.cegel.sirecc.entidades.EDAgenciaViaje"; //8
        } else if (documento instanceof EDImportadoresExportadores) { //9
            path = "cu.uci.cegel.sirecc.entidades.EDImportadoresExportadores"; //10
        } else if (documento instanceof EDTramiteMigratorio) { //11
            path = "cu.uci.cegel.sirecc.entidades.EDTramiteMigratorio"; //12
        }
        Class clase = Class.forName(path); //13
        for (int i=0; i< listaVariableValor.size(); i++) { //14
            String varJasper = listaVariableValor.get(i).getVariable(); //15
            String varMetodo = listaVariableValor.get(i).getValor(); //15
            Class [ ] arg = new Class [ ]; //15
            mapaReporte.put(varJasper, clase.getMethod(valorMetodo), arg).invoke(documento,
            new Object[]{})); //15
        }
    }

    InputStream logoCamara = new ByteArrayInputStream(EDReporte.getLogo()); //16
    mapaReporte.put("logo", logoCamara); //16
    mapaReporte.put("fecha", getFechaActualSistema()); //16
    mapaReporte.put("telefonos", FactoriaContexto.getFactoria().getFrameworkClienteContexto.
    getusuarioAutenticado().getOficina().getNombre()); //16
}
```

Cálculo de la complejidad ciclométrica:

$V(G) = (A - N) + 2$ Donde "A" es la cantidad de aristas y "N" la cantidad de nodos.

$$V(G) = (22 - 16) + 2 = 8$$

$V(G) = P + 1$ Siendo "P" la cantidad de nodos predicados (nodos donde parten $A \geq 2$).

$$V(G) = 7 + 1 = 8$$

$V(G) = R$ Donde "R" representa la cantidad de regiones en el grafo.

$$V(G) = 8$$

El cálculo efectuado mediante las fórmulas ha dado el mismo valor, por lo que se puede decir que la complejidad ciclométrica del código es de 8, lo que significa que existen 8 posibles caminos por donde el flujo puede circular, este valor representa el límite mínimo del número total de CP para el procedimiento tratado, ver (Tabla 6).

No.	Camino Básico
1	1-2-3-4-13-14-16
2	1-2-3-5-6-13-14-16
3	1-2-3-5-7-8-13-14-16
4	1-2-3-5-7-9-10-13-14-16
5	1-2-3-5-7-9-11-12-13-14-16
6	1-2-3-5-7-9-11-13-14-16
7	1-2-3-5-7-9-11-13-14-15-14-16
8	1-16

Tabla 6: Caminos básicos del flujo.

Posteriormente de haber determinado los caminos básicos se procedió a ejecutar los Casos de Pruebas para cada uno de estos. Para definir los CP se consideraron las descripciones, condiciones de ejecución, entradas, resultados esperados y los resultados obtenidos. Estos Casos de Prueba pueden ser consultados en el documento Casos de Prueba para el Camino Básico, ubicado en (González Hernández, y otros, 2014).

Evaluación de la Calidad Interna (Aplicación de las Pruebas de Caja Negra)

Para realizarlas no fue necesario conocer los detalles internos del programa y su objetivo fundamental fue probar que el software se encontraba acorde a los requisitos. Para la ejecución de este tipo de pruebas se realizaron un conjunto de CP para determinar si los requisitos estaban parcial o completamente satisfactorios. En total se realizaron 44 CP, probándose cada una de las funcionalidades y evaluándose sus resultados. Estos Casos de Prueba pueden ser consultados en (González Hernández, y otros, 2014). Estas Pruebas de Caja Negra se dividirán en internas y externas.

Resultados de las Pruebas Internas

El componente fue sometido a 2 iteraciones de pruebas de caja negra. En la primera iteración se detectaron 39 no conformidades, de las cuales 14 fueron de aplicación y 25 de documentación. Dentro de las no conformidades referentes a las de aplicación se encuentran 1 de validación, 1 de interfaz, 5 de funcionalidad, 6 de formato y 1 de ortografía. En cuanto a las de documentación se encontraron 17 no conformidades de redacción y 8 de correspondencia. Las no conformidades fueron resueltas posterior a su detección, lo que permitió que el componente pasara a una segunda iteración.

En esta nueva iteración se detectaron 17 no conformidades, clasificadas en 14 de aplicación: 2 de interfaz, 2 de funcionalidad, 3 de formato y 7 de ortografía. En la documentación se detectaron 2 no conformidades de formato y 1 de ortografía. Al igual que en la primera iteración, se resolvieron las deficiencias encontradas y se comenzó con la tercera iteración en la que no se detectaron no conformidades y se obtuvieron resultados satisfactorios.

Resultados de las Pruebas de Liberación

En las pruebas de liberación, el componente fue sometido a 2 iteraciones de pruebas de caja negra. En la primera iteración se detectaron 2 no conformidades, de las cuales 2 fueron de aplicación y no se encontraron inconsistencias de documentación. Dentro de las no conformidades referentes a las de aplicación se encuentran 1 de ortografía y 1 de funcionalidad. Las no conformidades fueron resueltas posterior a su detección, lo que permitió que el componente pasara a una segunda iteración en la que no se detectaron no conformidades, obteniendo resultados satisfactorios.

El módulo desarrollado fue presentado al cliente, luego de haber interactuado con la aplicación y comprobado su funcionalidad se mostró satisfecho con el producto entregado. Prueba de lo planteado anteriormente constituye el acta de aceptación emitida, en la cual consta que el componente realizado cumple con las condiciones de calidad necesarias y satisface las necesidades plasmadas por el cliente, además contribuye a mejorar la integridad de los reportes en el Sistema de Informatización Registral de la Cámara de Comercio.

A estas pruebas de liberación realizadas por CALISOFT se le agregó las comprobaciones mediante la Lista de Chequeo Usabilidad de Desktop. Esta lista de chequeo fue confeccionada con la selección de los puntos a evaluación, los cuales se clasificaron antes de ser aplicados, teniendo en cuenta su pertinencia, además de tener un punto donde se referencia las observaciones en cuanto al aspecto a evaluar. Esta plantilla permitió recoger los puntos eficientes e ineficientes que tienen los elementos chequeados. A continuación se tabulan los resultados de la aplicación de la lista de chequeo.

Resultados de la Lista de Chequeo de Usabilidad de Desktop			
Elementos definidos	Cantidad indicadores evaluados	Cantidad de indicadores con peso crítico	Cantidad de no conformidades
Visibilidad del sistema	5	5	0
Control y libertad del usuario	8	6	0
Consistencias y estándares	1	1	0
Prevención de errores	2	0	0
Reconocimiento	0	0	0
Flexibilidad y eficiencia de uso	2	0	0
Estética	5	4	0
Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	1	1	0
Ayuda y documentación	0	0	0
Accesibilidad	0	0	0
Comprobaciones técnicas	0	0	0
Semántica de la aplicación	1	0	0

Resultados: las no conformidades fueron documentadas y posteriormente resueltas.

Tabla 7: Resultados de la Lista de Chequeo de Usabilidad de Desktop

Los parámetros chequeados por CALISOFT permitieron evaluar los requerimientos no funcionales. En correspondencia, los resultados de estas revisiones arrojaron resultados satisfactorios; no se encontraron no conformidades en las evaluaciones realizadas, reafirmando la calidad con que fue desarrollado el módulo. Estos elementos de chequeo pueden ser consultados en el documento Lista de Chequeo Usabilidad de Desktop que se encuentra en (González Hernández, y otros, 2014).

Evaluación de la Calidad Externa

Por otra parte, la Calidad Externa se comportó de manera correcta según los resultados obtenidos con los indicadores citados por la norma ISO, donde se realizaron cuestionarios a los clientes de la cámara. En este caso, los especialistas de atención de las áreas de la empresa evaluaron los parámetros citados en: Funcionalidad 20.58% de 21%, Fiabilidad 21.78% de 22%, Usabilidad 13.05% de 15%, Eficiencia 18% de 18%, Facilidad de Mantenimiento 15.52% de 16%, Portabilidad 8% de 8%. Estos cálculos fueron establecidos entre los especialistas de las áreas y los informáticos de la entidad, además se encontraban presentes los desarrolladores del módulo. Para una visualización detallada de los resultados consultar el Anexo 16.

