

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 9



**TRABAJO DE DIPLOMA PARA OPTAR POR EL
TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS**



**TÍTULO: SUBSISTEMA PARA LA ADMINISTRACIÓN DE
METADATOS GEOLÓGICOS EN LA OFICINA NACIONAL DE
RECURSOS MINERALES.**

AUTORES: Gerdys Ernesto Jiménez Moya
Randy Ramírez Leal

TUTOR: Ing. Eylín Hernández Luque

CO-TUTOR: Lic. Osmel de la Cruz Amador

CONSULTORES: Héctor Rodríguez
César Rosales

Ciudad de La Habana, julio del 2008
"Año 50 de la Revolución"

DEDICATORIA

De Gerdys

En especial a mis padres, pues este sueño es también de ellos.

A mi hermana con mucho cariño.

A mi novia con mucho amor.

A mi familia en general y a mis amigos.

De Ramdy

A Dios.

Por haberme permitido llegar hasta aquí.

A mis padres.

Por haberme dado todo su apoyo y amor hasta en los momentos más difíciles.

A mi familia.

Por brindarme su apoyo incondicional y confianza.

A todos mis amigos.

Que me han ayudado personal, profesional y espiritualmente.

AGRADECIMIENTOS

De Gerdys:

A mis padres que con tanto amor, cariño, apoyo, dedicación, confianza, esfuerzo y mucho sacrificio sobre todo..., me han guiado por el difícil camino de la vida.

A Clari, por ser mí mejor hermana, por todo su cariño y por mortificarme demasiado.

A Elianys, mi amorcito, por todo su amor, por compartir tanto tiempo conmigo, por ser mi confidente, por cuidarme mucho y por quererme tanto.

A mis tíos, en especial a Emilio y a El Bala; a mis abuelos y a mi familia en general.

A Lili por su ayuda infinita.

A Eylín, nuestra tutora, por su ayuda y apoyo ilimitado, por ser además nuestra amiga.

A Felito y Lidia por acordarse de mí y por portarse como una familia.

A mis amigos Enrique, Lainer, Sol, Ramdy, Yorji, José Luis, Dasiel y a todos mis colegas de aula por compartir todo este tiempo juntos.

En fin a todo el que de una forma u otra me ha apoyado para llegar hasta aquí.

De Ramdy:

A ti Señor Jesús, mi fiel amigo, por haber hecho que esté en paz en medio de la tormenta, por haberme levantado y cuidado con el poder de tu amor, te doy las gracias y te agradezco que me hayas permitido llegar hasta aquí, sin ti, nada hubiera sido posible.

A mi madre y a mi padre, que con tanto amor y paciencia han sabido guiarme a lo largo de estos años, gracias por su cariño, su confianza, su esfuerzo, su optimismo y su apoyo incondicional. Con inmenso amor les dedico mi trabajo.

A mi abuelita Elisa, porque su amor, preocupación y apoyo han sido inmensamente grandes. Gracias abuela.

A mis tíos Mony y Luis, a ustedes que han sido mis hermanos mayores e inspiración para mi vida.

A mi tutora Eylin, por su ayuda incansable y apoyo incondicional en los momentos que más la necesitamos.

A todos mis amigos, en especial Andy Marcos Ortega, Yandy Reyes, Sandy Puig, Gerdys E. Jiménez, Anamary Alcantara, Dasiel Sanchez, Odiel Estrada y Yuniel Torres, y todos mis compañeros de apto por su apoyo y amistad.

DECLARACIÓN DE AUTORÍA

Ciudad de la Habana, Junio del 2008

“Año 50 de la Revolución”

Nosotros, Gerdys Ernesto Jiménez Moya y Ramdy Ramírez Leal, estudiantes de la Universidad de las Ciencias Informáticas (UCI), declaramos que somos los únicos autores de la presente investigación titulada: “SUBSISTEMA PARA LA ADMINISTRACIÓN DE METADATOS GEOLÓGICOS EN LA OFICINA NACIONAL DE RECURSOS MINERALES” y autorizo a hacer uso de la misma en su beneficio a la UCI y a cualquier otro centro educacional.

Y para que así conste, firmamos la presente:

Gerdys Ernesto Jiménez Moya

Autor

Ramdy Ramírez Leal

Autor

Ing. Eylín Hernández Luque

Tutor

DATOS DE CONTACTO

Síntesis de la Tutora Ing. Eyllin Hernández Luque

Profesión: Ingeniero en Ciencias Informáticas

Categoría docente: Adiestrado

Años de graduado: 1

Responsabilidad: Líder del Polo Geoinformática de la Facultad 9

Síntesis del Co-tutor Lic. Osmel de la Cruz Amador

Profesión: Graduado en Ciencias de la Computación.

Categoría docente: Adiestrado

Años de graduado: 2

Responsabilidad: Jefe del proyecto Metadatos del Polo Geoinformática.

Síntesis del Consultante Ing. Héctor Rodríguez

Profesión: Dr. En Ciencias Geológicas

Categoría docente: Profesor Titular

Años de graduada: 30

Responsabilidad: Grupo de Desarrollo y Geomática. Oficina Nacional de Recursos Minerales

Síntesis del Consultante Ing. César Rosales

Profesión: Ingeniero Geofísico

Categoría docente: No tiene

Años de graduada: 31

Responsabilidad: Jefe del Grupo de Servicios Informáticos de la Empresa Geominera Oriente.

OPINIÓN DEL TUTOR

Universidad de las Ciencias Informáticas

Tesis para optar por el título: Ingeniero en Ciencias Informáticas

Título: “Subsistema para la administración de metadatos geológicos en la Oficina Nacional de Recursos Minerales”

Autores: Gerdys Ernesto Jiménez Moya.

Ramdy Ramírez Leal.

Tutora: Ing. Eylín Hernández Luque

INFORME DE TUTORÍA

Hoy día, el mundo está en constante cambio y evolución tecnológica; nuestro país se ha incorporado a dicho desarrollo informático, sumándose todas las Entidades y Ministerios, entre los que se encuentra el Ministerio de la Industria Básica (MINBAS) y como oficina adjunta a la misma, la Oficina Nacional de Recursos Minerales (ONRM). Esta es una de las razones que permite expresar que la investigación desarrollada presenta actualidad y es de vital importancia para el mejoramiento de la gestión del conocimiento geológico en Cuba.

Los diplomantes Gerdys Ernesto Jiménez Moya y Ramdy Ramírez Leal, han demostrado responsabilidad y compromiso en el desarrollo de la investigación científica que por producción se les encomendó.

Demonstraron laboriosidad por lo que se cumplió con éxito el objetivo trazado en la investigación facilitando la captación de ficheros de metadatos geológicos. Los estudiantes mostraron independencia absoluta en cada fase de la tarea a realizar, logrando resultados satisfactorios. Además los diplomantes han trabajado de manera organizada y planificada, dando muestras de poseer una buena preparación metodológica y un gran sentido del compromiso laboral.

El análisis bibliográfico es actual y fundamenta correctamente la necesidad del estudio realizado. El diseño de la investigación se ajusta a la problemática que se aborda, existiendo correspondencia entre el problema a desarrollar, los objetivos planteados así como la idea a defender, lo que demuestra dominio del método científico. El documento realizado posee una buena organización; está basado en métodos investigativos científicamente aprobados. Es capaz de llevar al lector la información referida al contenido y a los objetivos perseguidos con su trabajo. La modelación del análisis y diseño de la aplicación cumple con el objetivo trazado, por lo cual brindan beneficios sociales y económicos para la gestión del conocimiento geológico en la Oficina Nacional de Recursos Minerales.

Por todo lo anteriormente expresado, considero que los estudiantes están aptos para ejercer como Ingeniero en Ciencias Informáticas y propongo que se le otorgue al Trabajo de Diploma la calificación de 5 puntos.

Ing. Eylín Hernández Luque
Tutora

Julio del 2008

Fecha

RESUMEN

La Oficina Nacional de Recursos Minerales (ONRM) es una de las entidades pertenecientes al Ministerio de la Industria Básica (MINBAS) y es la institución encargada de mantener el control de los recursos minerales en el país.

Con el objetivo de informatizar el conocimiento geológico referente a nuestro territorio nacional y con la necesidad de desarrollar aplicaciones capaces de gestionar toda esta información, la ONRM lleva a cabo junto a la Universidad de las Ciencias Informáticas (UCI) el Programa Nacional de Informatización del Conocimiento Geológico (PICG). El conocimiento geológico cubano se encuentra almacenado en varias bases de datos y en diversos formatos por todo el país, esto provoca gran dificultad para gestionar esta información, pues en la actualidad no existe una aplicación que permita administrar y proveer este conocimiento.

Debido a esto, en esta investigación, se propone el uso de metadatos geológicos con el fin de documentar toda esta información, se exponen los resultados del estudio realizado sobre la captación de ficheros de metadatos, así como el análisis y diseño de un sistema informático capaz de realizar la captación de ficheros de metadatos geológicos basado en los conceptos de software libre.

PALABRAS CLAVES

-  Metadatos
-  Geología
-  Entidades Geológicas
-  Sistema
-  Aplicación Web
-  Captación

ABSTRACT

The National Office of Mineral Resource (ONRM) is one of the organizations that takes part of The Ministry of the Basic Industry (MINBAS) and it is the institution in charge of maintain the control of the mineral resources in the country.

With the aim of computerizing the geologic knowledge referring to our national territory and with the need to develop some applications able to manage all this information, the ONRM carries out with the University of Informatics Sciences (UCI) The National Program of Computerization of Geologic Knowledge (PICG). The Cuban geologic knowledge is stored in several data bases and established on several formats all over the country, this brings about great difficulties to manage this information, because of for the time being does not exist an application that allows to administer and to provide all this knowledge.

Due to this, this investigation proposes, the use of geologic metadata with the purpose of to document all this information, the results of the study accomplished about the pickup of files of metadata are exposed in this project, as well as the analysis and design of an informatics system able to carry out the pickup of files of geologic metadata based on the concepts of free software.

KEYWORDS

- ✚ Metadata
- ✚ Geologies
- ✚ Entities geological
- ✚ System
- ✚ Web Application
- ✚ Captation

ÍNDICE DE TABLAS

<i>Tabla 1 Actor del Negocio</i>	52
<i>Tabla 2 Trabajadores del Negocio</i>	52
<i>Tabla 3 Descripción textual CUN Revisar Metadato</i>	54
<i>Tabla 4 Descripción textual CUN Archivar Metadato</i>	55
<i>Tabla 5 Actores del Sistema</i>	61
<i>Tabla 6 Descripción CUS "Autenticar"</i>	63
<i>Tabla 7 Descripción CUS "Brindar información sobre los servicios del sistema"</i>	64
<i>Tabla 8 Descripción CUS "Gestionar Metadato"</i>	66
<i>Tabla 9 Descripción CUS "Actualizar servicio de información"</i>	67
<i>Tabla 10 Descripción CUS "Gestionar Logs del Sistema"</i>	69

ÍNDICE DE FIGURAS

<i>Figura 1 Metadato. Secciones y modelos de documentación</i>	10
<i>Figura 2 Disciplinas a través de las iteraciones</i>	32
<i>Figura 3 Modelo de Casos de Uso del Negocio</i>	53
<i>Figura 4 Diagrama de actividades para CUN Revisar Metadato</i>	56
<i>Figura 5 Diagrama de actividades del CUN Archivar Metadato</i>	56
<i>Figura 6 Modelo de Objetos</i>	57
<i>Figura 7 Diagrama de Casos de Uso del Sistema</i>	61
<i>Figura 8 Diagrama de Paquetes</i>	71
<i>Figura 9 El flujo de trabajo de Symfony</i>	74
<i>Figura 10 Paquete Brindar Información. Caso de uso "Autenticar"</i>	75
<i>Figura 11 Paquete Brindar Información. Caso de uso "Brindar Información"</i>	75
<i>Figura 12 Paquete Brindar Información. Caso de uso "Brindar Información". Sección "</i>	76
<i>Figura 13 Paquete Administración. Caso de uso "Gestionar metadato"</i>	76
<i>Figura 14 Paquete Administración. Caso de uso "Gestionar metadato". Sección "Importar Archivo"</i>	77
<i>Figura 15 Paquete Administración. Caso de uso "Gestionar metadato". Sección "Emitir Reporte"</i>	77
<i>Figura 16 Paquete Administración. Caso de uso "Actualizar servicio de información"</i>	78
<i>Figura 17 Paquete Administración. Caso de uso "Actualizar servicio de información"</i>	78
<i>Figura 18 Paquete Administración. Caso de uso "Actualizar servicio de información"</i>	79
<i>Figura 19 Paquete Administración. Caso de uso "Gestionar Logs". Sección "Mostrar Logs"</i>	79
<i>Figura 20 Paquete Administración. Caso de uso "Gestionar Logs". Sección "Salvar Logs"</i>	80
<i>Figura 21 Paquete Administración. Caso de uso "Gestionar Logs". Sección "Eliminar Logs"</i>	80
<i>Figura 22 Diagrama de clases persistentes</i>	83
<i>Figura 23 Modelo de datos</i>	84
<i>Figura 24 Modelo de Despliegue</i>	85
<i>Figura 25 Modelo de Componentes. CU "Autenticar"</i>	86
<i>Figura 26 Modelo de Componentes. CU "Brindar Información"</i>	86
<i>Figura 27 Modelo de Componentes. CU "Gestionar Metadato"</i>	87
<i>Figura 28 Modelo de Componentes. CU "Actualizar servicio de Información"</i>	88
<i>Figura 29 Modelo de Componentes. CU "Gestionar Logs"</i>	89
<i>Figura 30 DCA. Caso de uso "Autenticar"</i>	94
<i>Figura 31 DCA. Caso de uso "Brindar Información"</i>	94

<i>Figura 32 DCA. Caso de uso "Gestionar Metadato"</i>	95
<i>Figura 33 DCA. Caso de uso "Actualizar servicio de Información"</i>	95
<i>Figura 34 DCA. Caso de uso "Gestionar Logs"</i>	96
<i>Figura 35 Diagrama de Colaboración. CU "Autenticar"</i>	97
<i>Figura 36 Diagrama de Secuencia. CU "Autenticar"</i>	97
<i>Figura 37 Diagrama de Colaboración. CU "Brindar Información". Asistencia técnica</i>	98
<i>Figura 38 Diagrama de Secuencia. CU "Brindar Información". Asistencia técnica</i>	98
<i>Figura 39 Diagrama de Colaboración. CU "Brindar Información". Servicio de preguntas</i>	99
<i>Figura 40 Diagrama de Secuencia. CU "Brindar Información". Servicio de preguntas</i>	99
<i>Figura 41 Diagrama de Colaboración. CU "Gestionar Metadato". Actualizar Metadato</i>	100
<i>Figura 42 Diagrama de Secuencia. CU "Gestionar Metadato". Actualizar Metadato</i>	100
<i>Figura 43 Diagrama de Colaboración. CU "Gestionar Metadato". Emitir Reporte</i>	101
<i>Figura 44 Diagrama de Secuencia. CU "Gestionar Metadato". Emitir Reporte</i>	101
<i>Figura 45 Diagrama de Colaboración. CU "Gestionar Metadato". Importar archivo</i>	102
<i>Figura 46 Diagrama de Secuencia. CU "Gestionar Metadato". Importar archivo</i>	102
<i>Figura 47 Diagrama de Colaboración. CU "Actualizar servicio de Información"</i>	103
<i>Figura 48 Diagrama de Secuencia. CU "Actualizar servicio de Información"</i>	103
<i>Figura 49 Diagrama de Colaboración. CU "Actualizar servicio de Información"</i>	104
<i>Figura 50 Diagrama de Secuencia. CU "Actualizar servicio de Información"</i>	104
<i>Figura 51 Estructura de clases generadas por Symfony. Modelo genérico</i>	105
<i>Figura 52 Acción relacionadas con el paquete Modelo</i>	105

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: Fundamentación Teórica.....	8
1.1. Introducción.....	8
1.2. Conceptos asociados al dominio del problema.....	8
1.2.1. Datos geológicos.....	8
1.2.2. Metadatos.....	8
1.2.3. Metadatos geológicos.....	9
1.2.4. Sistema.....	11
1.2.5. Sistema informático.....	12
1.2.6. Aplicaciones Web.....	13
1.3. Objeto de Estudio.....	15
1.3.1. Descripción General.....	15
1.3.2. Descripción actual del dominio del problema.....	23
1.3.3. Situación Problemática.....	24
1.4. Análisis de otras soluciones existentes.....	25
1.5. Conclusiones parciales.....	26
CAPITULO 2: Tendencias y tecnologías actuales a desarrollar.....	27
2.1. Introducción.....	27
2.2. Las tecnologías de la Información y las Comunicaciones.....	27
2.2.1. Principales aportes de las TICs.....	28
2.3. Lenguaje Unificado de Modelado (UML).....	29
2.4. Metodologías de desarrollo de software.....	30
2.4.1. Fundamentación de la metodología utilizada (RUP).....	31
2.5. Modelo Cliente – Servidor.....	33
2.6. Arquitectura en Capas.....	34
2.7. Tecnologías del lado del cliente.....	35
2.7.1. HTML.....	35
2.7.2. CSS.....	36
2.7.3. JavaScript.....	36
2.8. AJAX.....	37
2.8.1. Ventajas del AJAX.....	38
2.8.2. Desventajas del AJAX.....	38
2.9. XML.....	39
2.10. Lenguajes de programación del lado del servidor.....	39
2.10.1. Fundamentación del lenguaje del lado del servidor a utilizar.....	40
2.11. Servidor Web Apache.....	43
2.12. Sistemas de Gestión de Bases de Datos.....	44
2.12.1. Fundamentación de la selección de PostgreSQL como SGBD.....	44
2.13. Herramientas a utilizar.....	46
2.13.1. Frameworks: Symfony.....	46
2.13.2. Visual Paradigm.....	47
2.13.3. Eclipse.....	48
2.13.4. Subversion.....	49
2.13.5. DotProyect.....	49
2.13.6. Gestor de referencias: EndNote.....	50
2.14. Conclusiones parciales.....	50

CAPÍTULO 3: Presentación de la solución propuesta	51
3.1. Introducción	51
3.2. Modelo de Negocio.	51
3.2.1. Reglas generales del Negocio.....	51
3.2.3. Modelo de Casos de Uso del Negocio.....	53
3.2.4. Descripción textual de los Casos de Uso del Negocio.....	53
3.2.5. Diagrama de actividades para cada CU del Negocio.....	56
3.3. Requisitos del Sistema.....	57
3.3.1. Requerimientos Funcionales.....	57
3.3.2. Requerimientos No Funcionales.....	58
3.4.1. Descripción de los actores del sistema.....	61
3.4.2. Diagrama de Casos de Usos del Sistema	61
3.4.3. Descripción de los Casos de Uso del Sistema (CUS).....	62
3.5. Conclusiones parciales.....	69
CAPÍTULO 4: Construcción de la solución propuesta.....	70
4.1. Introducción	70
4.2. Modelo de Análisis	70
4.2.1. Diagrama de Clases del Análisis	70
4.2.2. Diagramas de Interacción.....	71
4.3. Modelo de Diseño.....	71
4.3.1 Diagrama de clases de Diseño.....	71
4.4. Principios de Diseño.....	81
4.4.1. Diseño de la interfaz de la aplicación.....	81
4.5. Diseño de la Base de Datos.....	83
4.6. Modelo de Despliegue.....	85
4.7. Modelo de Implementación.....	85
4.8. Conclusiones parciales.....	89
CONCLUSIONES.....	90
RECOMENDACIONES.....	91
REFERENCIAS BIBLIOGRÁFICAS.....	92
ANEXOS	94
Anexo 1. Diagrama de clases del Análisis.....	94
Anexo 2. Diagramas de Interacción del Análisis. Colaboración y Secuencia.....	97
Anexo 3. Estructura de clases generadas por Symphony. Modelo genérico.....	105
Anexo 4. Generalidades sobre la relación entre las Action y el paquete Modelo.....	105
Anexo 5. Entrevista realizada.....	106
GLOSARIO.....	107

INTRODUCCIÓN

En la actualidad, debido al enorme esfuerzo realizado por el hombre para mejorar su forma de vida y la sociedad, ha impulsado la innovación tecnológica, el desarrollo de la economía, la cultura y la ciencia en general. Muchos de estos avances han sido dedicados a la investigación geológica, ejemplo de ello lo constituyen:

Los sistemas de información geográfica, herramientas que cada día son más utilizadas en el ámbito del manejo de los recursos naturales por sus múltiples aplicaciones debido a:(Ahumada 2007)

- ✚ La capacidad de integración de información espacial y de otro tipo dentro de un solo sistema.
- ✚ Ofrece una consistente estructura para el análisis de los datos geográficos.
- ✚ Permite manejar mapas y otros tipos de información espacial en forma digital.
- ✚ La capacidad para realizar conexiones entre actividades basadas en la proximidad geográfica.