Para la validación del **tiempo de generación de reportes (variable dependiente)** se realizó un estudio de los tiempos que brinda la solución desarrollada al problema de la investigación. Se cuantificaron las medidas de las unidades antes mencionadas, se corroboraron su correspondencia con los requisitos no funcionales y los beneficios ofrecidos en cuanto a la disminución de los tiempos de ejecución en los procesos de la entidad. A continuación los resultados obtenidos durante las mediciones.

3.3.3. Tiempo

Para obtener los valores del tiempo de generación de reportes, se midieron los tiempos de ejecución de las consultas en la Base de Datos, donde se pudo apreciar que todas las consultas realizadas tardaban menos de 1 segundo en ejecutarse, comportándose como un tiempo favorable. Se midieron los tiempos de respuesta finales de los reportes incluyendo las consultas pertinentes a cada uno, en estas pruebas realizadas se constató que al igual que los anteriores, los tiempos de respuesta obtenidos fueron menores que 1 segundo.

Se comprobaron las dos clasificaciones de los reportes, tanto los documentos de salida, como los reportes de tablas del sistema. Los tiempos de respuesta obtenidos durante la obtención de reportes de información cumplieron con las expectativas y criterios establecidos en el RnF-04, con tiempos menores que 3 segundos. Para corroborar los tiempos de respuesta del sistema en cuanto a reportes, puede consultarse el documento Tiempo de Generación de Reportes en (González Hernández, y otros, 2014).

La mejora en el tiempo de generación de reportes se convirtió en un incremento en el número de número de empresas a atender en un día, ejemplo de ello es el área encargada del registro de empresas Importadoras-Exportadoras que anteriormente solo atendían a 20 solicitudes de afiliación a la Cámara diariamente. Con el uso del Módulo de Reportes la documentación necesaria para estos procesos se obtiene de manera rápida y son atendidas más de la cantidad anteriormente mencionada. Otra ventaja constatada fue la consulta concurrente de los documentos de los procesos informatizados, donde varias personas pueden consultar documentación única.

El formato y estandarización de los informes generados en las diferentes áreas de la entidad, así como la reestructuración de la documentación en: encabezado, cuerpo y pie de página permitieron cumplir con lo establecido en el Manual de Identidad Visual de la Cámara de Comercio, bajo las normas, modelos y patrones ahí expresados.

3.4. Conclusiones del capítulo

En este capítulo se realizó la validación del componente, arrojando resultados importantes como:

- ✓ Se validaron las fases de desarrollo abarcadas durante el desarrollo del producto como buenas prácticas permitiendo obtener un producto de calidad.
- ✓ Se realizaron las pruebas de software, arrojando resultados satisfactorios en el desarrollo del módulo como producto.
- ✓ Se realizó la validación de la variable independiente mediante las métricas de Alcance y Calidad.
- ✓ Se realizó la validación de la variable dependiente mediante la evaluación de los tiempos de respuesta de los reportes.

CONCLUSIONES

Con la culminación del presente trabajo de diploma se puede concluir que se desarrollaron todas las tareas a fin de cumplir el objetivo propuesto, resaltando que:

- ✓ Se realizó el marco teórico referente a la investigación donde se establecieron las tendencias marcadas en cuanto al desarrollo de reportes de información y permitió fundamentar la necesidad de desarrollar el módulo.
- ✓ Se desarrolló el Módulo para el proyecto SIRECC abarcando las etapas establecidas por el Modelo de Desarrollo: Programa de Mejora. La implementación del Módulo de Reportes se realizó utilizando las herramientas, lenguajes y tecnologías definidas por el modelo utilizado.
- ✓ Las pruebas realizadas arrojaron resultados satisfactorios, demostrando la calidad del componente desarrollado, mientras que el módulo fue validado mediante el uso de métricas y técnicas aplicables al desarrollo de la solución.

RECOMENDACIONES

Como recomendaciones del presente trabajo, se tienen:

- ✓ Utilizar componentes del Módulo de Reportes en aplicaciones del tipo registral con el objetivo de facilitar el trabajo y la reutilización.
- ✓ Incorporar el Módulo de Reportes al Marco de Trabajo GEFORT como componente de desarrollo permitiendo su utilización por futuros proyectos, permitiendo reutilizar componentes del subsistema en aplicaciones registrales.
- ✓ Implementar la funcionalidad que permita generar nuevos reportes dinámicos según parámetros de búsqueda establecidos por los especialistas de las áreas.

BIBLIOGRAFÍA

- Abreu Medina, Aldis Joan. 2012.** *Generador Dinámico de Reportes*. Universidad de las Ciencias Informáticas (UCI). La Habana : s.n., 2012. pág. 6, Comunicación Corta. 11.
- BIOMUNDI. 2008.** *Inteligencia Empresaria. Qué y Cómo*. [ed.] Marlene Sardina Prado. La Habana : Palcograf, Palacio de las Convenciones, 2008. pág. 345.
- Business Objects. 2012.** SAP Crystal solutions. [En línea] 2012. [Citado el: 18 de diciembre de 2013.] <http://www.crystalreports.com/es/>.
- CALISOFT. 2011.** *Aseguramiento de la Calidad del proceso y el producto* . 2011. pág. 4.
- CALISOFT2. 2010.** Centro Nacional de Calidad de Software. [En línea] 2010. [Citado el: 15 de enero de 2013.] https://calisoft.uci.cu/index.php?option=com_phocadownload&view=category&id=6&Itemid=149.
- Cámara de Comercio de la República de Cuba. 2009.** Cámara de Comercio de la República de Cuba. *El portal del Empresario en Cuba*. [En línea] 2009. [Citado el: 20 de 10 de 2013.] <http://www.camaracuba.cu/>.
- Carrasco Puebla, Yoan Arlet y Chaviano Gómez, Enrique. 2009.** GestioPolis. [En línea] 15 de mayo de 2009. [Citado el: 6 de diciembre de 2014.] <http://www.gestiopolis.com/administracion-estrategia/erp-arquitectura-orientada-a-servicios.htm>.
- Comercio, Cámara de. 2012.** *Manual de Identidad de la Cámara de Comercio*. La Habana : s.n., 2012. pág. 57.
- Días, Gilberto. 2011.** Web del Profesor. [En línea] 2011. [Citado el: 17 de enero de 2014.] http://webdelprofesor.ula.ve/ingenieria/gilberto/paralela/05_LeyDeAmdahlyMoore.pdf.
- Febles, Aylin. 2011.** *Programa de mejora*. [entrev.] Dunieski Sarmiento Méndez. 2011.
- González Hernández, Ernesto Pastor y Torres Ruíz, Allens. 2014.** Entregables Módulo de Reportes proyecto SIRECC. La Habana, Cuba : s.n., 2014.
- Grape City. 1997.** Active Report Server. [En línea] 1997. [Citado el: 18 de noviembre de 2013.] <http://www.activereportserver.com/>.
- Hernández Sampieri, Roberto, Fernández-Collado, Carlos y Baptista Lucio, Pilar. 2006.** *Metodología de la Investigación Científica*. [ed.] Ricardo A. del Bosque Alayón. Cuarta. México D. F. : The McGraw Hill, 2006. pág. 850.
- IEEE2. 1999.** IEEE Advancing Technology for Humanity. [En línea] 1999. [Citado el: 27 de octubre de 2013.] <http://www.ieee.org/index.html>.

- Institute, Project Management. 2012.** Líder de Proyecto. [En línea] 2012. [Citado el: 2014 de febrero de 15.] <http://www.liderdeproyecto.com/sitio/component/agorapro/default-topic/1299-diferencia-entre-la-definici%C3%B3n-de-alcance-del-producto-y-del-proyecto.html>.
- ISO. 2006.** Determinación de los Requerimientos de Calidad del Producto Software basados en Normas Internacionales. [En línea] abril de 2006. [Citado el: 2014 de febrero de 18.] http://www.ewh.ieee.org/reg9etransieeeeissuesvol04vol4issue2April20064TLA2_06Davila.pdf.
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software*. Madrid : Addison Wesley, 2000. pág. 464.
- JasperSoft. 2011.** *The JasperReports Ultimate Guide*. Tercera. 2011. pág. 333.
- Jorrín Perdomo, Jorge Yuniel y Prieto Álvarez, Julio Cesar. 2011.** *Proyecto Técnico*. La Habana : s.n., 2011. pág. 44.
- Jorrín Perdomo, Jorge Yuniel. 2013.** *Proyecto Técnico SIRECC*. Universidad de las Ciencias Informáticas (UCI). La Habana : s.n., 2013. pág. 44, Técnico.
- Larman, Craig. 1999.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. Segunda. Ciudad de México : Prentice Hall, 1999. pág. 536.
- Lorenz, M y Kidd, J. 1994.** *Object-Oriented Software Metric*. 1994.
- Microsoft. 2012.** TN Reporting Services (SSRS). [En línea] Microsoft, 2012. [Citado el: 24 de noviembre de 2013.] <http://technet.microsoft.com/es-es/library/ms159106.aspx>.
- Microsoft. 2013.** Windows Server. [En línea] 2013. [Citado el: 21 de enero de 2014.] <http://technet.microsoft.com/es-es/library/cc753794%28v=ws.10%29.aspx>.
- MSDN.Microsoft. 2013.** Revisiones de código y estándares de codificación. *msdn*. [En línea] 2013. [Citado el: 19 de enero de 2014.] <http://msdn.microsoft.com/es-es/library/aa291591%28v=vs.71%29.aspx>.
- OMG Available Specification. 2008.** *Business Process Modeling Notation, V1.1*. 2008. pág. 318.
- Oracle Corporation. 2013.** NetBeans IDE. *The Smarter and Faster Way to Code*. [En línea] 2013. [Citado el: 22 de octubre de 2013.] <https://netbeans.org/>.
- ORACLE. 2013.** GlassFish - World's first Java EE 7 Application Server. [En línea] 2013. [Citado el: 15 de enero de 2014.] <https://glassfish.java.net/>.
- Panda, Debu, Rahman, Reza y Lane, Derek. 2007.** *EJB 3 IN ACTION*. s.l. : Manning Publications Co., 2007. pág. 707.

- Prieto Álvarez, Julio Cesar. 2012.** *Plan de Desarrollo de Software*. Universidad de las Ciencias Informáticas. La Habana : s.n., 2012. pág. 30.
- Producción, Dirección Técnica de. 2011.** Estándar de codificación para Java. La Habana, Cuba : s.n., 2011. pág. 24.
- Programa de Mejora. 2011.** *Ciclo de vida de proyectos pilotos del Programa de Mejora*. La Habana : s.n., 2011. pág. 9.
- Project Management Institute. 2008.** *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK)*. Cuarta. Pennsylvania : s.n., 2008. pág. 502.
- Reese, Richard M. 2011.** *EJB 3.1 Cookbook*. Birmingham : Packt Publishing, 2011. pág. 436.
- Roberto, Hernández Sampieri, Carlos, Fernández Collado y Pilar, Baptista Lucio. 2006.** *Metodología de la Investigación Científica*. Iztapalapa, México : s.n., 2006. 4ta Edición.
- S. Pressman, Roger. 2007.** *Software Engineering. A Practitioner´s Approach*. Seventh. New York : Mc Graw Hill, 2007. pág. 980.
- Sánchez-Segura, Maribel y Ana M. 2010.** *Patrones de Usabilidad: Mejora de la Usabilidad del Software desde el momento de Arquitectura*. Madrid : s.n., 2010.
- SEI. 1990.** Software Engineering Institute. [En línea] 1990. [Citado el: 19 de octubre de 2013.] <http://www.sei.cmu.edu/>.
- SIRECC. 2012.** Criterios para validar requisitos del cliente. La Habana, Cuba : s.n., 2012. pág. 2.
- Sommerville, Ian. 2005.** *Ingeniería de Software*. Séptima. Madrid : Pearson Addison Wesley, 2005. pág. 712.
- Suárez del Villar, Claudia García, Brito Rodríguez, Julio Cesar y Brizuela Figueredo, Clara Elena. 2013.** *Herramienta para importar reportes*. La Habana : Serie científica de la UCI, 2013.
- The PostgreSQL Global Development Group. 1996.** PostgreSQL. *The world´s most advanced open source database*. [En línea] 1996. [Citado el: 6 de diciembre de 2013.] <http://www.postgresql.org/>.
- Torres, Maritza y Rojas, Daniel. 2008.** *Modelo de evaluación de la calidad de la información corporativa*. s.l. : Revista Venezolana de la Información, Tecnología y Conocimiento, 2008. 3.
- Universidad Central de Chile. 2011.** Universidad de Chile. Departamentos de ciencias de la computacion.Sitio Web DCC. [En línea] 2011. [Citado el: 6 de febrero de 2014.] www.dcc.uchile.cl/~psalinas/uml/modelo.html.
- Visual Paradigm. 2013.** Visual Paradigm. [En línea] Visual Paradigm, 2013. [Citado el: 13 de enero de 2013.] <http://www.visual-paradigm.com/>.

ANEXOS

Anexo 1: Encuesta a especialistas de la Cámara de Comercio

La generación de reportes es esencial para los sistemas de información, puede producir la información requerida y transmitirla a los puntos o centros de información que la soliciten. Esta transmisión de información se puede efectuar mediante el movimiento físico de los elementos de almacenamiento (papel, cintas magnéticas, disco 3 ½, y otros) o mediante la comunicación de señales eléctricas digitales o analógicas a dispositivos receptores (terminales, convertidores, estaciones remotas u otro computador). Permiten conocer la evolución de una empresa en cada aspecto de interés, visualizar la empresa desde diferentes puntos de vista, identificar situaciones inesperadas y tomar decisiones en base a estadísticas ciertas.

El objetivo de esta encuesta es identificar ¿Cómo se realiza el procedimiento de la obtención de reportes en la Cámara de Comercio, sus deficiencias y facilidades de informatización?

1. ¿Cómo se realizan actualmente los procesos de registro de Empresas Cubanas, Sucursales Extrajeras, Agencias de Viajes, Importadores y Exportadores, así como Trámites Migratorios?
 - 1.1 ___ Manual
 - 1.2 ___ Sistemas ofimáticos (Microsoft Word, Access, etc.)
 - 1.3 ___ Aplicación informática
 - 1.4 ___ Otros

En caso de seleccionar la opción 1.4 ¿Cuáles?

2. ¿Son satisfechas todas las necesidades de obtención información de la institución?
 - 2.1. ___ Sí
 - 2.2. ___ No
 - 2.3. ___ Parcialmente

En caso de seleccionar las opciones 2.2 o 2.3 como respuesta anterior: ¿Qué deficiencias son notadas en el momento de adquisición de los datos? Describa brevemente estas deficiencias. ¿Cuentan con alguna estrategia para disminuir estas deficiencias anteriormente mencionadas?

3. Dentro de estos procesos de registro: ¿Qué documentación se genera en cada proceso?

4. ¿Forman parte de estas salidas los informes estadísticos?

4.1. si

4.2. no

5. ¿Los informes estadísticos son generados con la eficiencia y precisión que demanda la Cámara de Comercio? En caso de ser necesario puede seleccionar más de una opción.