Con dicha herramienta ha sido posible, entre otras cosas, la elaboración rápida de planes de ordenamiento geológico del territorio que sirven para evaluar, normar y programar el uso de los recursos naturales, para proponer acciones concretas de manejo de dichos recursos, proporcionando de esta forma grandes avances científicos en la rama de la Geología.(Ahumada 2007)

Se plantea que la Geología es la ciencia que se encarga del estudio del subsuelo en cuanto a su composición, estructura, evolución, recursos minerales; utiliza las diferentes teorías que explican la evolución física del planeta para:

- ✚ Dar uso correcto a los recursos minerales y las aguas subterráneas.
- ✚ Manejar la peligrosidad sísmica y el riesgo geológico.
- ✚ Evaluar los suelos para la construcción.

Esta ciencia basa sus métodos de estudio en una serie de aproximaciones sucesivas expresada por una escala, partiendo de lo más general a lo particular, desde el planeta y sus grandes estructuras (continentes) hasta tipo de roca o mineral como posibles objetos de la investigación.(Diseño 2006)

El sector geológico en Cuba en estos últimos años ha tenido un desarrollo considerable y sostenido, aportando importantes ganancias al desarrollo económico. El objetivo fundamental de dicho sector, lo constituye la inclusión de aquellas actividades que aunque emplean el conocimiento geológico este no

forma parte del activo fundamental de su actividad productiva o de servicio, pero la gran información que brinda puede ser usada por múltiples entidades de la economía, la ciencia y el gobierno. Con la llegada del ALBA estas potencialidades se han multiplicado, pues se han abierto nuevos mercados y posibilidades de inversión extranjera, así como otros tipos de convenios con los países pertenecientes a este acuerdo, conocido también como Oportunidad de Inversión con Capital Extranjero.

La Oportunidad de Inversión con Capital Extranjero se refiere a aquellos proyectos de inversión que responden a las necesidades de las empresas nacionales cuyos requerimientos fundamentales estén relacionados con el aporte de financiamiento a largo plazo, mercado externo y/o tecnología y que se prevén realizar por alguna modalidad de negocios previstas en la ley 77 o el acuerdo 5290 .(Cuba 1995) Todos estos procesos están bajo el mando del Ministerio de la Industria Básica (MINBAS).

El Ministerio de la Industria Básica (MINBAS) es la organización estatal cubana que tiene la responsabilidad de tres importantes sectores de la economía de nuestro país, la Energía, la Geología y la Minería así como la Química Básica. Está formada por siete grupos empresariales QUIMEFA, CUBAPETROLEO, UNION ELECTRICA, CUBANIQUEL, GEOMINSAL, CEMENTO, GEIQ que reúnen 147 empresas con 100000 trabajadores, de estos 14 600 poseen nivel universitario y 25 000 son mujeres. Además tienen entidades independientes adscriptas al ministerio como el Perfilactorio Nacional Obrero, la Escuela Superior de la Industria Básica y Politécnico (CNCI), Tecnomática, la Empresa de Servicios (SERVIBASICA) y la Oficina Nacional de Recursos Minerales (ONRM). Algunos de los objetivos fundamentales de esta institución son:(MINBAS 2007)

- ✚ Generación por centrales eléctricas y grupos electrógenos, transmisión, distribución y comercialización de la energía eléctrica.
- ✚ La producción y comercialización de combustibles derivados del petróleo y lubricantes, así como la comercialización mayorista del alcohol combustible doméstico, la búsqueda, exploración y explotación de petróleo y gas.
- ✚ Investigación geológica, búsqueda, explotación y procesamiento de minerales sólidos, extracción, producción y comercialización mayorista de minerales sólidos, sal y sus derivados, investigación geológica y búsqueda de aguas minero medicinal.
- ✚ Investigación del reconocimiento de patrones, la minería de datos y otras tecnologías de avanzada según los requerimientos y exigencias del desarrollo del país.

Esta organización, para su dirección empresarial, se agrupa en sus siete Uniones de Empresas, que integran bajo su responsabilidad trabajos de producción, investigación, proyectos, mantenimiento, construcción, comercialización interna, importación de materias primas e insumos y exportación de productos terminados. (MINBAS 2007)

Las Entidades Geológicas son el conjunto de instituciones autorizadas para la generación, administración y uso fundamental del conocimiento geológico. Esas instituciones son:

- ✚ La Oficina Nacional de Recursos Minerales (ONRM).
- ✚ Centros de investigación (Institutos Científicos y Universidades).
- ✚ Empresas que realizan Investigaciones Geológicas para diferentes fines.
- ✚ Empresas Mineras.
- ✚ Empresas de perforación y extracción de Petróleo.(Diseño 2006)

La ONRM es una de las empresas pertenecientes al MINBAS y muy relacionada con el sector geológico cubano, esta constituye la autoridad minera en el país. Fue constituida en el año 1995 mediante la Ley de Minas. Tiene como principales funciones el control y fiscalización del Proceso Concesionario en Cuba, la protección de los recursos minerales y de hidrocarburos, el control de la explotación racional de los recursos minerales y la preservación del medio ambiente durante el desarrollo de actividades mineras. (Cuba 1995)

Además en la ONRM se maneja un elevado número de información y de datos que hacen referencia al desarrollo geológico del país de sus últimos cuatro siglos, en su mayoría archivados en papel o en varias bases de datos con diversos formatos, lo que provoca:

- ✚ Pérdida de información.
- ✚ Repetición y errores en la información manejada.
- ✚ Atrasos en la localización y procesamiento de la información requerida por la ONRM y el resto de las entidades nacionales subordinadas a ella.

Las empresas, tanto nacionales como internacionales, interesadas en establecer relaciones de negocio en el país en la esfera geológica se les dificulta enormemente este proceso, pues no poseen fácil acceso a la información referente a los datos geológicos, además las vías existentes para establecer los negocios son poco factibles y se les hacen demasiado costosos los trámites para realizar estas acciones.

Las dificultades enfrentadas por la ONRM en cuanto a la localización del conocimiento geológico del país, conllevaron a la puesta en práctica a nuevas concepciones para el desarrollo del Programa Nacional de Informatización del Conocimiento Geológico (PNICG), pues este conocimiento constituye un elevado número de información de gran importancia y de vital necesidad para establecer relaciones de negocios tanto dentro como fuera de nuestro territorio en la rama de la geología.

El PNICG es llevado a cabo por la ONRM y la Universidad de las Ciencias Informáticas (UCI), es el plan para la informatización de las actividades más importantes de la institución, entre las cuales se pueden mencionar: captación y consultoría de información primaria geológica además de la gestión de la actividad administrativa de la entidad. Algunas de las ventajas que aporta el PNICG son:

- ✚ Constituye una etapa superior para el aprovechamiento más racional del conocimiento de la geología, la minería y el petróleo.
- ✚ Permitirá la disponibilidad de los datos geológicos para las entidades y personas interesadas.
- ✚ La iniciativa tiene el objetivo de abarcar, sistematizar y poner a disposición de la sociedad el conocimiento que el país ha ido acumulando a lo largo de décadas de estudio y trabajo geológico.
- ✚ El proyecto contribuiría a una mejor ubicación de redes eléctricas ya que lleva implícito la georreferenciación de la información.(Latina 2007)

Sin lugar a dudas la ONRM necesita garantizar la perdurabilidad de la información del conocimiento geológico del país, incrementar la velocidad de localización de los datos geológicos, mejorar el procesamiento de los datos dentro de la oficina o por sus clientes, ofrecer mejor eficiencia en cuanto a la accesibilidad de la información geológica, mantener un control adecuado sobre el acceso a la información asegurando su integridad. Esta institución requiere de una solución inmediata a los problemas antes mencionados para evitar de esta forma el deterioro de la información, logrando formatos estándares en su almacenamiento y una digitalización adecuada para conformar el Metadato a través de los datos geológicos.

Debido a lo antes planteado se ha identificado como **problema a resolver**:

¿Cómo facilitar la captación de ficheros de Metadatos geológicos para la Base de Datos del proyecto Metadatos de la Oficina Nacional de Recursos Minerales?

Identificando como **objeto de estudio** de la investigación:

La captación de ficheros de Metadatos geológicos.

Determinándose como **objetivo general**:

Diseñar un subsistema que permita la captación de ficheros referentes a los Metadatos geológicos para la Base de Datos del proyecto Metadatos de la Oficina Nacional de Recursos Minerales.

El **campo de acción** lo constituye:

La informatización para la captación de ficheros de Metadatos geológicos para la Base de Datos del proyecto Metadatos de la Oficina Nacional de Recursos Minerales.

La **Idea a defender**:

Un sistema informático desarrollado a partir del diseño que se propone, facilitará la captación de ficheros de Metadatos geológicos para la Base de Datos del proyecto Metadatos de la Oficina Nacional de Recursos Minerales.

Para el desarrollo de la investigación se trazaron las siguientes **tareas investigativas** a cumplir:

- ✚ Definición del diseño teórico y metodológico de la investigación.
- ✚ Realización de entrevistas al cliente.
- ✚ Análisis del uso de Metadatos en Cuba y el mundo.
- ✚ Comprobación de propuestas de sistemas para la creación de Metadatos de datos geológicos.
- ✚ Conformación de la fundamentación teórica de la investigación.
- ✚ Análisis de las diversas arquitecturas para aplicaciones de Internet, gestores de bases de datos y lenguajes de programación a utilizar.
- ✚ Análisis de los procesos de negocio a automatizar.
- ✚ Levantamiento de requisitos del sistema.
- ✚ Desarrollo del análisis y diseño para la administración de información de los Metadatos.

Los **métodos científicos** a utilizar son:

 **Métodos Teóricos:**

- Análisis histórico - lógico: Se utiliza para el análisis histórico de los procesos de gestión de información en la ONRM y de las características de los principales servicios geológicos existentes en el mundo.
- Análisis y síntesis: Se usa para el estudio de la bibliografía utilizada y para realizar una recopilación de la misma.
- Modelación: Se utiliza para la modelación de diagramas, representar el proceso de desarrollo y propiciar mejor entendimiento de la solución a implementar.

 **Métodos Empíricos:**

- Observación: Se utiliza para identificar los principales procesos que se desarrollan en la ONRM y para definir los requerimientos tanto funcionales como no funcionales.
- Análisis de documentos: Se utiliza principalmente para la revisión de documentos usados en la investigación de las características de los Metadatos geológicos, por ejemplo el Perfil de Cataluña y Procedimiento para la documentación del Metadato de la empresa Geominera de Oriente.
- Entrevistas: Se realiza entrevistas con el fin de validar la propuesta que se presenta, recopilar toda la información necesaria sobre los procesos de captación de ficheros de Metadatos y para definir todos los requerimientos del sistema a desarrollar.

La población definida en la investigación está constituida por todos los especialistas que trabajan en la ONRM. La técnica de muestreo utilizada fue No probabilística – intencional, se escogió el personal de forma directa, pues se entrevistó a los especialistas vinculados con la gestión de la información en la oficina.

Como **posible resultado** de esta investigación se espera:

-  Disponer de la documentación del proceso de desarrollo del sistema de software para facilitar el soporte del mismo.
-  Lograr la eficiencia en el proceso de captación de ficheros y en el trabajo con Metadatos geológicos.
-  Disponer del análisis y diseño de un sistema informático capaz de captar ficheros de Metadatos para la base de datos del Metadato.

El presente trabajo de diploma se estructura de la siguiente forma:

Capítulo 1. Fundamentación Teórica: Se describe la situación actual de la Oficina Nacional de Recursos Minerales y de los procesos que ocurren en la misma, se aborda la fundamentación teórica de la investigación.

Capítulo 2. Tendencias y tecnologías actuales a desarrollar: Se realiza un estudio referente a las tendencias y tecnologías existentes en la actualidad para la confección del proyecto.

Capítulo 3. Presentación de la solución propuesta: Se describen los procesos actuales a través de un modelo de negocio, el cual sirve de base para determinar qué se va a desarrollar. Se determinan las funcionalidades del sistema propuesto.

Capítulo 4. Construcción de la solución propuesta: Aborda aspectos relacionados con la construcción de la solución propuesta, se modelan los diagramas de clases de diseño, se plantea el modelo de datos, y se especifican los principios para el diseño gráfico y la implementación.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En el presente capítulo se abordan los temas relacionados con la fundamentación teórica de la investigación. Se describen los conceptos asociados al dominio del problema, así como la descripción de los flujos de trabajo de la Oficina Nacional de Recursos Minerales que conciernen a la captación de ficheros de Metadatos haciendo uso de conceptos y definiciones, además de plantear y profundizar sobre la situación problemática. Se analizan algunas soluciones existentes de gestión a nivel internacional con el objetivo de identificar sus principales funcionalidades.

1.2. Conceptos asociados al dominio del problema

1.2.1. Datos geológicos

Se considera como datos geológicos a la unidad de información que se ha digitalizado u organizado como producto de la ejecución de un proyecto de conformación de bases de datos. (Oriente Oct/2007)

Habitualmente localizar un dato geológico que se necesite para un fin determinado se torna muy difícil, pues no existe un mecanismo lo suficientemente general que difunda las características fundamentales de los datos existentes así como su disponibilidad. Se necesita entonces una estructura bien organizada que permita documentar fácilmente la información existente y la que se cree posteriormente, por lo que se propone catalogar los datos haciendo uso de los Metadatos.

1.2.2. Metadatos

Se puede definir como Metadato que estos son datos altamente estructurados que describen información, el contenido, la calidad, la condición y otras características de los datos. Es "Información sobre información" o "datos sobre los datos". Entre los principales usos de los Metadatos están: Organizar y mantener el acervo del conjunto de datos de una organización. Proveer información necesaria para interpretar y procesar datos transferidos por otra organización. Los Metadatos están estructurados por un mínimo de elementos tales como: título, autor, fecha de creación, etc. Típicamente, los elementos que conforman un Metadato están definidos por algún estándar, donde los

usuarios que deseen compartir Metadatos están de acuerdo con un significado preciso de cada elemento.(Instituto Nacional de Estadística 2003)

Se arriba al planteamiento de que un Metadato es aquella información descriptiva sobre el contexto, calidad, condición o características de un recurso, dato u objeto que tiene la finalidad de facilitar su recuperación, autenticación, evaluación y preservación.

1.2.3. Metadatos geológicos

Rafael Oliva plantea que un metadato geológico es un conjunto de información que identifica diferentes aspectos relacionados a grupos de datos o a datos específicos y permite conocer características de estos, que los particularizan dentro de un conjunto. Describe aspectos de los datos geoespaciales como son: calidad, actualización, referencia geoespacial, autor, entre otras. Constituyen información sobre la forma y el contenido de los recursos informativos.(Rafael Oliva Santos 2006) .

Los metadatos geológicos son documentados basados en el estándar internacional ISO 19115.TC 211 desarrollado en el año 2003 por el comité 211 de la ISO¹. El objetivo de este estándar internacional es proporcionar una estructura para describir datos geográficos digitales. Este estándar está pensado para ser utilizado por analistas, programadores y desarrolladores de sistemas de información geográfica, así como todos aquellos que quieran entender los principios básicos y los requerimientos de la estandarización de la información geográfica. Además define los elementos de los metadatos, proporciona un esquema y establece un conjunto común de terminología de metadatos y definiciones. En esta norma los metadatos geológicos se representan a través de paquetes UML².

A nivel nacional la Empresa Geominera de Oriente por orientación de la ONRM realizó el análisis de contenido del metadato para los conjuntos de datos geológicos y desarrollar un perfil para la documentación del metadato cubano, es decir, crear nuestro propio estándar basado en las normas ISO.