5.1. Están acordes a las necesidades de la Cámara

5.2. Inexistencia de campos, variables, valores, etc.

5.3. Se generan incompletos

5.4. No cuentan con un formato estandarizado

5.5. No son almacenados en el formato adecuado

5.6. Existen diferencias en las letras de la documentación

5.7. Manejo de colores inadecuados

5.8. No cuentan con el logotipo de la entidad

5.9. No cuentan con la fecha

5.10. No cuentan con la dirección de la entidad

5.11. ____ No cuentan con la información completa del contacto (correo, página web, fax, etc.)

6. ¿Se ha comportado de manera satisfactoria la entrega de estos datos a cada una de las áreas para su posterior tramitación y análisis?

7. ¿Existe alguna documentación oficial que sea necesario generar en este procedimiento de registro?

8. De esta documentación final, ¿Se necesitan generar algunos cambios en la forma de representación de esta información?

9. ¿Cumplen estos reportes con la estructura plasmada en el Manual de Identidad de la Cámara de Comercio? En cualquier caso justifique.

9.1. ____ si

9.2. ____ no

10. Hasta el momento, si más de una de las partes involucradas en los procesos deben consultar un mismo reporte: ¿Se garantiza la rapidez en la entrega y la creación de varios reportes sin necesidad de espera prolongada? Justifique.

11. ¿Qué importancia le brinda a la Cámara de Comercio la generación de estos reportes como fuente de información para cada una de las áreas y la gerencia?



12. ¿Cómo valora el factor tiempo en su estrategia de trabajo para la generación de informes?

- 12.1. ____ Importante/Esencial
- 12.2. ____ Poco importante/Irrelevante
- 12.3. ____ Elemento normal

13. En el siguiente campo puede expresar otras consideraciones que estime necesarias en la realización de los reportes:

Anexo 2: Entrevista a Especialista del Modelo CMMI en la UCI

Nombre de la persona: _____

Cargo: _____

Departamento o área: _____

La UCI como entidad de desarrollo de software, certificó el Nivel II de CMMI (Capability Maturity Model Integration), siendo este el nivel básico para la gestión de proyectos de software. Sobre este nivel de capacidad responda:

1- ¿En qué año certificó la universidad el Nivel II de CMMI?

2- ¿En qué consiste el nivel dos de CMMI? Y ¿Qué significa alcanzar este nivel de capacidad?

3- ¿Cuáles son los beneficios de adquirir este nivel de madurez de desarrollo de software?

4- ¿Se puede considerar el Programa de Mejora un modelo, una metodología o un proceso de desarrollo de software?

5- ¿Por qué se define este Programa de Mejora en la universidad como guía de desarrollo de software?

6- ¿Qué plantea el programa de mejora en cuanto a la organización del desarrollo de software?

7- ¿Cómo queda definida la estructura del expediente de proyecto según el Programa de Mejora?

Anexo 3: Comparación de los sistemas estudiados

Parámetros	Dynamic Reports Generator	Jasper Report	PHP Reports Marker	Crystal Reports	Active Reports	Reporting Services Microsoft SQL
Tipo de software	Libre	Libre	Libre	Privativo	Privativo	Privativo
Multiplataforma	no	si	si	no	no	no
Reportes dinámicos	si	si	si	si	si	si
Personaliza al detalle	si	si	si	no	si	si
Escalabilidad	si	si	no	si	no	si
Salidas	PDF	DOC, PDF	PDF	PDF	PDF	PDF
Arquitectura	Modular	Librería	Extensión	Librería	Librería	Servidor
Genera Subreportes	si	si	si	si	si	si

Tabla 8: Tabla comparativa de los sistemas estudiados.

Anexo 4: Medición del Tamaño Operacional de la Clase

Parámetros	Evaluación	
	Categoría	Criterio
Responsabilidad	Baja	\leq Promedio
	Madia	Entre Promedio y $2 * \text{Promedio}$
	Alta	$> 2 * \text{Promedio}$
Complejidad de Implementación	Baja	\leq Promedio
	Madia	Entre Promedio y $2 * \text{Promedio}$
	Alta	$> 2 * \text{Promedio}$
Reutilización	Baja	$> 2 * \text{Promedio}$
	Madia	Entre Promedio y $2 * \text{Promedio}$
	Alta	\leq Promedio

Tabla 9: Cálculo de dominio del TOC

Anexo 5: Cálculo del valor de calidad según el modelo definido por la ISO.

Calidad Externa e Interna					
Característica	Peso %	Peso Real %	Subcaracterísticas	Peso %	Peso Real %
Funcionalidad	21		Aplicabilidad	39	
			Precisión	36	
			Interoperabilidad	7	
			Seguridad	8	
			Conformidad de funcionalidad	10	
Fiabilidad	22		Madurez (hardware/software/datos)	40	
			Tolerancia a fallos	36	
			Recuperabilidad (datos, proceso, tecnología)	14	
			Conformidad de fiabilidad	10	
Usabilidad	15		Entendibilidad	29	
			Facilidad de aprendizaje	23	
			Operabilidad	27	
			Atractividad	11	
			Conformidad de usabilidad	10	
Eficiencia	18		Comportamiento en el tiempo	41	
			Utilización de recursos	35	
			Conformidad de eficiencia	24	
Facilidad de Mantenimiento	16		Analizabilidad	21	
			Cambiabilidad	23	
			Estabilidad	21	
			Testeabilidad	27	
			Conformidad de facilidad de mantenimiento	8	
Portabilidad	8		Adaptabilidad	25	
			Instabilidad	25	

	Co-existencia	25	
	Reemplazabilidad	13	
	Conformidad de portabilidad	13	

Tabla 10: Modelo de calidad definido por la ISO.

Anexo 6: Descripción de las clases de implementación.

Para cada responsabilidad	
Nombre	Descripción
EDReporte	Contiene toda la información referente a los reportes como la plantilla Jasper, etc.
DSGenerico	Se encarga de la fabricación de los reportes.
FactoriaEDReporte	Se encarga de convertir de entidad dominio a entidades persistentes y viceversa.
GestorADReporte	Permite el acceso a las bases de datos de los reportes.
AccionReporteArgumentacion	Permite la creación del documento Argumentación con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCriterioJuntaEjecutiva	Permite la creación del documento Criterio de la Junta Ejecutiva de admisión como asociado con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCartaAlPresidente	Permite la creación del documento Carta al Presidente con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCertificacion	Permite la creación del documento Certificación con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteAsambleaGeneral	Permite la creación del documento Reporte de la Asamblea General con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCantidadEmpresasProvincias	Permite la creación del documento Cantidad de Empresas por Provincias con los datos de la

	empresa y permite la visualización del reporte correspondiente.
AccionReporteCantidadEmpresasSecciones	Permite la creación del documento Cantidad de Empresas por Secciones con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCantidadEmpresasSectores	Permite la creación del documento Cantidad de Empresas por Sectores con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCuotasAsociadosActivos	Permite la creación del documento Cuotas de Asociados Activos con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteDatosSitioWeb	Permite la creación del documento Datos del Sitio Web con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteEmpresasParaInvitacionesFIHAV	Permite la creación del documento Empresas para invitaciones de FIHAV con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteObservaciones	Permite la creación del documento Observaciones con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaInscripcion	Permite la creación del documento Licencia de inscripción con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaARenovar	Permite la creación del documento Licencia a renovar con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePorClasificadores	Permite la creación del documento Entidades por clasificadores con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePorActividadAutorizada	Permite la creación del documento Entidades por actividad autorizada con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteObjetoSocial	Permite la creación del documento Entidades por Objeto Social con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePagos	Permite la creación del documento Reporte de Pagos con los datos de la empresa y permite la

	visualización del reporte correspondiente.
AccionReporteLicenciaRenovacion	Permite la creación del documento Licencia de Renovación con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteAnexos	Permite la creación del documento Anexos con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaAVNOE	Permite la creación del documento Licencia de Agencia de Viajes Nacionales o Extranjera con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaAVR	Permite la creación del documento Licencia de Agencia de Viajes representada con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaRenovacionAVNOE	Permite la creación del documento Licencia de Renovación de Agencia de Viajes Nacionales o Extranjera con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteLicenciaRenovacionAVR	Permite la creación del documento Licencia de Renovación de Agencia de Viajes representada con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteVisualizarInscripcionesAVNOE	Permite la creación del documento de Inscripciones de Agencias de Viajes Nacionales o Extranjeras con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteVisualizarInscripcionesAVR	Permite la creación del documento de Inscripciones de Agencias de Viajes Representada con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCertificadoInscripcionIE	Permite la creación del documento Certificado de Inscripción de Importadora/Exportadora con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCancelacionFacultades	Permite la creación del documento Cancelación de las facultades con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteCertificacionTerceroExistencia	Permite la creación del documento Certificación a Tercero de Existencia con los datos de la empresa y permite la visualización del reporte correspondiente.

AccionReporteCertificacionTerceroNoExistencia	Permite la creación del documento Certificación a Tercero de no Existencia con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteVisualizarEDiferentesParametros	Permite la creación del documento Entidades por diferentes parámetros con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReporteVisualizarEActividadEconomica	Permite la creación del documento Entidades por actividad económica con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePlanillaAO-3	Permite la creación del documento Planilla AO-3 con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePlanillaCEC-4	Permite la creación del documento Planilla CEC-4 con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePlanillaRegistroTramites	Permite la creación del documento Planilla Registro de Trámites con los datos de la empresa y permite la visualización del reporte correspondiente.
AccionReportePlanillaPermisosTrabajo	Permite la creación del documento Planilla Permisos de Trabajo con los datos de la empresa y permite la visualización del reporte correspondiente.
AcciónReporteActa	Permite la creación del documento Acta con los datos de la empresa y permite la visualización del reporte correspondiente.

Tabla 11: Descripción de las clases implementadas.

Anexo 7: Diagrama lógico del Marco de Trabajo

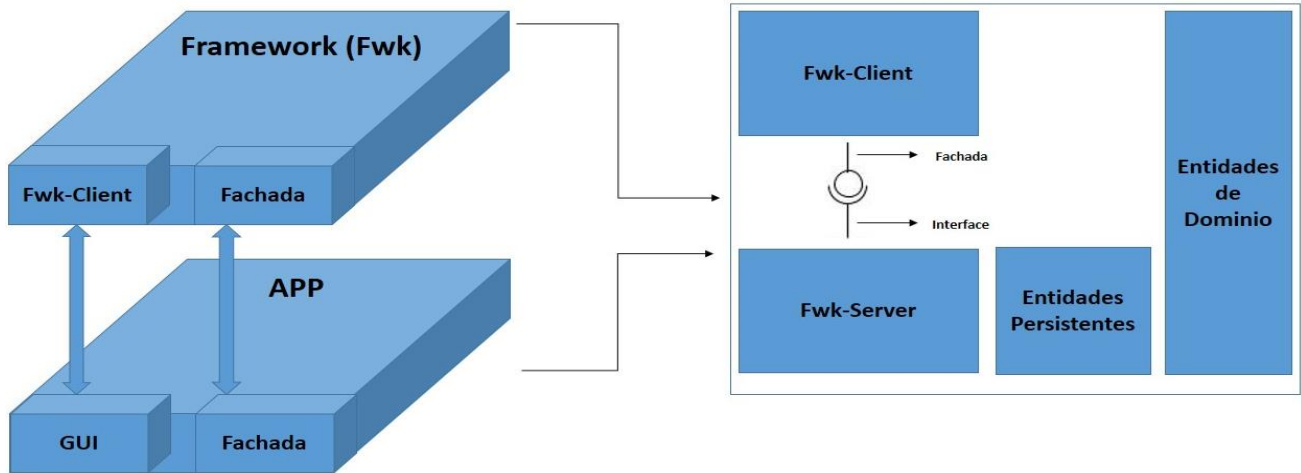


Imagen 8: Vista Lógica del Marco de Trabajo GEFORT

Anexo 8: Vista física de los estilos arquitectónicos empleados en la solución.

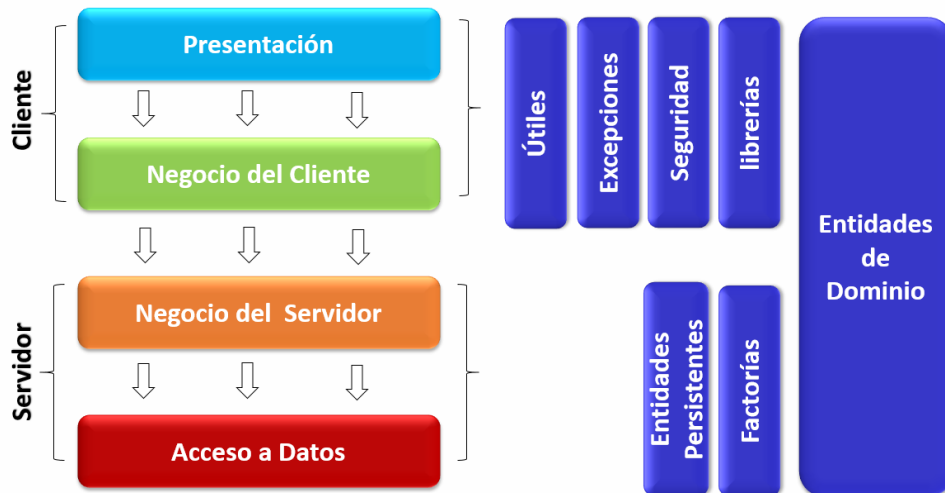


Imagen 9: Vista física de los estilos arquitectónicos empleados en la solución

Anexo 9: Cálculo del TOC para los atributos Responsabilidad y Complejidad de Implementación.

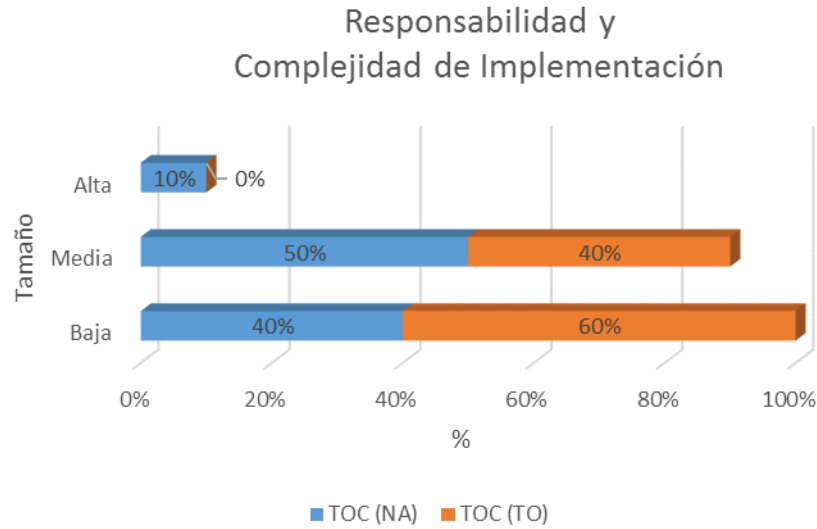


Imagen 10: Cálculo del TOC para los atributos Responsabilidad y Complejidad de Implementación

Anexo 10: Cálculo del TOC para el atributo Reutilización.