¹ Organización Internacional de Normalización

² Lenguaje Unificado de Modelado

Se definieron como las secciones diferenciadas para el metadato las siguientes:

- ✚ Información del Metadato.
- ✚ Identificación de los Datos.
- ✚ Información del Contenido.
- ✚ Calidad de los Datos.
- ✚ Representación Espacial.
- ✚ Sistema de Referencia.
- ✚ Distribución de los Datos.
- ✚ Información sobre las Citas.

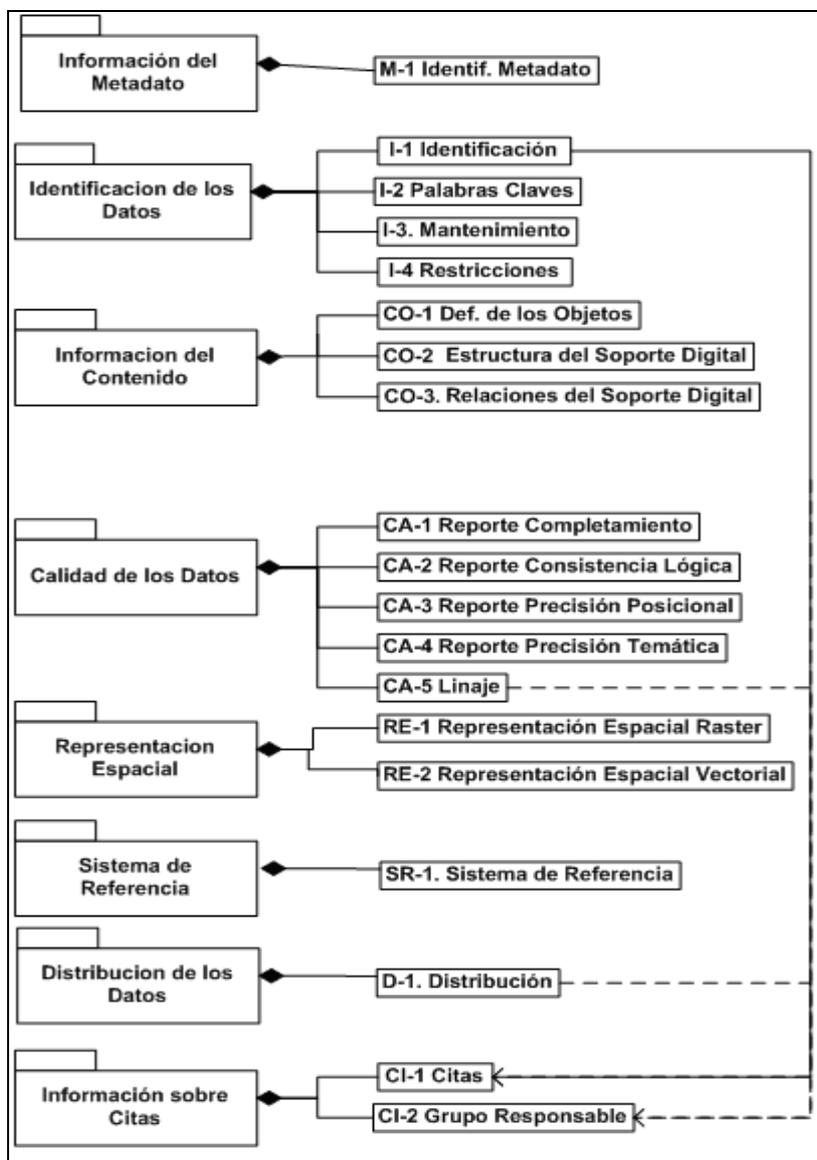


Figura 1 Metadato. Secciones y modelos de documentación.

La documentación de los datos se ha convertido en un trabajo necesario para realizar una correcta gestión y explotación de los mismos. El aprovechamiento de los datos no está pensado para el uso interno de una organización o empresa, sino también para poderla compartir e intercambiar entre los diferentes productores de datos, o entre los productores y los usuarios. Para que esta documentación sea comprensible para todos, es decir, para que todos “hablen el mismo idioma”, se han diseñado los estándares de metadatos antes mencionados que pretenden poner en común los puntos de vista de los diferentes productores de datos.

Las ventajas de los metadatos geológicos son:

- ✚ Propicia mejor organización de la institución o compañía y con ello mejores resultados.
- ✚ Mantiene el valor de la inversión en datos geográficos. Hace persistente el conocimiento acerca de los datos cuando el personal calificado que los creó o que tiene relación directa con ellos y conoce todas sus características, deja de prestar servicios en la institución.
- ✚ Brinda información sobre los datos disponibles de manera que posibilita su catalogación, así como facilita información de los lugares y formas de almacenamiento de dichos datos.
- ✚ Provee información rápida a potenciales comercializadores de los datos geográficos.
- ✚ Facilita la ubicación y la detección de los datos, para así lograr identificarlos, localizarlos, accederlos y que puedan ser utilizados por personas o instituciones que los requieran.
- ✚ Provee una guía de los datos geográficos en cuanto a sus rasgos principales.
- ✚ Propicia un mecanismo para mantener y auditar los datos existentes.
- ✚ Facilita la comunicación y el intercambio entre el personal especializado y el mundo exterior. Fortalece las relaciones empresa-sociedad.

1.2.4. Sistema

Un sistema es el conjunto de medios interconectados (objetos, seres humanos, informaciones), utilizados según un proceso dinámico con el fin de alcanzar los objetivos señalados. Se considera a un todo formado por partes, estas partes son afectadas por factores internos y externos a ese todo. Además, cada parte debe interrelacionarse e interactuar entre sí y con otras provenientes del medio externo. También, éste todo debe, a su vez, interactuar como un conjunto lógico para dar respuestas a sus propios requerimientos y a los que el medio externo espera de él. Si cada parte de este todo, actúa e interrelaciona interna y externamente mediante plan, método y orden, se estará entonces, en presencia de un sistema totalmente integrado.

La IEEE³ plantea que: "Sistema es un todo integrado, aunque compuesto de estructuras diversas, interactuantes y especializadas. Cualquier sistema tiene un número de objetivos, y los pesos asignados a cada uno de ellos pueden variar ampliamente de un sistema a otro. Un sistema ejecuta una función imposible de realizar por cualquiera de las partes individuales. La complejidad de la combinación está implícita."

Se puede definir sistema como el conjunto de procesos o elementos interrelacionados (definiciones, nombres, símbolos) con un medio para formar una totalidad encauzada hacia un objetivo común. Tienen tres características estructurales básicas: Los elementos que lo componen, las relaciones entre los mismos y los límites que determinan los elementos que pertenecen o no al sistema.

1.2.5. Sistema informático

Andrés Berdasco Blanco conceptualiza Sistema Informático como el conjunto formado por elementos hardware y software que constituyen los recursos a los cuales llegan las peticiones de los usuarios para ser atendidas, en otras palabras se puede decir que es uno o varios ordenadores con un sistema operativo y con los programas (software) necesarios por los usuarios.(BERDASCO BLANCO 2007)

Un Sistema informático es aquel sistema que se encarga del manejo de información en la computadora, a través de la cual el usuario controla las operaciones que realiza el procesador.

Los sistemas informáticos se pueden caracterizar desde diferentes puntos de vistas:

- ✚ Para una organización: un sistema informático puede ser: un departamento como cualquier otro, con recursos, que está a disposición de la organización.
- ✚ En el mundo de la informática: es un conjunto de servidores, redes y terminales de trabajo para "hacer software"
- ✚ Para los usuarios: es una herramienta más que les da la organización para mejorar el funcionamiento de sus tareas.

Se define que un sistema informático es aquella unión que se concibe entre los programas y los componentes físicos, ofreciéndoles respuestas a aquellos usuarios que han hecho de una forma o de otra un determinado pedido, las cuales pueden ser aplicaciones de escritorio o aplicaciones Web.

³ Standard Dictionary of Electrical and Electronic Terms

A lo largo de la historia de las aplicaciones se ha demostrado con grandes resultados que las aplicaciones informáticas, ya sean de escritorio o Web han tenido un impacto crucial en la economía de un país, pues estas han jugado un gran papel en el progreso de la humanidad en todas las esferas (educación, salud, cultura, deporte, economía, entre otras), estos avances se han demostrado con el perfeccionamiento intelectual humano que día a día se va alcanzando en el mundo, esto ha hecho al hombre más capaz pues ha puesto sus conocimientos en función de la evolución según las necesidades por las cuales ha pasado.

Para poder manejar con eficiencia una aplicación de escritorio, aquellos usuarios que vayan a hacer uso de una, deben ante todo ostentarla en su puesto de trabajo (PC), donde estará ocupando parte del espacio disponible de su PC para hacer uso de otros recursos y servicios, se puede decir que entre las aplicaciones Web y de escritorio, esta última es la más segura, por sus facilidades de uso, manejo, entendimiento y sobre todo la gran facilidad de ayuda que estas brindan. Hoy en día muchas empresas hacen uso de este tipo de aplicaciones y aunque en los últimos años la mayoría de las empresas están emigrando hacia las aplicaciones Web, les está solucionando de cierta forma los problemas que allí tienen.

Por otra parte las aplicaciones Web; para su uso sólo necesitan estar intercomunicadas con un servidor, los usuarios pueden acceder sin necesidad alguna de tener instalada una aplicación en su puesto de trabajo, con sólo sentarse en una PC cualquiera puede acceder con facilidad a las mismas, son más eficientes que las aplicaciones de escritorio en cuanto a la problemática de que se puede tener el servidor en una provincia o país y estar sentado en otra u otro respectivamente ya que el cliente sólo tiene que conectarse a dicho servidor, por otra parte es más beneficiosa que la de escritorio porque no ocupa espacio en su puesto de trabajo. (Cabrera and Velázquez 2007)

1.2.6. Aplicaciones Web

Una aplicación Web es una aplicación informática que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una intranet. Es un sitio que se encuentra en la Red, al cual los usuarios acceden con el fin de beneficiarse con el uso de la información que posea. Aborda un determinado tema. Se dice que los usuarios se benefician con la información que estos les brindan porque una vez que acceden a él, pueden interactuar con ella, actualizarla o modificarla según los privilegios de seguridad de la aplicación que estos tengan. Las características básicas de una aplicación Web son las siguientes:

- ✚ Está alojada en un servidor.
- ✚ Es accesible a través de una red telemática y mediante la utilización de un navegador Web.
- ✚ La lógica de la aplicación Web se ejecuta íntegramente en el servidor Web y el navegador del cliente sólo representa los datos.
- ✚ La navegación a través de la Web y la entrada de datos por parte de un usuario afectan el estado de la lógica de la aplicación
- ✚ Se ejecuta íntegramente en el servidor y puede interactuar con otros procesos y aplicaciones Web.
- ✚ El acceso a la aplicación puede ser público o restringido, dependiendo hacia quién vaya dirigida.
- ✚ Las actualizaciones y el mantenimiento de la aplicación es transparente para los usuarios, ya que éstos no deberán instalar ni actualizar nada.
- ✚ Las aplicaciones son multiplataforma porque pueden ejecutarse en cualquier Sistema Operativo que posea un navegador Web.

La creciente popularidad de las aplicaciones Web se debe a sus múltiples ventajas, entre las cuales se pueden citar:(C. G. Hernandez 2006)

- ✚ Multiplataforma: Con un sólo programa y un único ejecutable, las aplicaciones pueden ser utilizadas a través de múltiples plataformas, tanto de hardware como de software.
- ✚ Actualización instantánea: Debido que todos los usuarios de la aplicación hacen uso de un sólo programa que radica en el servidor, los usuarios siempre utilizarán la versión más actualizada del sistema.
- ✚ Suave curva de aprendizaje: Los usuarios, al utilizar la aplicación a través de un navegador, hacen uso del sistema tal como si estuvieran navegando por Internet, por lo cual su acceso es más intuitivo.
- ✚ Fácil de integrar con otros sistemas: Debido a que se basa en protocolos estándares, la información manejada por el sistema puede ser accedida con mayor facilidad por otros sistemas.
- ✚ Acceso móvil: El usuario puede acceder a la aplicación con la única restricción de que cuente con un acceso a la red privada de la organización o a Internet, dependiendo de las políticas de dicha organización; puede hacerlo desde una computadora de escritorio o desde una agenda electrónica; desde su oficina, hogar u otra parte del mundo.

Las aplicaciones Web además de las ventajas que presentan tienen algunas desventajas, las cuales son:

- ✚ Acceso limitado: La necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- ✚ La interactividad: No se produce en tiempo real, en las aplicaciones Web cada acción del usuario conlleva un tiempo de espera algunas veces excesivo, hasta que se obtiene la reacción del sistema.
- ✚ Elementos de interacción muy limitados: En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.
- ✚ Diferencias de presentación entre plataformas y navegadores: La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

1.3. Objeto de Estudio

1.3.1. Descripción General

La Oficina Nacional de Recursos Minerales

En 1995 es promulgada la ley 76 “Ley de Minas”(Cuba 1995) que creó la ONRM, autoridad minero petrolera de la República de Cuba que es la encargada de administrar el conocimiento geológico y toda la información geológica, minera y petrolera de la nación.

En el 2001 el Comité Ejecutivo del Consejo de Ministros aprobó las funciones de la ONRM (acuerdo 3985), adscripta al Ministerio de la Industria Básica y tiene entre sus misiones fundamentales:

- ✚ Fiscalizar y controlar la actividad minera y el uso racional de los recursos minerales, según lo dispuesto en la legislación vigente.
- ✚ Fiscalizar y controlar las actividades de exploración-producción de hidrocarburos líquidos y gaseosos, según las disposiciones legales vigentes.
- ✚ Asesorar al Ministerio de la Industria Básica y demás organismos de la administración central del estado sobre las actividades mineras de exploración producción de hidrocarburos, sin perjuicio de sus debidas competencias.

- ✚ Responder por el registro minero y el registro petrolero y mantener actualizadas las anotaciones sobre las concesiones mineras reservadas, los contratos de exploración y producción de petróleo, yacimientos, manifestaciones minerales y de hidrocarburos, áreas en investigaciones, minas y pozos de hidrocarburos en explotación o abandonados.
- ✚ Ser el depositario de la información geológica, minera y petrolera de la nación, recibir, organizar y conservar la información, así como brindar servicios de información técnica.
- ✚ Velar por el cumplimiento de las normas de seguridad minera previstas en la legislación vigente para evitar la ocurrencia de accidentes o peligros para la seguridad humana.
- ✚ Procesar y generalizar la información especializada conservada en su archivo técnico.

La Oficina Nacional de Recursos Minerales tiene una estructura de un solo nivel, su oficina central, que tiene las direcciones que a continuación se relacionan:

- ✚ Dirección General.
- ✚ Dirección de Documentación e Informática.
- ✚ Dirección de Control Económico.
- ✚ Dirección de Registro Minero.
- ✚ Dirección de Evaluación Económica.
- ✚ Dirección de Minerales.
- ✚ Dirección de Níquel.
- ✚ Dirección de Hidrocarburos.(Diseño 2006)

En cuanto a la captación de ficheros:

Para razonar sobre la captación de ficheros en la ONRM es necesario dominar algunos términos importantes, por ejemplo: ¿Qué es un fichero? ¿Cuáles son las características fundamentales de un fichero? y ¿En qué consiste básicamente la captación?

Un fichero es la información que se encuentra en un soporte de almacenamiento informático: textos, documentos, imágenes, bases de datos, ficheros de sonido, hojas de cálculo, etc. Se identifica por un nombre, un punto y una extensión (indica de qué tipo es el fichero). Cada uno de ellos se caracteriza por tener un nombre que lo identifica. El nombre puede estar seguido de un punto y una extensión, compuesta por tres caracteres que identifican el tipo de fichero del que se trata. Algunas extensiones comunes son: *EXE* y *COM* (ficheros ejecutables, programas), *TXT* (ficheros de texto) y *DOC* (documentos de Word), etc. (Interbusca 2008)

Fichero es un conjunto de información relacionada, tratada como una unidad de almacenamiento y organizada de forma estructurada para facilitar la búsqueda de datos individuales. Un fichero está compuesto por registros homogéneos que contienen información organizada en campos, siendo un campo la mínima unidad de procesamiento con significado propio. En un lenguaje de alto nivel, el fichero no es manejado directamente por el propio programa, sino por el Sistema Operativo. La vida de todo fichero comienza cuando se crea y acaba, cuando se borra. Los ficheros se pueden clasificar atendiendo a distintos criterios:(UMA 2005)

- ✚ Por el tipo de sus registros.
- ✚ Por la forma de almacenar los datos.
- ✚ Por su contenido.
- ✚ Por su duración o tiempo de vida.
- ✚ Por su uso.
- ✚ Por su organización.

En cuanto a la organización de los ficheros hay varios motivos para estructural la información en los mismos:

- ✚ Acceso rápido a los registros.
- ✚ Economía de almacenamiento.
- ✚ Facilitar la actualización de los registros.
- ✚ La estructura permite reflejar la organización real de la información.

Se debe, pues, optar por una u otra organización, atendiendo a la forma en que se va a usar el fichero. Si el fichero está almacenado en un dispositivo físico direccionable, es posible realizar actualizaciones directas, y también: (UMA 2005)

- ✚ Consulta: Si el fichero contiene registros de longitud fija, es posible determinar la posición de comienzo de cada uno a partir de su posición relativa en el fichero.
- ✚ Modificación: Una vez localizado un registro, se puede reescribir este en el propio fichero, siempre que al modificar el registro no aumente su longitud.
- ✚ Borrado: No es posible eliminar un registro del fichero. Se realiza un borrado lógico.

Importancia del uso de los ficheros:

- ✚ Constituye una forma muy efectiva de comunicarse con el entorno.
- ✚ Deja evidencia de la ejecución de un programa.
- ✚ La cantidad de información que se almacena es mucho mayor que en memoria.

Básicamente la captación es la acción de captar, archivar o recolectar, la cual puede depender de determinado criterio de captación, ya sea por nombre, tamaño, tipo, contenido u otra característica específica.

La norma ISO 19115 proporciona una estructura para describir información geográfica mediante elementos de metadatos y establece una terminología común para los mismos pero no desarrolla como poder llevar a cabo su implementación. La ISO 19139 ⁴es una especificación técnica que desarrolla una implementación en XML del modelo de metadatos descrito por ISO 19115. XML es un lenguaje de marcado que se utiliza para crear documentos que contengan información estructurada. Para la creación de estos documentos es necesario definir etiquetas y relaciones entre las mismas. Basado en estas normas se hace uso de ficheros XML para la documentación del metadato y con el fin de generar un archivo de intercambio de metadatos.

Un archivo de intercambio de metadatos, acorde con el estándar ISO19115 y en formato XML, va a ser un documento XML que siga la sintaxis definida por la especificación técnica ISO19139. Es decir, esta especificación técnica define un conjunto de esquemas en XML que van a describir los metadatos asociados a cada nivel de información, permitiendo así su descripción, asegurando su validación y su posterior intercambio a través de archivos de metadatos. Estos esquemas XML se han generado a partir de los modelos UML definidos en la ISO19115 aplicando las reglas de codificación definidas en la norma ISO 19118⁵. Esta norma establece un conjunto de reglas de codificación para transformar los esquemas conceptuales UML descritos en cualquiera de las normas de la serie ISO 19100 en esquemas XML. (Maganto and Ballari 2008)

Una de las formas de definir la sintaxis de los documentos XML es mediante una tecnología asociada a XML y denominada "XML-Schema". Para cada lenguaje derivado de XML, se debe crear un documento siguiendo la especificación de XML-Schema, comúnmente denominado "Esquema XML", que describa la estructura de los documentos XML y permita posteriormente validarlos, garantizando

⁴ ISO/TS 19139-Geographic Information-Metadata -XML schema implementation

⁵ Geographic Information-Encoding

así que la estructura sea válida para un contexto determinado. El propósito de un esquema es definir los componentes válidos de un documento XML: elementos que pueden aparecer, atributos, elementos hijos, orden y número de los elementos, tipos de datos, valores por defecto de los elementos, etc. Algunas de las características de los documentos XML-Schema para la definición de los lenguajes derivados XML son las siguientes:

- ✚ Los tipos de datos a utilizar pueden ser simples (no tienen ni elementos hijos ni atributos) o complejos (tienen elementos hijos y/o atributos).
- ✚ La declaración de posibles espacios de nombres (namespace). Los espacios de nombres son colecciones de nombres, identificados mediante una referencia URI⁶ que se utiliza como tipos de elementos y nombres de atributos. La ventaja de la utilización de los espacios de nombres es que permitirá eliminar las ambigüedades y solucionar los problemas de homonimia que se producen en todos los documentos XML, ya que pueden existir etiquetas con un mismo nombre pero con diferentes significados y espacios semánticos.

La implementación de esquemas XML, mediante codificaciones basadas en reglas construidas para los modelos UML de las series 19100, permite que los esquemas creados cumplan una serie de características: (Maganto and Ballari 2008)

- ✚ La interoperabilidad: Entre esquemas procedentes de especificaciones de las series ISO 19100.
- ✚ La previsibilidad: Estos esquemas son previsibles para cualquier clase, atributo, asociación, etc., se codifican igual para cualquier elemento UML del mismo tipo.
- ✚ La usabilidad: Tener una codificación basada en reglas permite generar XML de un modo automático o semiautomático.
- ✚ La extensibilidad: Se han definido extensiones mediante la creación de un diccionario de datos, con las mismas características que el definido en ISO 19115, para facilitar la interoperabilidad, facilidad en el uso, etc. Entre las extensiones que se definen están:
 - Para el entorno WEB.
 - Culturales y lingüísticas.
 - Las basadas en la transferencia de información espacial: para la transferencia, para los catálogos.

⁶ Uniform Resource Identifier

- Para las enumeraciones y las lista de códigos.
- ✚ El polimorfismo: La habilidad para asumir diferentes formas, es decir, facilitar la adaptabilidad cultural y lingüística.

Para la captación de ficheros, en la oficina, existe un consejo formado por especialistas encargados de revisar el fichero de metadatos siguiendo las normas y estándares de calidad necesarios para la documentación del metadato. Este fichero es generado por una aplicación de escritorio donde el geólogo entra manualmente los datos necesarios para formar el metadato geológico, es recibido luego de su revisión y validación por el administrador de la base de datos de metadatos geológicos, encargado además de captarlo o archivarlo para actualizar la base de datos y la información necesaria para las consultas en el módulo de recuperación.

Es importante mencionar que el administrador de la base de datos es una persona seleccionada por el consejo, el cual debe cumplir con una serie de aptitudes y condiciones mínimas para desempeñar este papel.

En cuanto a la seguridad:

La seguridad es un aspecto vital de las aplicaciones Web. El primer paso para crear una aplicación segura es conocer los tipos de problemas de seguridad que pueden surgir en aplicaciones basadas en Web. Además se deben comprender las estrategias básicas que se utilizan para proteger la aplicación y el sistema, para definir que tipo de seguridad debe ser implementada, ya sea una seguridad personalizada o no personalidad según el resultado del estudio de las estrategias básicas.

La seguridad en aplicaciones Web, se encuentra relacionada pura y exclusivamente con: la lógica, la escritura de código y el contenido de una aplicación Web. Si bien está claro, que toda aplicación Web, requiere de un entorno conformado por elementos externos, tales como: sistemas operativos, servidor Web, servicios, etc. para poder cumplir su función, los inconvenientes o controles relativos a estos últimos, no deben ser considerados problemas propios de la aplicación Web. El tema de seguridad está muy relacionado con algunos aspectos mencionados a continuación:

- ✚ Permisos: Especifican que puede hacer exactamente un usuario en un contexto dado. Es un parámetro que especifica si su poseedor dispone por ejemplo de acceso a una determinada función del sistema o a una parte de la interfaz de usuario del sistema.
- ✚ Roles: Un rol es un conjunto de permisos que puede asignarse a un usuario.

- ✚ Autenticación: Es el proceso de validación de la identidad de un usuario para permitir o denegar una solicitud. Por ejemplo, un usuario debe proporcionar un nombre de usuario y una contraseña que se comprueban en una autoridad (por ejemplo, una base de datos o un servidor de dominios).
- ✚ Autorización: Es el proceso de conceder o denegar el acceso a los recursos a usuarios específicos. En cualquier tipo de red, se debe iniciar la sesión (es decir, autenticarse). Una vez iniciada la sesión de forma satisfactoria, se puede tener acceso a archivos, carpetas, impresoras y otros recursos específicos, o sea se dispone de autorización para obtener acceso a los recursos en función de las credenciales de inicio de sesión.
- ✚ Confidencialidad: Es una condición que se mantiene al asegurar que sólo los usuarios autorizados puedan tener acceso a la información. Garantiza el cumplimiento del nivel necesario de reserva en cada intersección del procesamiento de datos para evitar una revelación sin autorización.
- ✚ Integridad: Implica proteger la precisión y la totalidad de los métodos de información y procesamiento. Se mantiene cuando existe veracidad y confiabilidad de la información dentro del entorno del sistema, y cuando se evita la modificación no autorizada de datos. El almacenamiento de datos incorrectos dentro del sistema puede ser tan perjudicial como la misma pérdida de datos.

Es importante señalar que en las aplicaciones Web no existe el método de seguridad perfecto, incluso en los sistemas seguros, los usuarios malintencionados pueden utilizar herramientas sofisticadas para intentar introducirse en el sistema. Afortunadamente, la mayoría de los ataques provienen de usuarios relativamente poco sofisticados que sólo buscan sitios en los que resulte fácil penetrar. El objetivo inicial debería ser proteger el sitio de los ataques comunes y redundantes. Algunas amenazas para la seguridad de las aplicaciones son:

- ✚ Suplantación: Suplantar (spoof) es utilizar los datos de identificación de otro usuario o proceso de forma no autorizada. En su versión más simple, la suplantación consistiría en introducir las credenciales de un usuario diferente.
- ✚ Manipulación: Significa cambiar o eliminar un recurso sin autorización. El ejemplo típico consiste en desfigurar una página Web, para lo cual, el usuario malintencionado logra acceso al sitio y cambia algunos archivos. Un modo indirecto de manipular son los ataques mediante secuencias de comandos en los que el usuario malintencionado consigue que se ejecute

código (secuencia de comandos) enmascarándolo como la entrada de datos de un usuario en un formulario o como un vínculo.

- ✚ Repudio: Las amenazas de repudio tienen como objetivo ocultar las pruebas de un ataque. En una aplicación Web, esto puede significar que se está suplantando a un usuario inocente usando sus credenciales.
- ✚ Revelación de información: Significa simplemente robar o desvelar información que se supone que es confidencial. Un ejemplo clásico es robar contraseñas, pero puede implicar el acceso a cualquier archivo o recurso del servidor.
- ✚ Denegación de servicios: Un ataque de denegación de servicio consiste en hacer deliberadamente que una aplicación esté menos disponible de lo que debería. Un ejemplo típico es sobrecargar una aplicación Web de forma que no pueda servir a los usuarios normales. Como alternativa, los usuarios malintencionados pueden intentar simplemente bloquear el servidor.
- ✚ Concesión de privilegio: Consiste en usar medios para obtener más permisos de los asignados normalmente. Por ejemplo, en un ataque de concesión de privilegios que tenga éxito, un usuario malintencionado logra obtener permisos de administrador en el servidor Web, lo que le da carta blanca para sembrar el caos.

Chris Shiflett ⁷ dijo que: “ *Ningún Lenguaje de programación puede prever código inseguro, aunque las características del lenguaje puedan ayudar a bloquear revelación de información confidencial* ” Es importante destacar que existen más ataques, de los cuales no se habla puesto que se expone sólo las vulnerabilidades Web más comunes. Los conceptos también se aplican a cualquier lenguaje de programación. Los ataques explicados a continuación son: Ejecución de código remotamente: Esta vulnerabilidad permite al atacante ejecutar código en el servidor vulnerable y obtener información almacenada en él. Los errores de codificación impropios resultan de esta vulnerabilidad. A veces es difícil descubrir vulnerabilidades durante la puesta a prueba del sistema, pero tales problemas son a menudo revelados mientras se hace la revisión de código.

- ✚ Inyección de código SQL: Afecta directamente a las bases de datos de una aplicación. El problema radica al filtrar erróneamente las variables utilizadas en parte de la página con código SQL. Una Inyección SQL consiste en insertar o inyectar código SQL malicioso dentro de código

⁷ Analista de seguridad, estudioso de temas relacionados con los servicios de seguridad en aplicaciones Web.

SQL, para alterar el funcionamiento normal y hacer que se ejecute el código “invasor” dentro del sistema.

- ✚ Cross Site Scripting (XSS): Basado en la explotación de vulnerabilidades del sistema de validación de HTML incrustado. El problema es que normalmente no se valida correctamente. Esta vulnerabilidad puede estar presente de forma directa (foros, mensajes de error) o indirecta (redirecciones).

1.3.2. Descripción actual del dominio del problema

Entre los principales problemas que presentan la ONRM en cuanto a la gestión de la información está; la pérdida de información sensible, el difícil procesamiento de la información, el manejo óptimo de datos y la deficiente accesibilidad a la información geológica. En todos los casos podemos identificar como los más afectados a la misma ONRM y los posibles clientes que puedan estar interesados en la información geológica para establecer negocios geológicos en el país. Se prevé que el volumen de información supere los 4 TBytes, depositados en centros de almacenamiento en todo el país y en varios formatos diferentes.

La deficiente accesibilidad a la información geológica provoca el retraso en los convenios de trabajo y en el establecimiento del negocio, violaciones de la ley y el desaprovechamiento de clientes potenciales. La ONRM exige la incorporación de sistemas informáticos debidamente protegidos y asegurados, robustos, que sigan estándares informáticos de calidad y normas requeridas por la geociencia, que apoyen la gestión, la comunicación, la información y sirvan de apoyo al establecimiento de mecanismos de control y seguimiento del conocimiento geológico. Por esto se hace necesario la creación de un sistema de Metadatos que facilite la captación de ficheros de Metadatos y el control de los usuarios de la Base de Datos de Metadatos.

La información que se maneja es la referente a los usuarios responsables de realizar las actualizaciones, sus datos personales, los datos de la persona que hace entrega del fichero de actualización, la fecha de actualización, la lista de reportes. Además luego de recibir el fichero de Metadatos Geológicos se almacena en la Base de Datos todos los datos geológicos con sus respectivos campos de almacenamiento, teniendo en cuenta que todos los campos priorizados estén llenos correctamente.

Una solución satisfactoria pudiera implicar un mejor posicionamiento de la ONRM en el mercado internacional y una mayor capacidad de respuestas ante las distintas situaciones que se le presenten a la organización en lo que se refiere al conocimiento geológico así como el incremento de nuevos clientes potenciales.

1.3.3. Situación Problemática

La ONRM es la entidad geológica encargada de administrar y gestionar el conocimiento geológico obtenido a lo largo de más de 40 años. Uno de los principales problemas que afectan en la actualidad el funcionamiento de dicha entidad, es la disposición de la información geológica, pues esta se encuentra distribuida por todo el territorio nacional y en distintos formatos, volviéndose difícil su localización en el momento de satisfacer determinado interés, lo que obstaculiza su desarrollo, comercialización e intercambio. Existe la necesidad de preservar de manera sustentable y poner a disposición de la economía nacional todo el conocimiento geológico, con la rapidez que exige la toma de decisiones en los momentos actuales. Para ello se hará uso del Metadato geológico con el objetivo de digitalizar toda esta información.

Los Metadatos tienen como funcionalidad práctica fundamental ayudar a los usuarios a discernir si la información es la más adecuada para un fin concreto sin tener que consultarla directamente. El usuario podrá obtener elementos para la identificación de la información buscada a través de palabras claves que caracterizan la información geográfica.

La ONRM necesita garantizar la perdurabilidad de la información del conocimiento geológico del país, incrementar la velocidad de localización de los datos geológicos, mejorar el procesamiento de los datos dentro de la oficina o por sus clientes, ofrecer mejor eficiencia en cuanto a la accesibilidad de la información geológica, mantener un control adecuado sobre el acceso a la información asegurando su integridad. Esta institución requiere de una solución inmediata a los problemas antes mencionados para evitar de esta forma el deterioro de la información, logrando formatos estándares en su almacenamiento y una digitalización adecuada para conformar el Metadato a través de los datos geológicos, propiciando un nivel de seguridad óptimo para todos estos datos.

Como consecuencia surge la necesidad de crear un sistema que mantenga la seguridad pertinente durante la captación de ficheros de Metadatos para la Base de Datos del proyecto de Metadatos de la ONRM

Con la implementación de una aplicación Web para el proceso de captación de ficheros de Metadatos geológicos para la Base de Datos del proyecto Metadatos de la ONRM, se logrará un nivel de seguridad adecuado para lograr la integración de los Metadatos y su fácil recuperación, gestionando eficientemente las solicitudes de los clientes mediante un manejo óptimo, propiciando rapidez y eficiencia en los servicios prestados, además de lograr flexibilidad a posibles actualizaciones del sistema. Esta aplicación formará parte de las herramientas a usar por esta institución para lograr el perfeccionamiento en sus procesos de trabajo.

1.4. Análisis de otras soluciones existentes

Con el acelerado auge de las Tecnologías de la Información y las Comunicaciones (TIC), se han desarrollado diversas aplicaciones que surgen con la necesidad de informatizar el conocimiento geológico debido a las ventajas y facilidades que este puede proporcionar para algunos sectores, por ejemplo el económico. A continuación se presentan algunas características de aplicaciones que se monitorearon con el fin de identificar sus principales funcionalidades y aportes novedosos a tener en cuenta.

El **Servicio Geológico Mexicano (SGM)**(Minería 2007) el cual es un sistema que permite el control de los procesos de Atención, Servicios y Ventas, Mapas en línea, presta servicios mediante un Banco de Datos de Bases de Datos, Inventarios mineros, Estadísticas mineras, Sistemas de Información geográfica (SIG), otros servicios como Certificación de reservas, Contratos de servicios, Asesoría geológica y Biblioteca, además oferta publicaciones de revistas, boletines y determinados artículos electrónicos relacionados con la geociencia.

El **Servicio Geológico Minero Argentino (SEGEMAR)**(Minera 2007) constituye otro ejemplo de aplicaciones de este tipo. Este oferta algunos productos y servicios como Catálogos, un Buscador Gráfico de productos, hace referencia a algunos proyectos de colaboración llevados a cabo por SEGEMAR, ofertas mineras brindando datos sociales, demográficos y económicos, así como publicación de artículos científicos.

El **Servicio Geológico de Brasil (CPRM)**(Minerales 2007) también es otro de los ejemplos de aplicaciones desarrolladas con el fin de automatizar el conocimiento geológico. Es el responsable del Programa Geológico de Brasil, brinda información sobre el acceso a mapas, a la información de los recursos hidráulicos y mineros constituyendo un gran potencial para la economía brasileña por ser un

territorio netamente rico en minerales, haciendo mención a los servicios de laboratorios y librerías relacionadas con este tipo de información.

Science for a Changing World (USGS)(Survey 2007) es el Servicio Geológico de los Estados Unidos, es una agencia de investigación independiente, que recopila, estudia, analiza y provee información científica sobre las condiciones, asuntos y problemas concernientes a los recursos naturales, tales como el agua, los minerales, el carbón, el petróleo y el gas. Brinda servicios de ventas, banco de imágenes tanto satelitales como de vistas aéreas, mapas actualizados con una rica gama de información de todo su territorio, estadísticas mineras, sistemas de información geográfica. Buscadores gráficos de mapas e información.

Después de un estudio se plantea que en todos los casos analizados se maneja de forma muy eficiente el acceso a la información por los diferentes usuarios que accedan al sistema, manteniendo un control adecuado sobre toda la administración del mismo. Además estas aplicaciones no pueden ser usadas por Cuba debido a que estos sistemas clasifican como software propietario y requieren de licencias para su uso, además cada área geográfica tiene sus características particulares en cuanto al conocimiento geológico, por lo que se necesita implementar un sistema específico para cada zona que se adapte a las peculiaridades requeridas y que asegure un funcionamiento exitoso del mismo en cuanto a la gestión de los datos.

1.5. Conclusiones parciales.

- ✚ El estudio detallado sobre los fundamentos teóricos de la captación de ficheros de Metadatos propició desarrollar el marco conceptual de la investigación.
- ✚ Debido al monitoreo de las aplicaciones a nivel internacional y a la inexistencia de una de este tipo en Cuba, se definió que la construcción de un sistema informático capaz de captar ficheros de Metadatos resulta beneficioso para la Base de Datos del proyecto de Metadatos de la ONRM.