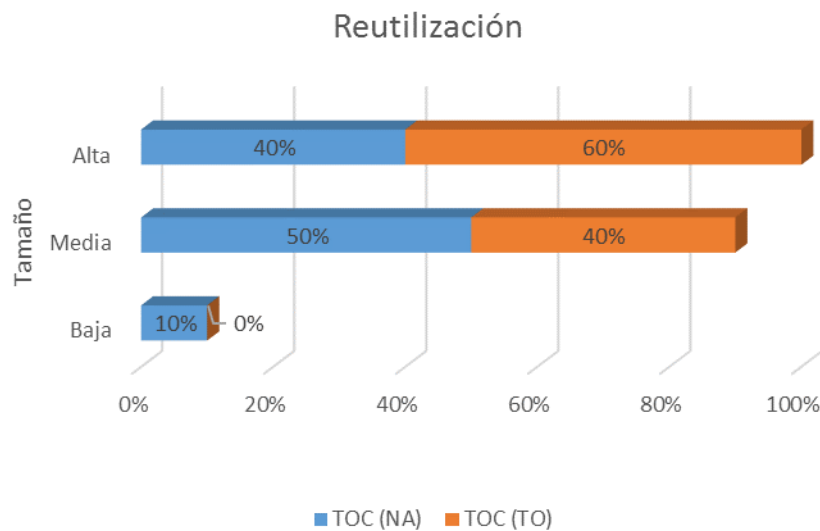


Imagen 11: Cálculo del TOC para el atributo Reutilización

Anexo 11: Modelo de Dominio del Módulo de Reportes.

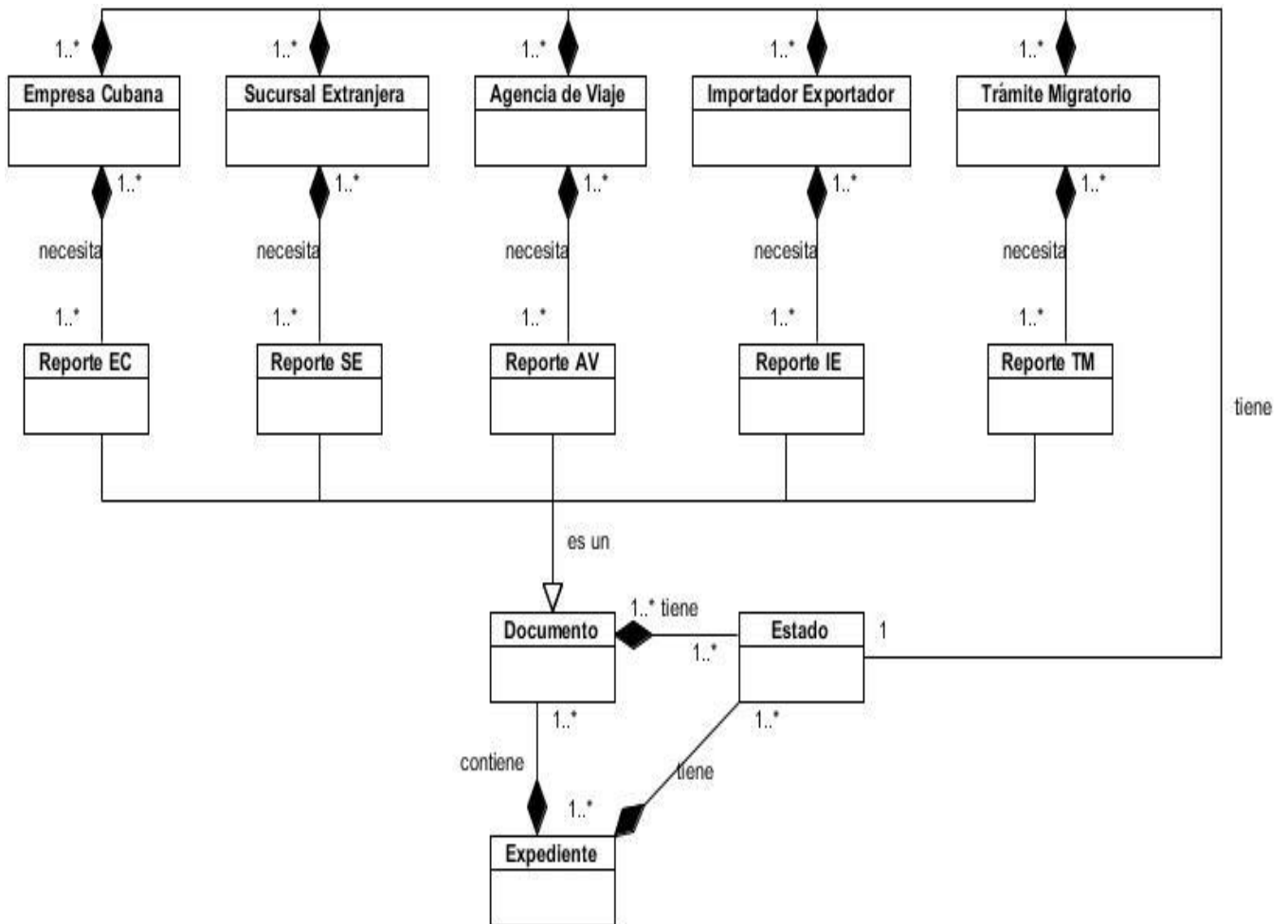


Imagen 12: Modelo de Dominio del Módulo de Reportes

Anexo 12: Diagrama de Clases del Diseño.

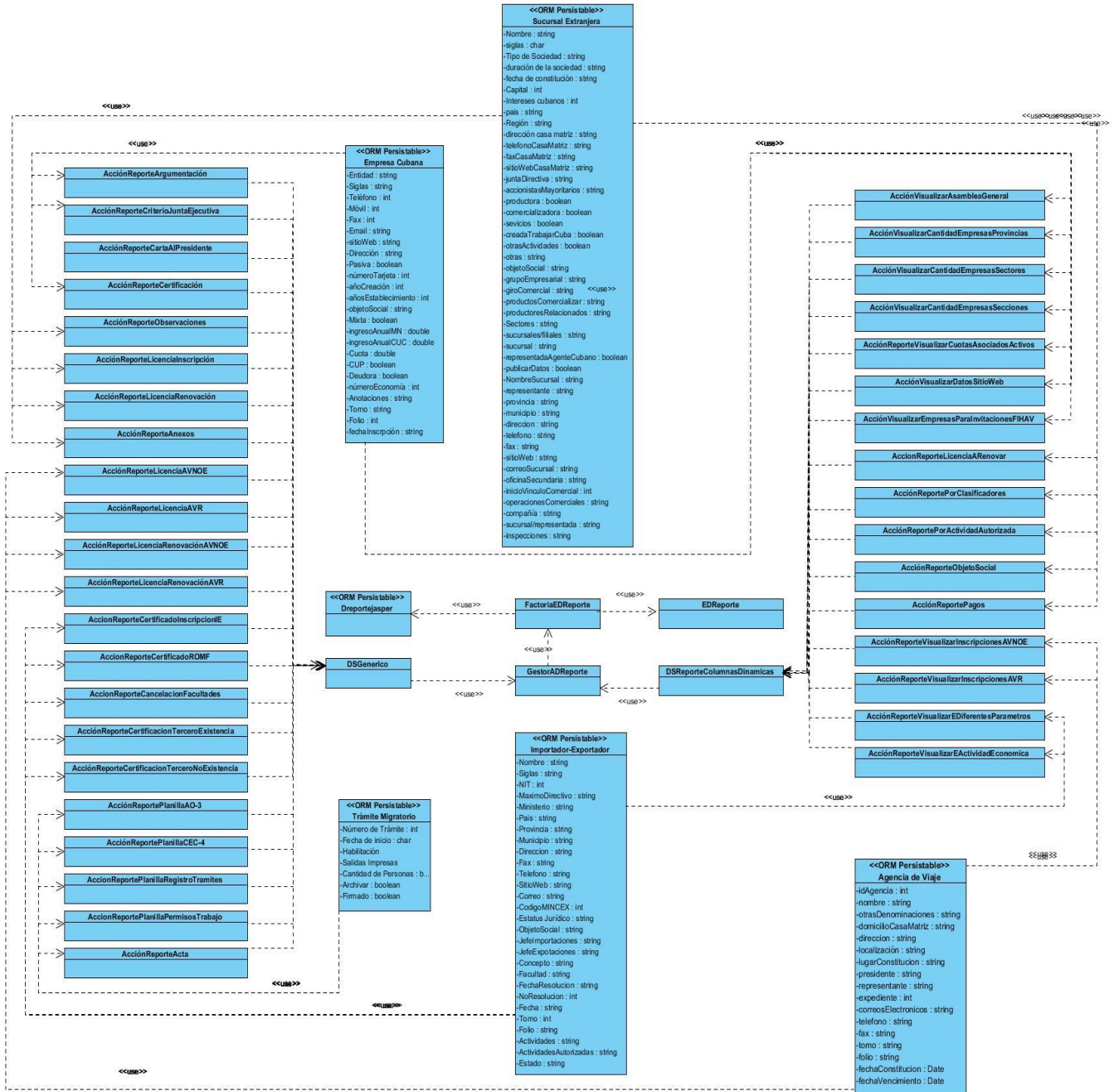


Imagen 13: Diagrama de Clases del Diseño

Anexo 13: Generar documento Criterios de la junta ejecutiva sobre admisión como asociado



The screenshot shows a window titled "Documento generado" with a standard Windows-style title bar. The main content area contains the following text and form elements:

- Logo of the Chamber of Commerce of the Republic of Cuba (a stylized 'C' with a hand).