CAPÍTULO 2 TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR

2.1. Introducción

Con la introducción de Internet, se han abierto infinidad de posibilidades en cuanto al acceso a la información desde casi cualquier sitio del mundo. Esto representa un desafío a los desarrolladores de aplicaciones, ya que los avances en tecnología demandan cada vez aplicaciones más rápidas, ligeras y robustas que permitan utilizar y explotar las ventajas de la Web. El objetivo principal de este capítulo es precisar y fundamentar las tecnologías a utilizar en la investigación, teniendo presente las particularidades de las necesidades planteadas por la ONRM.

2.2. Las tecnologías de la Información y las Comunicaciones.

Las Tecnologías de la Información y las Comunicaciones (TICs) son aquellas que se necesitan para la gestión y transformación de la información, en particular el uso de las computadoras y los software que permiten crear, modificar, almacenar, proteger y recuperar información.

Las TICs constituyen un elemento esencial de la Sociedad de la Información, estas brindan la capacidad universal de acceder y contribuir la información, las ideas y el conocimiento. Promover el intercambio y el fortalecimiento de los conocimientos mundiales en favor del desarrollo permitiendo un acceso equitativo a la información para actividades económicas, sociales, políticas, sanitarias, culturales, educativas y científicas dando acceso a la información que está en el dominio público sólo son posibles gracias a las TICs.

El uso de las TIC en las empresas posibilita una mejora de los procesos productivos internos al proporcionar herramientas que facilitan la clasificación, organización, manejo y filtro de la información, y de cara al exterior mejoran los procesos de interacción con los agentes externos: clientes, proveedores y socios, permitiendo la independencia del tiempo y del espacio, posibilitando el acceso desde cualquier lugar y a cualquier hora, abriendo la posibilidad de nuevos negocios en la red.

2.2.1. Principales aportes de las TICs.

Sus principales aportes en las actividades humanas se concretan en una serie de funciones que facilitan la realización del trabajo porque, siempre requieren una cierta información para realizarlo, un determinado proceso de datos y a menudo también la comunicación con otras personas; y esto es precisamente lo que ofrecen las TIC.

- ✚ Fácil acceso a todo tipo de información, sobre cualquier tema y en cualquier formato, especialmente a través de la televisión e Internet pero también mediante el acceso a las numerosas colecciones de discos en soporte CD-ROM y DVD.
- ✚ Instrumentos para todo tipo de proceso de datos. Los sistemas informáticos, integrados por ordenadores, periféricos y programas, permiten realizar cualquier tipo de proceso de datos de manera rápida y fiable: escritura y copia de textos, cálculos, creación de bases de datos, tratamiento de imágenes, etc.
- ✚ Canales de comunicación inmediata, sincrónica y asíncrona, para difundir información y contactar con cualquier persona o institución del mundo mediante la edición y difusión de información en formato Web, el correo electrónico, los servicios de mensajería inmediata, los fórums telemáticos, las videoconferencias, etc.
- ✚ Homogeneización de los códigos empleados para el registro de la información mediante la digitalización de todo tipo de información: textual, sonora, icónica y audiovisual. (Graells 2000)

De todos los elementos que integran las TIC, sin duda el más poderoso y revolucionario es Internet, que abre las puertas de una nueva era, la Era Internet, en la que se ubica la actual Sociedad de la Información. Internet proporciona un tercer mundo en el que se puede hacer casi todo lo que se realiza en el mundo real y además permite desarrollar nuevas actividades, muchas de ellas enriquecedoras de personalidad y forma de vida (contactar con foros telemáticos y personas de todo el mundo, localización inmediata de cualquier tipo de información, teletrabajo, teleformación, teleocio). (Graells 2000)

2.3. Lenguaje Unificado de Modelado (UML)

UML (Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software. Sus creadores pretendieron con este lenguaje, unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas en un acercamiento estándar.

UML es, probablemente, una de las innovaciones conceptuales en el mundo tecnológico del desarrollo de software que más expectativa ha generado a lo largo de muchos años, comparable con la aparición e implantación de los lenguajes COBOL⁸, Basic, Pascal, C++, y actualmente con los más recientes Java, XML, C#. UML es ya un estándar de la industria del software, pero no sólo de la industria sino, que en general, de cualquier industria que requiera la construcción de modelos como condición previa para el diseño y posteriormente para la construcción de prototipos.

Además permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas que estén involucradas en el proceso de desarrollo de los sistemas, esto se lleva a cabo mediante un conjunto de símbolos y diagramas.

UML ha nacido como un lenguaje, pero es mucho más que un lenguaje de programación. En realidad se ha diseñado y construido un lenguaje que ha nacido con una madurez sólida si se le compara, incluso con los últimos desarrollos de HTML, C#, Java, Ajax, XML, los lenguajes por excelencia del mundo de la Internet. (Larman 1999)

UML ayuda a los usuarios a entender la realidad desde un punto de vista de la tecnología y la posibilidad de que reflexione antes de invertir y gastar grandes cantidades de dinero en proyectos que no estén seguros en su desarrollo, reduciendo el costo y el tiempo empleado en la construcción de los módulos que construirán el software. (RUMBAUHG 2000a)

Tiene una gran cantidad de propiedades las que han contribuido a hacer de UML el estándar de la industria en la actualidad. Algunas de las propiedades de UML como lenguaje de modelado son: (RUMBAUHG 2000b)

⁸ COmmon Business Oriented Language

- ✚ Es un lenguaje distribuido y adecuado a las necesidades de conectividades actuales y futuras. Ampliamente utilizado por la industria del software.
- ✚ Reemplaza a decenas de notaciones empleadas por otros lenguajes.
- ✚ Modela estructuras complejas.
- ✚ Las estructuras más importantes que soporta tienen su fundamento en la tecnología orientada a objeto, tales como objetos, clases, componentes y nodos.
- ✚ Comportamiento del sistema: casos de usos, diagramas de secuencia, de colaboración, que sirve para evaluar el estado de las máquinas.

Se utilizará como notación UML para lograr mayor entendimiento, pues se logra modelar y describir secuencialmente por pasos todos los procesos que se llevan a cabo según la problemática planteada. Es útil porque es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

2.4. Metodologías de desarrollo de software

Uno de los principales problemas en la actualidad en el desarrollo de software es seleccionar la metodología más adecuada que posibilite obtener los resultados óptimos que se desean; o sea, cómo trabajar eficientemente evitando las catástrofes que conllevan al fracaso de un gran porcentaje de proyectos a nivel mundial. Una metodología tiene como principal objetivo aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo.

Una metodología puede seguir uno o varios modelos de ciclo de vida. La metodología indica cómo hay que obtener los distintos productos parciales y finales. En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable. (Corporation 2006)

Existen numerosas metodologías de desarrollo de software, entre las más usadas se pueden mencionar Programación Extrema (XP), Proceso Unificado de Desarrollo de Software (RUP) y Desarrollo Guiado por Funcionalidad (FDD).

Programación Extrema (XP) es una metodología ágil, intenta reducir la complejidad del software por medio de un trabajo orientado a objeto, basado en las relaciones interpersonales y la velocidad de reacción. Intenta minimizar el riesgo de fallo del proceso por medio de la disposición permanente de un

representante competente del cliente a disposición del equipo de desarrollo. Este representante debe estar en condiciones de contestar rápida y correctamente a cualquier pregunta del equipo de desarrollo de forma que no se retrase la toma de decisiones. (MOLPECERES 2003)

El Proceso Unificado de Desarrollo de Software (RUP), es un proceso o metodología que ordenadamente concreta, define las tareas y sus responsables, no es más que una guía para utilizar el Lenguaje Unificado de Modelado (UML). RUP, es el resultado de la evolución e integración de diferentes metodologías de desarrollo de software. Permite sacar el máximo provecho de los conceptos asociados a la orientación a objetos y al modelado visual.

El Desarrollo Guiado por Funcionalidad se puede considerar a medio camino entre RUP y XP, aunque al seguir siendo un proceso ligero, se asemeja más a este último. Está pensado para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Se basa en un proceso iterativo con iteraciones cortas (2 semanas). Las iteraciones se deciden en base a funcionalidades (features⁹), que son pequeñas partes del software con significado para el cliente. (MOLPECERES 2003)

2.4.1. Fundamentación de la metodología utilizada (RUP).

La metodología de desarrollo seleccionada es RUP, pues esta actualmente lleva a la construcción de sistemas de software mucho más complejos y grandes. Debido al auge de las computadoras y a la vez del aumento del rigor de los usuarios, esto ha traído un rápido incremento en Internet para el intercambio de todo tipo de información. James Rumbaugh, Grady Booch e Ivar Jacobson, autores de “El proceso unificado de desarrollo de software”, opinan que “El problema del software se reduce a la dificultad que afrontan los desarrolladores para coordinar las múltiples cadenas de trabajo de un gran proyecto de software”. Cuenta con las mejoras prácticas del modelo de desarrollo de un software en particular. (RUMBAUHG 2000a; PRESSMAN 2002)

- ✚ Desarrollo de software de forma iterativa.
- ✚ Manejo de requerimientos.
- ✚ Utiliza arquitectura basada en componentes.
- ✚ Modela el software de forma visual, usando UML.
- ✚ Verifica la calidad del software.
- ✚ Controla los cambios.

⁹ Pequeñas partes del software con significado para el cliente

- ✚ Dirige las tareas de cada desarrollador por separado y del equipo como un todo.
- ✚ Especifica los artefactos que deben desarrollarse en cada fase de desarrollo del software.

RUP es una de estas metodologías, que al desarrollar un software se encarga muy bien de ordenar el flujo de trabajo, porque es un proceso que integra las múltiples facetas del desarrollo (tiene 9 flujos de trabajo, de los cuales 6 son ingenieriles y los demás de gestión, a su vez está integrado por 4 fases en la cuales se distribuyen las actividades a realizar, dando como resultado los artefactos¹⁰ de cada uno).

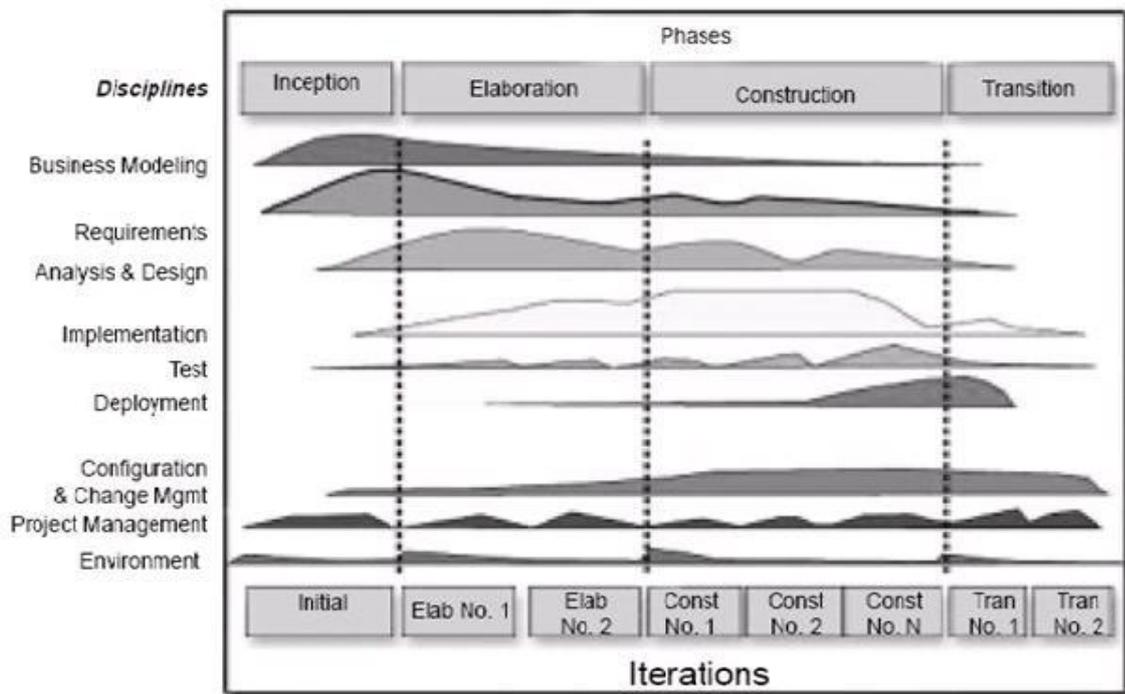


Figura 2 Disciplinas a través de las iteraciones.

En el caso específico de las aplicaciones Web, las actividades establecidas por dicho proceso son suficientes para garantizar cubrir todos los aspectos de los entornos de este tipo de aplicación. El Proceso Unificado Rational presenta una serie de características que convierte en único al proceso unificado. (RUMBAUHG 2000a)

- ✚ Dirigido por Casos de Uso
- ✚ Centrado en la Arquitectura
- ✚ Iterativo e Incremental

¹⁰ Productos tangibles

Esta metodología utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema de software. Es una parte esencial del Proceso Unificado de desarrollo de software y fueron desarrollados paralelamente por las mismas personas, haciendo que su integración sea un éxito.

2.5. Modelo Cliente – Servidor

Se dice que la arquitectura Cliente/Servidor es la integración distribuida de un sistema en red, con los recursos, medios y aplicaciones que, definidos modularmente en los servidores, administran, ejecutan y atienden las solicitudes de los clientes; todos interrelacionados física y lógicamente, compartiendo datos, procesos e información. Se establece así un enlace de comunicación transparente entre los elementos que conforman la estructura. Entre las principales características de la arquitectura Cliente/Servidor, se pueden destacar las siguientes:

- ✚ El servidor presenta a todos sus clientes una interfaz única y bien definida.
- ✚ El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- ✚ El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- ✚ Los cambios en el servidor implican pocos o ningún cambio en el cliente.

Ventajas de la arquitectura cliente-servidor: (C. G. Hernandez 2006)

- ✚ El servidor no necesita potencia de procesamiento, parte del proceso se reparte con los clientes.
- ✚ Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre.

Los programas cliente, que el usuario utiliza para acceder a los servicios de Internet, realizan dos funciones distintas. Por una parte, se encargan de gestionar la comunicación con el ordenador servidor, de solicitar un servicio concreto y de recibir los datos enviados por éste; y por otra, es la herramienta que presenta al usuario los datos en pantalla y que le ofrece los comandos necesarios para utilizar las prestaciones que ofrece el servidor.

El esquema Cliente/Servidor facilita la integración entre sistemas diferentes y comparte información permitiendo, por ejemplo que las máquinas ya existentes puedan ser utilizadas pero utilizando interfaces más amigables al usuario. De esta manera, se pueden integrar PCs con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional. (Alfaro 2007)

2.6. Arquitectura en Capas

La arquitectura de una aplicación es la vista conceptual de la estructura de ella misma. Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos. La arquitectura de las aplicaciones difiere según como está distribuido este código. (Cornejo 2001)

Las aplicaciones Web, por lo general, suelen estar estructuradas siguiendo un modelo basado en tres niveles. El primer nivel es la interfaz. Las funciones que se realizan en él se limitan únicamente a enviar las peticiones/consultas que genera el usuario, obtener los datos generados por la aplicación Web (servidor) y finalmente representar los datos en el navegador.

En el segundo nivel se sitúa la lógica. En éste se encuentra el núcleo principal de la aplicación y es el encargado de dotar a la aplicación Web del contenido dinámico.

Las funciones que pueden ser realizadas en este nivel son:

- ✚ interpretación de las consultas generadas por el cliente
- ✚ ejecución de la lógica necesaria para esas consultas
- ✚ obtención/almacenaje de datos
- ✚ interacción con otras aplicaciones Web
- ✚ envío de los datos generados al cliente

En el tercer nivel estarían los datos. Éste podría estar formado por una base de datos, por ficheros XML o por cualquier otro tipo de sistema de almacenaje de datos dependiendo de las necesidades de la aplicación.

Las ventajas que presenta el uso del estilo en capas, en específico el de 3 capas son:

- ✚ Soporta un diseño basado en niveles de abstracción crecientes.
- ✚ Permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.

- ✚ Admite optimizaciones y refinamientos.
- ✚ Proporciona amplia reutilización. Al igual que los tipos de datos abstractos.
- ✚ Se pueden utilizar diferentes implementaciones o versiones de una misma capa en la medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto conduce a la posibilidad de definir interfaces de capa estándar, a partir de las cuales se pueden construir extensiones o prestaciones específicas

De igual manera presenta desventajas como son:

- ✚ No admiten un buen mapeo en una estructura jerárquica. Incluso cuando un sistema se puede establecer lógicamente en capas.
- ✚ Las consideraciones de performance pueden requerir acoplamientos específicos entre capas de alto y bajo nivel.
- ✚ A veces es también extremadamente difícil encontrar el nivel de abstracción correcto.
- ✚ Los cambios en las capas de bajo nivel tienden a filtrarse hacia las de alto nivel, en especial si se utiliza una modalidad relajada. (Kicillof 2004)

2.7. Tecnologías del lado del cliente

Cada vez van surgiendo nuevas tecnologías que se relacionan con el desarrollo Web lo que hacen a éste más interactivo y apasionante. Entre las tecnologías utilizadas, están las que funcionan del lado del cliente y del lado del servidor, las del lado del cliente están insertadas en la página HTML del cliente, además son interpretadas y ejecutadas por el navegador, es decir, que su correcta funcionalidad depende del soporte de la versión del navegador a ser utilizado por el usuario visitante.

2.7.1. HTML

El lenguaje HTML, cuyas siglas significan Lenguaje de Formato de Documento de Hipertexto (en inglés *Hypertext Markup Language*), se utiliza para crear documentos que muestren una estructura de hipertexto. Un documento de hipertexto es aquel que contiene información cruzada con otros documentos, lo cual permite pasar de un documento al referenciado desde la misma aplicación con la que lo estamos visualizando. (C.Mateu 2004)

HTML permite, además, crear documentos de tipo multimedia, es decir, que contengan información más allá de la simplemente textual, como pueden ser imágenes, video o sonido. El lenguaje HTML no es el único lenguaje existente para crear documentos hipertexto. Hay otros lenguajes anteriores o

posteriores a HTML (SGML, XML, etc.), aunque para muchos HTML se ha convertido en el lenguaje estándar para la creación de contenido para Internet.

2.7.2. CSS

Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirá de estándar para los agentes de usuario o navegadores.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los *Estilos* definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento. (W3C 2006)

Las ventajas de utilizar CSS (u otro lenguaje de estilo) son:

- ✚ Control centralizado de la presentación de un sitio Web completo con lo que se agiliza de forma considerable la actualización del mismo.
- ✚ Los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio Web remoto, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- ✚ Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil, o ser "leída" por un sintetizador de voz.
- ✚ El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño.

2.7.3. JavaScript

JavaScript es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Al contrario que Java, JavaScript no es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de herencia, es más bien un lenguaje basado en prototipos, ya que las nuevas clases se generan

clonando las clases base (prototipos) y extendiendo su funcionalidad. Todos los navegadores interpretan el código JavaScript integrado dentro de las páginas Web. Para interactuar con una página Web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

2.8. AJAX

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript y XML asíncronos), es una técnica de desarrollo web para crear aplicaciones interactivas. Estas se ejecutan en el cliente, es decir, en el navegador del usuario, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la velocidad de interacción en la misma. (Garret 2005)

AJAX no constituye una tecnología en sí, sino que es un término que engloba a un grupo de éstas que trabajan conjuntamente y que se muestra a continuación: (Garret 2005)

- ✚ XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- ✚ Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.
- ✚ El objeto XMLHttpRequest para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto IFrame en lugar del XMLHttpRequest para realizar dichos intercambios.
- ✚ XML es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML Preformateado, texto plano y JSON.

2.8.1. Ventajas del AJAX

Interactividad

Las aplicaciones AJAX se ejecutan en la máquina del usuario, manipulando la página actual dentro de sus navegadores usando métodos de Document Object Model. Puede ser usado para multitud de tareas como actualizar o eliminar registros, expandir formularios web, devolver peticiones simples de búsqueda, o editar árboles de categorías; todo sin tener la necesidad de recargar toda la página de HTML cada vez que se realiza un cambio. Generalmente sólo requiere enviar pequeñas peticiones al servidor, y se devuelven respuestas relativamente cortas. Esto permite el desarrollo de aplicaciones interactivas con más interfaces de usuario más responsivas gracias al uso de las técnicas DHTML. (Garret 2005)

Portabilidad

Las aplicaciones construidas con AJAX utilizan características bien documentadas presentes en todos los navegadores importantes en la mayoría de las plataformas existentes. Aunque esta situación podría cambiar en el futuro, en este momento, los usos de AJAX son efectivos entre plataformas. Mientras que la plataforma de AJAX está más restringida que la plataforma de Java, las aplicaciones actuales de AJAX llenan con eficacia la parte de los Java Applets: ampliar el navegador con mini-aplicaciones ligeras.

2.8.2. Desventajas del AJAX

Críticas de usabilidad

Una de las mayores críticas contra el uso de AJAX en aplicaciones web es que puede fácilmente acabar con el comportamiento normal del botón atrás del navegador. Las diversas expectativas entre volver a una página que se ha modificado dinámicamente y la vuelta a una página estática pueden ser sutiles. Otro problema relacionado es que las actualizaciones dinámicas hacen difícil al usuario agregar a los marcadores/favoritos un momento particular de la aplicación.

JavaScript

Aunque AJAX no necesita ningún tipo de plug-in para el navegador, requiere que los usuarios tengan el JavaScript activado. Como ocurre con las aplicaciones DHTML, las de AJAX deben de ser probadas rigurosamente para adaptarse a los diferentes navegadores y plataformas.

2.9. XML

XML (Extensible Markup Language), es un estándar que lo constituye un conjunto de reglas para definir etiquetas semánticas que organizan un documento en diferentes partes. XML es un metalenguaje que define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados.

Es un Lenguaje de etiquetado extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Es un lenguaje muy similar a HTML pero su función principal es describir datos y no mostrarlos como es el caso de HTML. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones. Las tecnologías XML son un conjunto de módulos que ofrecen servicios útiles a las demandas más frecuentes por parte de los usuarios. XML sirve para estructurar, almacenar e intercambiar información.

Principales características de XML:

- ✚ Integración de los datos de las fuentes más dispares. Puede hacer el intercambio de documentos entre las aplicaciones tanto en la propia PC como en una red local o extensa.
- ✚ Datos compuestos de múltiples aplicaciones. La extensibilidad y flexibilidad de este lenguaje permite agrupar una variedad amplia de aplicaciones, desde páginas web hasta bases de datos.
- ✚ Gestión y manipulación de los datos desde el propio cliente web.
- ✚ Los motores de búsqueda devolverán respuestas más adecuadas y precisas, ya que la codificación del contenido web en XML consigue que la estructura de la información resulte más accesible.
- ✚ Se desarrolla de manera extensible las búsquedas personalizables y subjetivas. También conlleva a que los clientes web puedan ser más autónomos para desarrollar tareas que actualmente se ejecutan en el servidor.

2.10. Lenguajes de programación del lado del servidor.

Estas tecnologías pueden o no estar insertadas dentro de la página HTML. A diferencia de las tecnologías Web del lado del cliente, estas tecnologías no dependen del navegador ya que son interpretadas y ejecutadas por el servidor. Existen varios lenguajes de programación de este tipo, entre los más populares está Active Server Page (ASP), Java Server Page (JSP) y Personal Home Page (PHP).

ASP responde al nombre Active Server Page, que significa Páginas de servidores activas. Con esta tecnología creada por Microsoft, el usuario de Internet puede recibir páginas dinámicas en el servidor. Es parte del Internet Information Server (IIS) desde la versión 3.0 y es una tecnología de páginas activas que permite el uso de diferentes scripts y componentes en conjunto con el tradicional HTML para mostrar páginas generadas dinámicamente.

JSP es un acrónimo de Java Server Pages, que significa Páginas de Servidor Java. Es una tecnología orientada a crear páginas Web con programación en Java. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP se pueden escribir con el editor HTML/XML habitual.

PHP, es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. El PHP originalmente diseñado en Perl, seguidos por la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador Danés-Canadiense Rasmus Lerdorf en el año 1994 para mostrar su currículum vital y guardar ciertos datos, como la cantidad de tráfico que su página web recibía.

La versión más reciente de PHP es la 5.1.4, que incluye el novedoso PDO (PHP Data Objects) y mejoras utilizando las ventajas que provee el nuevo Zend Engine 2. Según estudios, más de un millón de servidores tienen esta capacidad implementada y los números continúan creciendo. (C. G. Hernandez 2006)

La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí misma. La meta del lenguaje es permitir rápidamente a la comunidad de desarrolladores la generación dinámica de páginas Web

2.10.1. Fundamentación del lenguaje del lado del servidor a utilizar.

El lenguaje del lado del servidor seleccionado para desarrollar la aplicación es PHP5 debido a sus características, pues estas proporcionan diversas ventajas a tener en cuenta respecto a los demás lenguajes mencionados.

Una de las características más potentes es su soporte para gran cantidad de bases de datos. Entre las que se pueden mencionar InterBase, mSQL, MySQL, Oracle, Informix, PostgreSQL, entre otras. PHP también ofrece la integración con varias bibliotecas externas, que dan al desarrollador la posibilidad de realizar cualquier tarea, desde generar documentos en pdf (Portable Document Format) hasta analizar

código XML y últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la librería GTK+.

Es software libre, lo que implica menos costes y servidores más baratos que otras alternativas. Es muy rápido y su integración con la base de datos PostgreSQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado.

Es multiplataforma, funciona tanto para Unix (con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene porqué modificarse al pasar a la otra.

Su sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si se está familiarizado con esta sintaxis, le resultará muy fácil aprender PHP. Su librería estándar es realmente amplia, lo que permite reducir los llamados "costes ocultos", uno de los principales defectos de ASP (Active Server Pages).

PHP tiene una de las comunidades más grandes en Internet, con lo que no es complicado encontrar ayuda, documentación, artículos, noticias, y más recursos. Ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y ponerse al día, a diferencia con el código de otros lenguajes. (C. G. Hernandez 2006)

Debido a su amplia distribución PHP está perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.

En ningún caso se envía código PHP al navegador, por lo que todas las operaciones realizadas son transparentes al usuario, el código PHP es ejecutado en el servidor y el resultado es enviado al navegador. El resultado es normalmente una página HTML. Por lo que al usuario le parecerá que está visitando una página HTML que cualquier navegador puede interpretar.

Se han realizado cambios estructurales en el lenguaje para ofrecer innovación en el nuevo PHP 5 y solucionar muchos de los problemas encontrados en PHP 4. Afortunadamente, lo nuevo de PHP 5 mejora muchas áreas en el lenguaje y su ejecución, como por ejemplo: (Mandrake 2004)

- ✚ Programación orientada a objetos (OOP).
- ✚ XML.
- ✚ Integración nativa con el Zend Engine.

Los diseñadores de PHP5 han realizado un cambio radical en el tratamiento de las variables objeto: en PHP5 todas las variables que nombran objetos son en realidad referencias. No hay que usar el operador '&' ni en las asignaciones, ni en el paso de parámetros que son objetos, ahorrándose con ello gran cantidad de potenciales errores. (Cantero 2006)

La principal novedad en las clases de PHP5 es la inclusión de modificadores de control de acceso para implementar la encapsulación (piedra angular en la programación orientada a objetos de la que adolecía PHP4).

PHP5 introduce tres palabras clave (public, private y protected) que sustituyen a var en la definición de variables miembro (atributos) de la clase, y que preceden a la definición de funciones miembro (métodos).

Otros lenguajes como Perl (Practical Extraction and Report Language), ASP (Active Server Pages) y JSP (Java Server Pages) tienen características similares al PHP aunque poseen rasgos que los marcan y por ello los distingue, entre ellos se puede encontrar: (C. G. Hernandez 2006)

- ✚ **Multiplataforma:** Menos el ASP, que es solamente aportado por la plataforma Windows, los demás lenguajes están soportados en múltiples plataformas.
- ✚ **Velocidad de ejecución:** la velocidad es mayor en PHP, seguidos por PERL y JSP.
- ✚ **Disponibilidad de recursos:** actualmente los más utilizados en la Internet son el PHP y el JSP, siendo más utilizado en la publicación de artículos y códigos de ejemplos. PHP tiene una de las comunidades más grandes en Internet, al igual que la de Java.
- ✚ **Familiaridad con el lenguaje:** En la universidad los lenguajes más utilizados por los programadores es el ASP y el PHP.

De acuerdo a estas comparaciones, el PHP resulta mucho más favorecido, por tanto, es el adecuado para implementar la propuesta de sistema de este trabajo, particularmente PHP5.

2.11. Servidor Web Apache

Apache es un servidor web de código libre robusto cuya implementación se realiza de forma colaborativa, con prestaciones y funcionalidades equivalentes a las de los servidores comerciales. Hoy en día es el servidor web más utilizado del mundo, encontrándose muy por encima de sus competidores, tanto gratuitos como comerciales. Es un software de código abierto que funciona sobre cualquier plataforma, Apache es uno de los servidores Web más potentes del mercado, ofreciendo una perfecta combinación entre estabilidad y sencillez. Las principales características de Apache son:

- ✚ Funcionalidad en múltiples plataformas.
- ✚ Elaborado índices de directorios.
- ✚ Soporte del último protocolo http 1.1.
- ✚ Sencilla administración basada en la configuración de un único archivo.
- ✚ Soporte para CGI (Common Gateway Interface).

El servidor Apache es un software que está estructurado en módulos, es decir, está dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor web. Esta modularidad es intencionada ya que la configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías: (C. G. Hernandez 2006)

- ✚ Módulos Base: Módulo con las funciones básicas del Apache.
- ✚ Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- ✚ Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiprocesos para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. El resto de funcionalidades del servidor se consigue por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software.

2.12. Sistemas de Gestión de Bases de Datos.

Un Sistema de Gestión de Bases de Datos (SGBD) puede definirse como un paquete generalizado de software, que se ejecuta en un sistema computacional anfitrión, centralizando los accesos a los datos y actuando de interfaz entre los datos físicos y el usuario. Las principales funciones que debe cumplir un SGBD se relacionan con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad.

Un SGBD tiene los siguientes objetivos específicos: (C. G. Hernandez 2006)

- ✚ Independencia de los datos y los programas de aplicación.
- ✚ Minimización de la redundancia.
- ✚ Integración y sincronización de las bases de datos.
- ✚ Integridad de los datos.
- ✚ Seguridad y protección de los datos.
- ✚ Facilidad de manipulación de la información.
- ✚ Control centralizado.

Entre los SGBD comúnmente utilizados en el mundo se tienen Oracle, MySQL, Microsoft SQL Server, PostgreSQL, Interbase, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional.

Como SGBD se seleccionó el PostgreSQL por las ventajas que ofrece y por requerimientos del cliente.

2.12.1. Fundamentación de la selección de PostgreSQL como SGBD.

PostgreSQL es un sistema de gestión de bases de datos Objeto –Relacionales (ORDBMS) libre, liberado bajo la licencia BSD (Berkeley Software Distribution). Es una alternativa a otros sistemas de bases de datos de código abierto (como MySQL, Firebird y MaxDB), así como sistemas propietarios como Oracle y SQLServer. El mismo ha sido desarrollado de varias formas desde 1977.

En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgre fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. PostgreSQL está ampliamente considerado como el

sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre.

La siguiente es una breve lista de algunas de esas características, a partir de PostgreSQL 7.1.x.

- ✚ **DBMS Objeto-Relacional:** PostgreSQL aproxima los datos a un modelo objeto -relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transacciones, optimización de consultas, herencia, y arreglos.
- ✚ **Altamente Extensible:** PostgreSQL soporta operadores, funciones métodos de acceso y tipos de datos definidos por el usuario.
- ✚ **Soporte SQL Comprensivo:** PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92.
- ✚ **Integridad Referencial:** PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- ✚ **Lenguajes Procedurales:** PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL.

Las principales mejoras en PostgreSQL incluyen:

- ✚ Los bloqueos de tabla han sido sustituidos por el control de concurrencia multiversión, el cual permite a los accesos de sólo lectura continuar leyendo datos consistentes durante la actualización de registros y permite copias de seguridad en caliente, mientras la base de datos permanece disponible para consultas.
- ✚ Se han implementado importantes características del motor de datos, incluyendo subconsultas, valores por defecto, restricciones a valores en los campos y disparadores (triggers).
- ✚ Se han añadido características adicionales que cumplen el estándar SQL92, incluyendo claves primarias, identificadores entrecomillados, forzado de tipos cadenas literales, conversión de tipos y entrada de enteros binarios y hexadecimales.
- ✚ Los tipos internos han sido mejorados, incluyendo nuevos tipos de fecha/hora de rango amplio y soporte para tipos geométricos adicionales.
- ✚ La velocidad del código del motor de datos ha sido incrementada aproximadamente en un 20-40%, y su tiempo de arranque ha bajado el 80% desde que la versión 6.0 fue lanzada.

2.13. Herramientas a utilizar.

2.13.1. Frameworks: Symfony

Un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. (Potencier and Zaninotto 2007)

Existen diferentes framework para PHP como cakePHP, Kumbia, PHP Prado, CodeIgniter y Symfony entre otros. Para el desarrollo de la propuesta de solución se ha seleccionado Symfony.

Symfony es un completo framework, desarrollado por Fabien Potencier, presidente de Sensio una empresa francesa de desarrollo de aplicaciones web conocida por sus innovaciones en este campo, y diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web. (Potencier and Zaninotto 2007)

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas (Unix, Linux, etc.) como en plataformas Windows.

Características que hace de Symfony el framework más óptimo a utilizar en la propuesta de solución: (Potencier and Zaninotto 2007)

- ✚ Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows así como en Unix y Linux)

- ✚ Independiente del sistema gestor de bases de datos, pues posee librerías encargadas de manejar la abstracción de datos.
- ✚ Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- ✚ Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador sólo debe configurar aquello que no es convencional.
- ✚ Sigue la mayoría de las mejores prácticas y patrones de diseño para la Web.
- ✚ Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- ✚ Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- ✚ Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.

2.13.2. Visual Paradigm

Visual Paradigm es una poderosa herramienta CASE que al igual que el Rational Rose utiliza UML para el modelado, es la herramienta por excelencia para ser utilizada en un ambiente de software libre. Permite crear tipos diferentes de diagramas en un ambiente totalmente visual. Es muy sencillo de usar, fácil de instalar y actualizar. Genera código para varios lenguajes. Tiene integrado el MS Visio y es compatible con otras ediciones. Posibilita un entorno de creación de diagramas para UML 2.0. (Paradigm, V 2007)

Como principales características presenta:

- ✚ Licencia: Gratuita y Comercial.
- ✚ Producto de calidad.
- ✚ Soporta aplicaciones Web.
- ✚ Varios idiomas.
- ✚ Generación de código para Java y exportación como HTML.
- ✚ Fácil de instalar y actualizar.
- ✚ Compatibilidad entre ediciones.

Visual Paradigm ofrece:

- ✚ Un entorno de creación de diagramas para UML 2.0.
- ✚ Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- ✚ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✚ Capacidades de ingeniería directa e inversa.
- ✚ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✚ Disponibilidad de múltiples versiones, para cada necesidad.
- ✚ Disponibilidad de integrarse en los principales IDEs.
- ✚ Disponibilidad en múltiples plataformas.

Posibilita la representación gráfica de los diagramas permitiendo ver el sistema desde diferentes perspectivas, como el de componentes, despliegue, secuencia, casos de uso, clase, actividad, estado, entre otros. Además, identifica requisitos y comunica información, se centra en cómo los componentes del sistema interactúan entre ellos, sin entrar en detalles excesivos, además, permite ver las relaciones entre los componentes del diseño y mejora la comunicación entre los miembros del equipo usando un lenguaje gráfico. (Castellano. 2007)

Tiene disponible distintas versiones: Enterprise, Professional, Standard, Modeler, Personal y Community. Facilita licencias especiales para fines académicos.

2.13.3. Eclipse

Eclipse es una plataforma de software de código abierto, independiente además de una plataforma para el desarrollo de proyectos conocidos como "Aplicaciones de Cliente Enriquecido". Esta plataforma, típicamente ha sido usada para desarrollar Entornos de Desarrollo Integrados (del inglés IDE).