- CAMARA DE COMERCIO DE LA REPUBLICA DE CUBA**
- Dirección de Formación y Membresía
- Criterios de la junta ejecutiva sobre admisión como asociado**
- Entidad: Empresa GEOCUBA Estudios Marinos
- Fields for "Presidente" (3 lines), "Vicepresidente" (2 lines), and "Secretario General" (2 lines).
- Cuota: 5000.00 MN
- Observaciones
- Aprobada:
- No aprobada:
- Acuerdo:
- Fecha:

At the bottom of the window, there are four buttons: "Editar plantilla", "Exportar", "Aceptar", and "Cancelar".

Imagen 14: Generar documento Criterios de la junta ejecutiva sobre admisión como asociado

Anexo 14: Diagrama de Componentes (General)

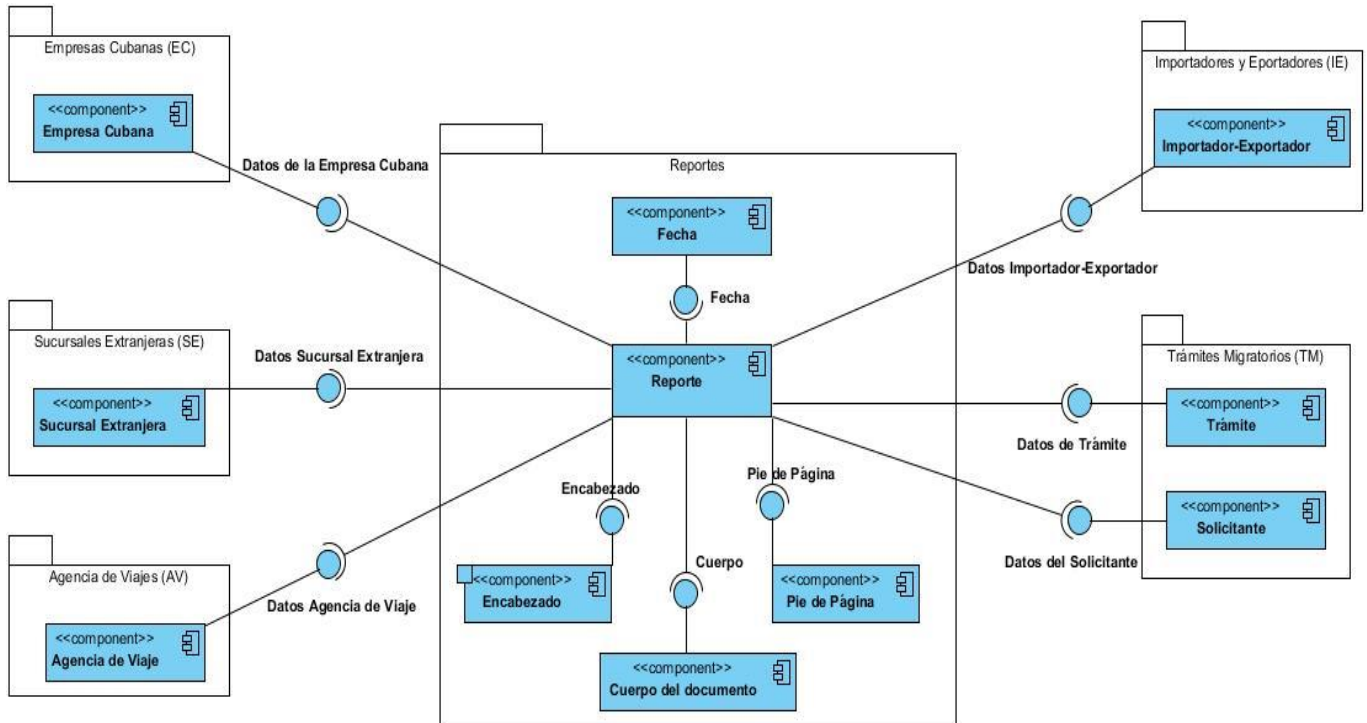


Imagen 15: Diagrama de Componentes General Módulo de Reportes

Anexo 15: Modelo de Datos Módulo de Reportes

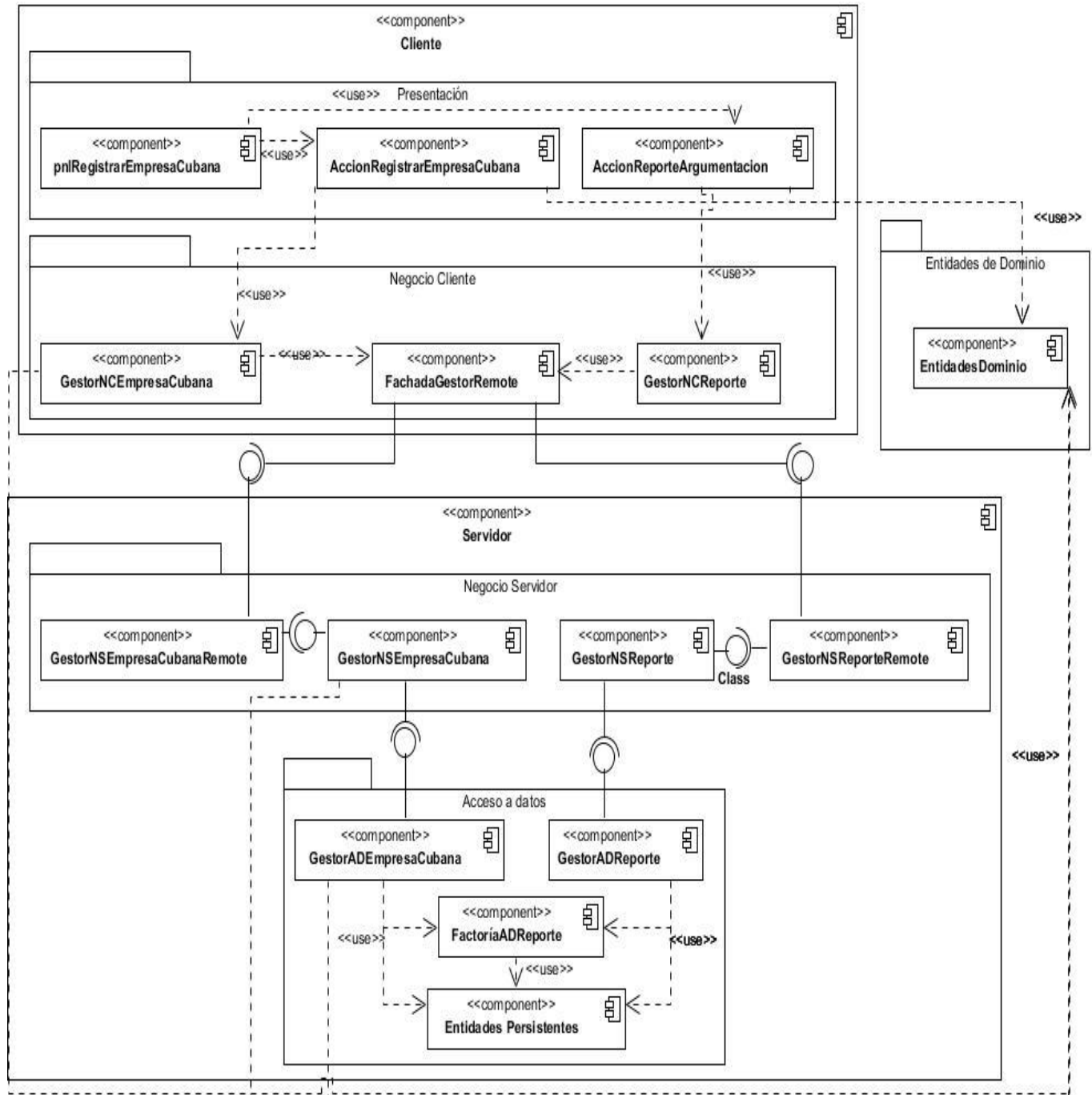


Imagen 16: Diagrama de Componentes del Módulo de Reportes (RF-1 Generar Argumentación)

Anexo 16: Modelo de Datos Módulo de Reportes

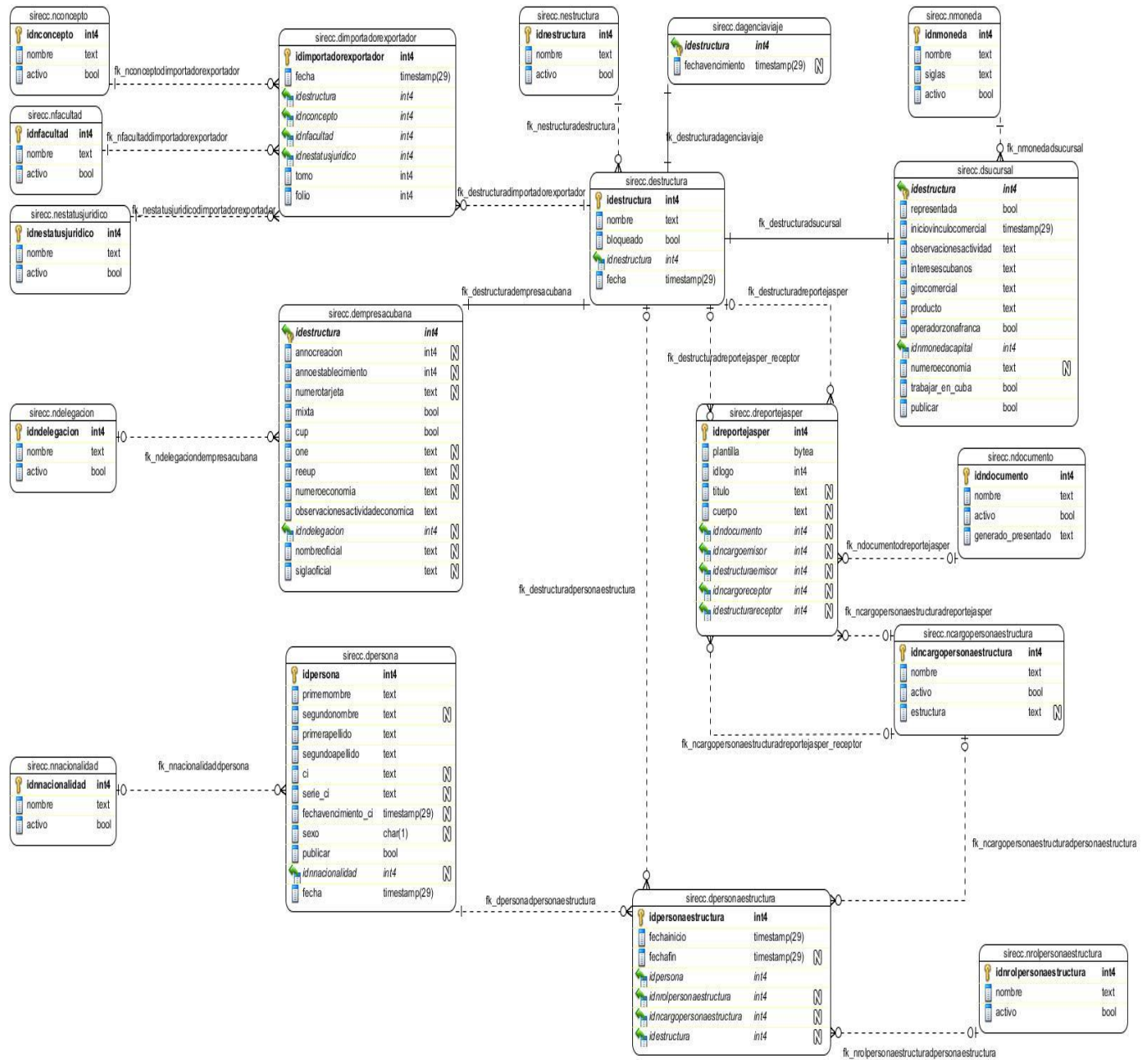


Imagen 17: Modelo de Datos General Módulo de Reportes

Anexo 17: Modelo de Datos Módulo de Reportes

Calidad Externa e Interna					
Característica	Peso %	Peso Real %	Subcaracterísticas	Peso %	Peso Real %
Funcionalidad	21	20,58	Aplicabilidad	39	39
			Precisión	36	35
			Interoperabilidad	7	7
			Seguridad	8	8
			Conformidad de funcionalidad	10	9
Fiabilidad	22	20,78	Madurez (hardware/software/datos)	40	40
			Tolerancia a fallos	36	35
			Recuperabilidad (datos, proceso, tecnología)	14	14
			Conformidad de fiabilidad	10	10
Usabilidad	15	13,05	Entendibilidad	29	25
			Facilidad de aprendizaje	23	20
			Operabilidad	27	25
			Atractividad	11	9
			Conformidad de usabilidad	10	10