Eclipse fue desarrollado originalmente por IBM como el sucesor de VisualAge. Ahora es desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Algunas ventajas de Eclipse son las siguientes:

- ✚ Muestra la estructura en métodos y propiedades de las clases, así como la herencia de las mismas.
- ✚ Inserta bloques PHPDoc útiles para comentarios.
- ✚ Autocompleta llaves, paréntesis, tabula el código de acuerdo a reglas predefinidas.
- ✚ Realiza compilación en tiempo real.
- ✚ Reconoce los posibles errores de código PHP, mostrando detalles sobre estos.
- ✚ Se integra muy bien con el framework Symfony, realizando incluso, el completamiento de código de las funciones que brinda el framework.
- ✚ Existe una gran comunidad de programadores que utilizan el IDE.

Desventajas de utilizar este IDE:

- ✚ Consume más recursos de hardware.
- ✚ Se necesita de cierto aprendizaje para su uso.

2.13.4. Subversion

Subversión es un sistema centralizado de control de versiones (VCS, por sus siglas en inglés) que permite realizar el seguimiento de los cambios en archivos empleados en proyectos de software. Basado en el trabajo inicial de CVS (Concurrent Versions System), constituye una implementación más eficiente de este último y ha conseguido desplazarlo en proyectos de software libre y abierto, y en organizaciones empresariales.

2.13.5. DotProject

DotProject fue creado con el objetivo de proporcionar una herramienta para la Gestión de Proyectos. Está construido por aplicaciones de Código Abierto y es mantenida por un pequeño grupo de voluntarios. Es una aplicación basada en tecnología Web, multiusuario, soporta varios lenguajes y es Software libre.

Está programada en PHP, y utiliza MySQL como base de datos (aunque otros motores como PostgreSQL también pueden ser utilizados). La plataforma recomendada para utilizar DotProject se denomina LAMP (Linux + Apache + MySQL + PHP). De todas formas, existen binarios para instalar

DotProject bajo otros sistemas operativos tales como Microsoft Windows(NT, 2000, XP) y Mac. (Alvarez 2007)

2.13.6. Gestor de referencias: EndNote

El EndNote permite crear, mantener, organizar, dar forma a bibliografías o referencias utilizadas durante un proceso de investigación, en fin, permite:

- ✚ Realizar la referencia bibliográfica adecuadamente.
- ✚ Insertar citas, referencias después de tener agrupadas todos los datos de la bibliografía consultada.
- ✚ Puede utilizar las referencias individualmente o como un juego completo.
- ✚ Puede exportar desde Internet o bases de datos consultadas hacia el propio manejador.

EndNote 9 amplía la posibilidad de recoger datos desde gran cantidad de fuentes online, es también muy rápido, añade compatibilidad con las nuevas versiones de Microsoft Word, contiene nuevos registros sintácticos y codificaciones de texto.

2.14. Conclusiones parciales.

Luego del estudio de las tecnologías a utilizar, se definió:

- ✚ RUP representa la guía más óptima para el desarrollo de la aplicación propuesta.
- ✚ El lenguaje de programación PHP 5 es viable para la implementación de la aplicación Web propuesta así como la selección de Eclipse como IDE de desarrollo.
- ✚ El PostgreSQL es el SGBD más factible para integrarlo con el framework Symfony.
- ✚ El EndNote 9.0 facilita el trabajo con la bibliografía utilizada.

CAPÍTULO 3 PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

3.1. Introducción

En el presente capítulo se describen los procesos de negocio que tienen relación con el objeto de estudio, se modela el negocio propuesto, se identifican los actores, trabajadores, los casos de uso correspondientes y se realiza la descripción de los mismos. Se enumeran los requisitos; tanto Funcionales como No Funcionales que debe cumplir el sistema que se propone y lograr una mejor concepción de forma general del mismo, se identifica mediante un Diagrama de Casos de Uso las relaciones de los actores que interactúan con el sistema y las secuencias de acciones con las que interactúan.

3.2. Modelo de Negocio.

3.2.1. Reglas generales del Negocio

- ✚ El técnico es el encargado de entregar el archivo de metadato documentado al grupo de revisores.
- ✚ El archivo de metadato al ser revisado por el grupo de revisores se debe emitir un certificado de aprobación que valide la revisión.
- ✚ El especialista encargado de archivar el metadato debe haber sido aprobado previamente por una comisión para realizar este proceso.
- ✚ Los metadatos archivados en la base de datos deben ser clasificados según el nivel de seguridad y restricción que tengan los mismos.

3.2.2. Descripción de los actores y trabajadores del negocio.

Un actor del negocio es cualquier individuo, grupo, organización, máquina o sistema de información externo que interactúa con el negocio. El término actor significa el rol que algo o alguien juega cuando interactúa con el negocio para beneficiarse de sus resultados. De acuerdo con esta idea un actor del negocio representa un tipo particular de usuario del negocio más que un usuario físico, ya que varios usuarios físicos pueden realizar el mismo papel en relación al negocio, o sea, ser instancias de un mismo actor.

Actor	Descripción
Técnico	El técnico es el encargado de iniciar las acciones de revisar el metadato y archivar metadato, al mismo tiempo es el principal beneficiado con el resultado de los procesos de negocio. Se utiliza además para representar a la persona encargada de entregar el archivo metadato al grupo de revisores y luego de ser revisado al especialista para actualizar la BD de metadatos.

Tabla 1 Actor del Negocio.

Un trabajador define el comportamiento y las responsabilidades de un individuo que actúa en el negocio, realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio.

Trabajador	Descripción
Especialista	Es el encargado de archivar el metadato una vez que ha sido aprobado. No se beneficia en ningún momento de las acciones realizadas en el proceso de negocio, sino que se limita a ejecutar dichas acciones.
Grupo de Revisores	Es la entidad encargada de revisar y aprobar los archivos antes de llevarlos a la BD, basados en estándares y/o normas de calidad. No se beneficia en ningún momento de las acciones realizadas en el proceso de negocio, sino que se limita a ejecutar dichas acciones.

Tabla 2 Trabajadores del Negocio.

3.2.3. Modelo de Casos de Uso del Negocio

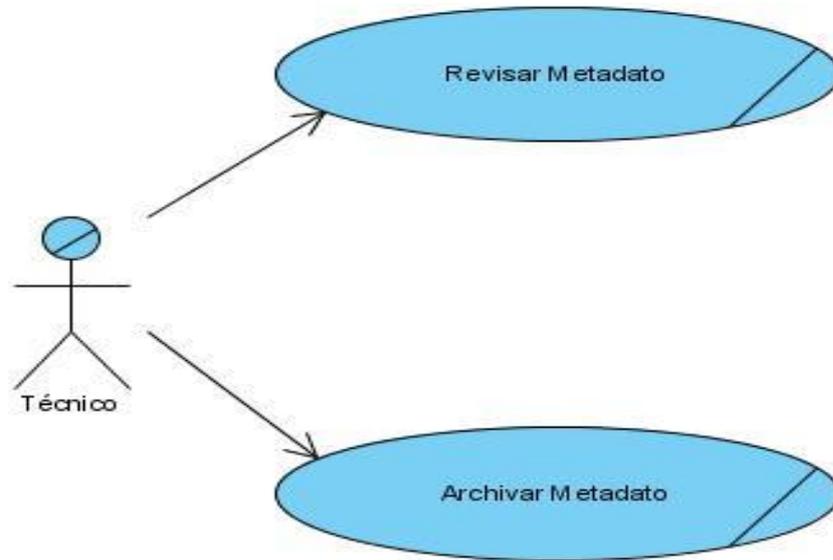


Figura 3 Modelo de Casos de Uso del Negocio.

3.2.4. Descripción textual de los Casos de Uso del Negocio.

Caso de uso del negocio	Revisar Metadato.
Actores	Técnico.
Resumen	El caso de uso comienza cuando el técnico presenta el archivo metadato para ser revisado, el Grupo de Revisores da revisión a dicho archivo velando por el cumplimiento de las normas ISO para el metadato y entrega un certificado que avala la calidad del mismo, terminado así el caso de uso.
Casos de uso asociados	.
Precondiciones	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del proceso de negocio

1- El técnico solicita al Grupo de Revisores la revisión del archivo metadato.	1.1- El Grupo de Revisores solicita la entrega del archivo de metadato.
2- El técnico hace entrega del archivo creado.	2.1- El Grupo de Revisores hace la revisión del archivo metadato haciendo uso de los estándares de documentación del metadato. 2.2- Si el archivo está correcto se crea un certificado que avale su calidad y se entrega.
3- El técnico recibe el archivo revisado y su certificación.	
Flujos Alternos	
Acción del Actor	Respuesta del proceso de negocio
	2.2- Si el archivo no está correcto se deniega y se le entrega al técnico para su correcta elaboración.
4- El técnico recibe el archivo y se marcha.	
Poscondiciones	Queda revisado el archivo metadato.
Mejoras propuestas	Este proceso no entra en el proyecto de informatización.

Tabla 3 Descripción textual CUN Revisar Metadato.

Caso de uso del negocio	Archivar Metadato.
Actores	Técnico.
Resumen	El caso de uso comienza cuando el técnico solicita archivar en la BD el archivo metadato, el especialista verifica su certificado de calidad y procede a ejecutar la acción de archivar, terminando así el caso de uso.
Casos de uso asociados	.
Precondiciones	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del proceso de negocio
1- El técnico solicita al especialista archivar metadato.	1.1- El especialista recoge el archivo metadato y su certificación. 1.2- El especialista verifica la validez del certificado. 1.3- Si el certificado es fidedigno entonces se archiva en la BD. 1.4- Después de almacenado el especialista informa del éxito del proceso.
Flujos Alternos	
Acción del Actor	Respuesta del proceso de negocio
2- El técnico recibe el archivo y se marcha.	1.3- Si el certificado no es fidedigno se le entrega al técnico el archivo para su revisión con el Grupo de Revisores.
Poscondiciones	Queda archivado en la BD el metadato.
Mejoras propuestas	Este sistema mejora el proceso de captación de fichero, haciéndolo más eficiente y seguro.

Tabla 4 Descripción textual CUN Archivar Metadato.

3.2.5. Diagrama de actividades para cada CU del Negocio

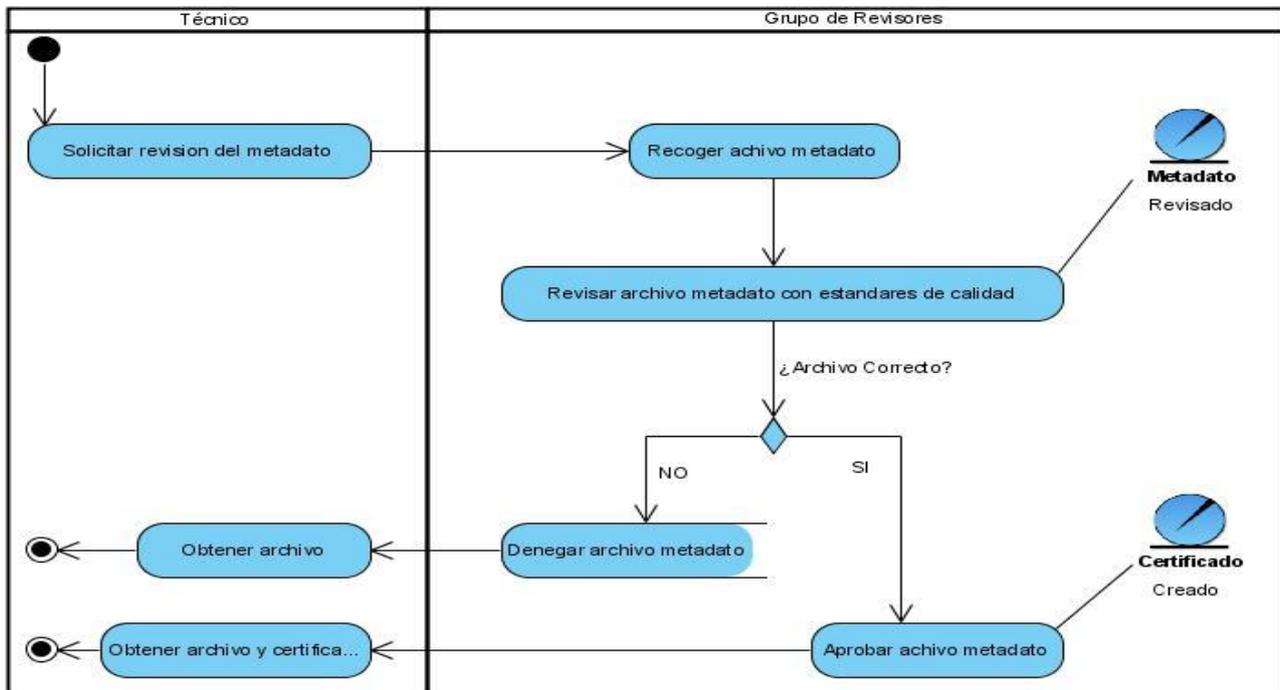


Figura 4 Diagrama de actividades para CUN Revisar Metadato.

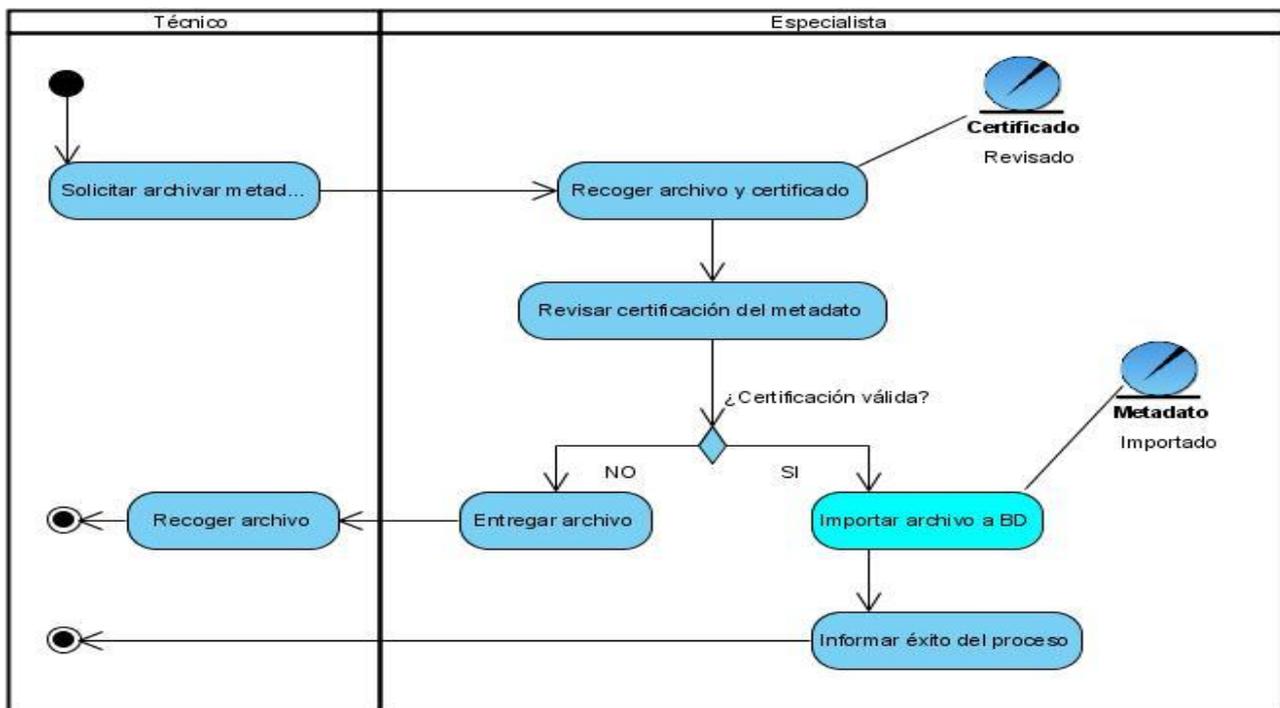


Figura 5 Diagrama de actividades del CUN Archivar Metadato.

3.2.6. Modelo de Objetos.

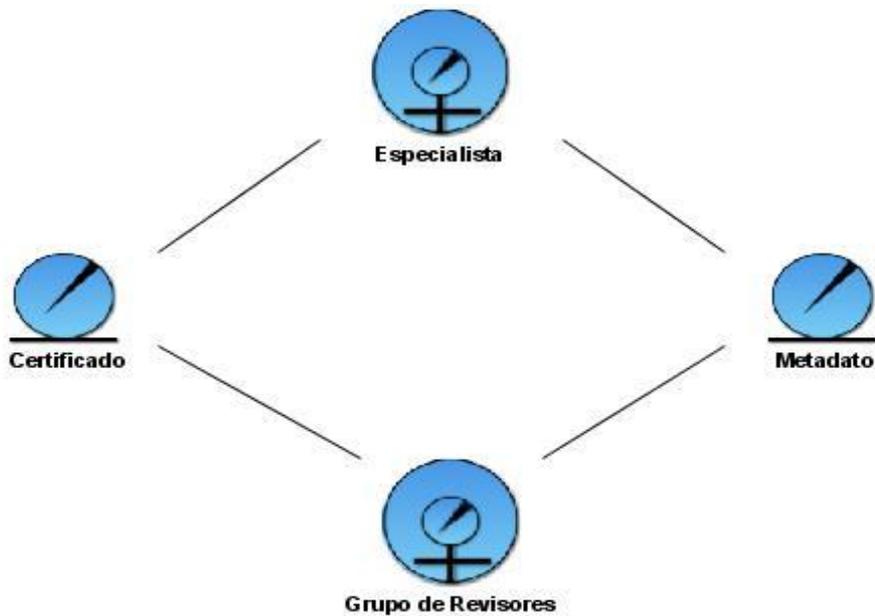


Figura 6 Modelo de Objetos.

3.3. Requisitos del Sistema.

3.3.1. Requerimientos Funcionales.

Los requerimientos funcionales son aquellos requisitos que, desde el punto de vista de las necesidades del usuario debe cumplir el sistema y que están fuertemente ligados a las opciones del programa. Teniendo en cuenta el objetivo planteado se enumeran los requerimientos funcionales de la aplicación, donde el sistema debe ser capaz de:

- R 1. Autenticar usuario.
- R 2. Actualizar registros de datos.
- R 3. Emitir reportes del estado del Metadato.
- R 4. Importar archivo de Metadato.
- R 5. Mostrar los Logs del sistema.
- R 6. Salvar los Logs del sistema.