Eficiencia	18	18	Comportamiento en el tiempo	41	41
			Utilización de recursos	35	35
			Conformidad de eficiencia	24	24
Facilidad de Mantenimiento	16	15,52	Analizabilidad	21	21
			Cambiabilidad	23	23
			Estabilidad	21	20
			Testeabilidad	27	25
			Conformidad de facilidad de mantenimiento	8	8
Portabilidad	8	8	Adaptabilidad	24	24
			Instabilidad	25	25
			Co-existencia	25	25
			Reemplazabilidad	13	13
			Conformidad de portabilidad	13	13

Profesora
Marina Salcedo Arce

Total → 96,93
 % → 96,93

Imagen 18: Evaluación de los atributos de calidad por los clientes.

GLOSARIO

Calidad: contempla dimensiones de evaluación que afectan la satisfacción de las necesidades de las personas. (Torres, y otros, 2008)

Escalabilidad: propiedad deseable de un sistema, se define como la facilidad con que una solución puede adaptarse a las necesidades a medida que surjan, indica su habilidad para reaccionar y adaptarse sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida.

Patrón: es un par problema/solución con nombre que se puede aplicar en nuevos contextos, con consejos acerca de cómo aplicarlo en nuevas situaciones y discusiones sobre sus compromisos.

Proceso de desarrollo de software: un conjunto de actividades y resultados asociados que producen un producto de software. Un proceso define ¿quién? está haciendo ¿qué?, ¿cuándo?, y ¿cómo? alcanzar un objetivo determinado. (Sommerville, 2005)

Variable: Es un hecho, proceso, fenómeno, atributo, propiedad, concepto que puede cambiar cuantitativa o cualitativamente.

Variable Dependiente: Es el consecuente o efecto que cambia por influencia de la variable independiente.

Variable Independiente: Es el antecedente o causa que genera cambios en otra variable.