R 7. Eliminar los Logs del sistema.

R 8. Brindar servicio de las preguntas más frecuentes.

R 9. Brindar asistencia técnica a los usuarios del sistema.

R 10. Agregar preguntas al servicio de preguntas frecuentes.

R 11. Eliminar preguntas del servicio de preguntas frecuentes.

R 12. Actualizar asistencia técnica.

3.3.2. Requerimientos No Funcionales

Son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido y confiable.

Apariencia o interfaz interna:

La interfaz no debe contener muchas imágenes para no demorar las respuestas al usuario. El diseño de la interfaz debe ser sencillo, formal, serio y con una navegación sugerente.

Usabilidad:

La aplicación Web será utilizada por los trabajadores que harán posible que la solicitud de los actores se lleve a cabo, por lo que se requerirá ser usuario y tener una clave para entrar al sistema y realizar las actividades de administración.

Rendimiento:

La disponibilidad de trabajo en red contra el servidor debe ser constante. Se garantizará que la respuesta a las solicitudes de los usuarios del sistema sea en un período de tiempo breve (de segundos) para evitar la acumulación de trabajo por parte de los responsables. El sistema deberá ser lo más estable y confiable posible. El servicio siempre debe mantenerse activo y con opciones de recuperación ante cualquier fallo de conexión.

Disponibilidad:

El sistema deberá estar disponible las 24 horas todos los días del año y soportar un gran número de conexiones simultáneas.

Portabilidad:

La aplicación Web podrá ser usada bajo cualquier versión del sistema operativo Linux, para su implementación se emplearon herramientas de programación y gestión de bases de datos que pertenecen a Software Libre. Además debe permitir que pueda ser consultado desde cualquier plataforma.

Soporte:

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra. El sistema propuesto debe ser fácil de instalar y configurar. Además debe brindar manuales de instalación y de soporte técnico.

Software:

Para el funcionamiento del sistema en el servidor, es necesario el Sistema Operativo Linux en sus versiones adecuadas para servidor.

- ✚ Cliente
 - Navegador Internet Explorer 4.0 o superior, o compatible con Mozilla.
 - Sistema operativo Linux ó Windows en algunas de sus versiones.
- ✚ Servidor de aplicaciones
 - Apache Web Server para Linux.
 - Sistema operativo Linux (Ubuntu 7.10)
- ✚ Servidor de Base de Datos
 - Sistema operativo Linux
 - PostgreSQL 8.2

Hardware:

- ✚ Cliente
 - 128 MB de memoria RAM, puede contar con un cliente ligero.
- ✚ Servidor de aplicaciones
 - Procesador Pentium IV.
 - 1 GB de memoria RAM.
 - 40 GB de disco duro.
- ✚ Servidor de Base de Datos
 - Procesador Pentium IV.
 - 1 GB de memoria RAM.
 - 120 GB de disco duro.

Seguridad:

El sistema se encarga de controlar los diferentes niveles de acceso y funcionalidad de usuarios al sitio, de identificar al usuario antes de que pueda realizar cualquier acción sobre el sistema. Garantiza que la información sea vista únicamente por quien tiene derecho a verla según el rol que este tenga en la aplicación.

Existe un primer nivel o nivel básico donde están las funciones asociadas al usuario general o común, que requieren solo la posibilidad de realizar las búsquedas. El segundo nivel está compuesto por funciones que las puede realizar el especialista. El tercer nivel está conformado con las funciones administrativas del sitio y del sistema por tanto es el nivel de mayor complejidad.

Se hacen validaciones de la información en el servidor, no obstante los usuarios acceden de manera rápida y operativa al sistema sin que los requerimientos de seguridad se conviertan en un retardo para ellos.

Legales:

El sistema se basa en un estándar que se rige por normas internacionales y cumple con las normas y leyes establecidas en nuestro país y a nivel internacional. La plataforma escogida para el desarrollo de la aplicación, está basada en la licencia GNU/GPL. Las normas ISO utilizadas son: ISO 19115 y 19139 regidas por los estándares internacionales para la documentación del Metadato.

Confiabilidad:

La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores. La aplicación constará con los procedimientos de recuperación ante fallos y errores propios del sistema, será ajeno al resto de errores de otros componentes que interactúen con el sistema y estos serán debidamente documentados. Se mantendrá un control adecuado de los Logs del sistema.

Procedimientos:

El sistema debe constar con manuales de procedimientos que aseguren el buen funcionamiento de los RF del sistema.

3.4. Modelamiento del Sistema.

3.4.1. Descripción de los actores del sistema.

Actor	Descripción
Usuario (usuario genérico)	Son aquellas personas que tienen acceso ha determinado grupo de acciones dentro del sistema.
Especialista	Es el encargado de la administración del sistema, de hacer posible que se gestione el metadato y brindar los reportes solicitados.
Subsistema Seguridad	Es el encargado de verificar la existencia en la BD de los usuarios que deciden autenticarse.

Tabla 5 Actores del Sistema.

3.4.2. Diagrama de Casos de Usos del Sistema

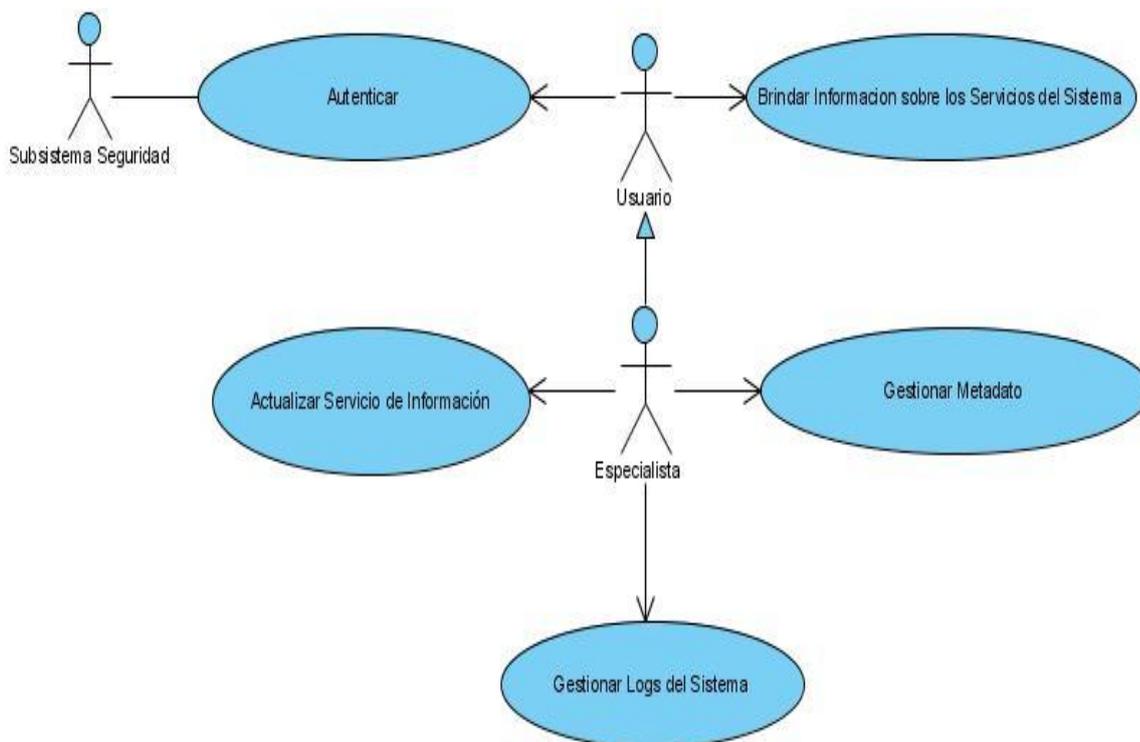


Figura 7 Diagrama de Casos de Uso del Sistema.

3.4.3. Descripción de los Casos de Uso del Sistema (CUS).

Caso de Uso:	Autenticar.
Actores:	Usuario.
Resumen:	El caso de uso se inicia cuando el actor Usuario decide acceder a la sesión administrar. El sistema captura sus datos enviándolos al Subsistema Seguridad para ser validados, de ser correctos el usuario accede a la sección de administración del sistema.
Precondiciones:	.
Referencias	R1
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El usuario decide autenticarse en el sistema.	1.1- El sistema muestra el formulario de autenticación exigiendo usuario y contraseña.
2- El usuario introduce sus datos.	2.1- El sistema captura los datos entrados por el usuario y son enviados a Subsistema Seguridad. 2.2- El sistema recibe respuesta de que el usuario existe en BD y sus datos son válidos. 2.3- Verifica si su rol tiene acceso a esta sesión. 2.4- Brinda acceso.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
	2.2- El sistema recibe respuesta con los datos ya validados, si no son correctos o no existe en la BD se muestra un mensaje de error y retorna al paso 3.

	2.3- Si su rol no tiene permiso para acceder a esta funcionalidad se muestra un mensaje de error.
Poscondiciones	El usuario ha sido autenticado y accede a las funcionalidades que tiene permiso.

Tabla 6 Descripción CUS "Autenticar".

Caso de Uso:	Brindar Información sobre los Servicios del Sistema.
Actores:	Usuario.
Resumen:	El caso de uso se inicia cuando el actor Usuario solicita al sistema información sobre los servicios del Metadato o solicita el servicio de preguntas frecuentes, el sistema en dependencia de la opción deseada muestra el servicio.
Precondiciones:	
Referencias	R8, R9.
Prioridad	Media.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El usuario entra en la sección servicios del sistema.	1.1- El sistema muestra las opciones al especialista: a) Asistencia Técnica. b) Servicio de Preguntas.
2- El usuario elige la opción deseada.	
Sección "Asistencia Técnica"	
Acción del Actor	Respuesta del Negocio
2- El usuario elige la opción de asistencia técnica.	2.1- El sistema muestra información actualizada de los servicios que brinda el metadato.
Sección "Servicio de Preguntas"	
Acción del Actor	Respuesta del Negocio
2- El usuario elige la opción de servicio de preguntas.	2.1- El sistema muestra el servicio de preguntas sobre el metadato.

Poscondiciones	Se muestra información actualizada de los servicios del sistema o del servicio de preguntas sobre el metadato.
-----------------------	--

Tabla 7 Descripción CUS "Brindar información sobre los servicios del sistema".

Caso de Uso:	Gestionar Metadato.
Actores:	Especialista.
Resumen:	El caso de uso se inicia cuando el Especialista solicita gestionar el metadato. El sistema permite actualizar, emitir reporte o importar un archivo metadato en dependencia de la solicitud, al finalizar cada acción se muestra un mensaje informando sobre el éxito del proceso.
Precondiciones:	El usuario debe estar autenticado. Cuando se va actualizar o emitir un reporte el metadato debe existir, no necesariamente al importar el archivo.
Referencias	R2, R3, R4.
Prioridad	Crítico.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Negocio
1- El especialista entra en la sección gestionar metadato.	1.1- El sistema muestra las opciones al especialista: a) Actualizar Metadato. b) Emitir Reporte. c) Importar Archivo.
2- El especialista elige la opción deseada.	

Sección "Actualizar Metadato"

Acción del Actor	Respuesta del Negocio
2- El especialista elige la opción actualizar.	2.1- El sistema muestra el formulario para la búsqueda exigiendo ID del metadato.
3- El especialista introduce el ID del metadato.	3.1- El sistema realiza una búsqueda del metadato. 3.2- El sistema muestra el metadato con sus posibles variables a actualizar.
4- El especialista introduce los datos de actualización.	4.1- El sistema actualiza el metadato. 4.2- El sistema muestra el mensaje "El

	metadato fue actualizado”.
Sección “Emitir Reporte”	
2- El especialista elige la opción de emitir reporte.	2.1- El sistema muestra el formulario para la búsqueda exigiendo ID del metadato.
3- El especialista introduce el ID del metadato.	3.1- El sistema realiza una búsqueda del metadato. 3.2- El sistema muestra el Reporte (Estado del metadato).
Sección “Importar archivo”	
2- El especialista elige la opción importar archivo.	2.1- El sistema muestra un formulario exigiendo la dirección donde se encuentra el archivo metadato.
3- El especialista indica la dirección donde se encuentra el archivo metadato.	3.1- El sistema importa el archivo metadato. 3.2- El sistema muestra el mensaje “Su metadato fue importado con éxito”.
Flujos Alternos	
Sesión “Actualizar Metadato”	
Acción del Actor	Respuesta del Negocio
	3.2- Si no encuentra el metadato muestra un mensaje de error y vuelve al paso 3.
Sesión “Emitir Reporte”	
Acción del Actor	Respuesta del Negocio
	3.2- Si no encuentra el metadato muestra un mensaje de error y vuelve al paso 3.
Sesión “Importar Archivo”	
Acción del Actor	Respuesta del Negocio
	3.1- Si el archivo no cumple con el formato metadato se muestra un mensaje de error. 3.1.1- Si el archivo ya está en la BD se muestra un mensaje de error.
<i>Prototipo de Interfaz</i>	

Poscondiciones	Se emite un reporte, se actualiza un metadato o se importa un archivo metadato.
-----------------------	---

Tabla 8 Descripción CUS "Gestionar Metadato".

Caso de Uso:	Actualizar Servicio de Información.	
Actores:	Especialista.	
Resumen:	El caso de uso se inicia cuando el actor Especialista solicita al sistema actualizar la información que brinda el mismo a todos sus usuarios, el sistema le permite realizar estas acciones, terminando así el caso de uso.	
Precondiciones:	La autenticación del usuario debe haber culminado con éxito.	
Referencias	R10, R11, R12.	
Prioridad	Media.	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	1- El especialista entra en la sección actualizar información.	1.1- El sistema muestra las opciones al especialista:
	2- El especialista elige la opción deseada.	a) Actualizar Asistencia Técnica. b) Agregar Preguntas. c) Eliminar Preguntas.
Sección "Actualizar Asistencia Técnica"		
	Acción del Actor	Respuesta del Negocio
	2- El especialista elige la opción de actualizar asistencia técnica.	2.1- El sistema muestra un formulario para que el usuario introduzca la nueva información en forma texto.
	3- El especialista introduce el texto actualizado.	3.1- El sistema actualiza la información.
Sección "Agregar Preguntas"		
	Acción del Actor	Respuesta del Negocio
	2- El especialista elige la opción de agregar preguntas.	2.1- El sistema muestra un formulario exigiendo pregunta y respuesta de la misma.
	3- El especialista introduce la pregunta y su respuesta.	3.1- El sistema agrega la pregunta y su respuesta.

Sección “Eliminar Preguntas”	
Acción del Actor	Respuesta del Negocio
2- El especialista elige la opción eliminar preguntas.	2.1- El sistema muestra un listado con las posibles preguntas a eliminar.
3- El especialista elige la pregunta a eliminar.	3.1- El sistema muestra un mensaje preguntando si desea continuar con la acción eliminar.
4- El especialista elige la opción Si.	4.1- El sistema elimina la pregunta seleccionada.
Flujos Alternos	
Sección “Eliminar Preguntas”	
Acción del Actor	Respuesta del Negocio
4- El especialista elige la opción No.	4.1- El sistema vuelve al paso 2.1.
Prototipo de Interfaz	
Poscondiciones	Queda actualizado el servicio de preguntas o el servicio de asistencia técnica.

Tabla 9 Descripción CUS “Actualizar servicio de información”.

Caso de Uso:	Gestionar Logs del Sistema.
Actores:	Especialista.
Resumen:	El caso de uso se inicia cuando el Especialista se encarga de gestionar los logs del sistema, mostrándolos, salvándolos o eliminándolos, el sistema permite realizar las acciones sobre dichas funcionalidades, terminado así el caso de uso.
Precondiciones:	La autenticación del usuario debe haber culminado con éxito.
Referencias	R5, R6, R7.
Prioridad	Media.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El especialista entra en la sección gestionar logs.	1.1- El sistema muestra las opciones al especialista:
2- El especialista elige la opción deseada.	a) Mostrar Logs.

	<ul style="list-style-type: none"> b) Salvar Logs. c) Eliminar Logs.
Sección "Mostrar Logs"	
Acción del Actor	Respuesta del Negocio
2- El especialista elige la opción mostrar logs.	2.1- El sistema muestra el formulario para mostrar logs por usuario o por fecha.
3- El especialista elige la opción deseada.	3.1- El sistema realiza búsqueda de logs. 3.2- El sistema muestra los logs encontrados.
Sección "Salvar Logs"	
Acción del Actor	Respuesta del Negocio
2- El especialista elige la opción de salvar logs.	2.1- El sistema muestra el formulario para salvar logs por usuario o fecha y el nombre del fichero de la salva.
3- El especialista elige la opción deseada e introduce nombre del fichero.	3.1- El sistema realiza búsqueda de logs. 3.2- El sistema salva el fichero en una carpeta predeterminada.
Sección "Eliminar Logs"	
Acción del Actor	Respuesta del Negocio
2- El especialista elige la opción eliminar logs.	2.1- El sistema muestra el formulario para eliminar logs por usuario o fecha.
3- El especialista elige la opción deseada.	3.1- El sistema realiza búsqueda de logs. 3.2- El sistema elimina el log. 3.3- El sistema muestra el mensaje "El log seleccionado fue eliminado".
Flujos Alternos	
Sección "Mostrar Logs"	
Acción del Actor	Respuesta del Negocio
	3.2- Si no se encuentra los logs muestra un mensaje de error y vuelve al paso 3.
Sección "Salvar Logs"	
Acción del Actor	Respuesta del Negocio
	3.2- Si no se encuentra los logs muestra un

	mensaje de error y vuelve al paso 3.
Sección "Eliminar Logs"	
Acción del Actor	Respuesta del Negocio
	3.2- Si no se encuentra el log muestra un mensaje de error y vuelve al paso 3.
<i>Prototipo de Interfaz</i>	
Poscondiciones	Se muestran, se salvan o se eliminan los ficheros logs del sistema.

Tabla 10 Descripción CUS "Gestionar Logs del Sistema".

3.5. Conclusiones parciales.

- ✚ La descripción detallada de la propuesta de solución a través de la modelación del negocio propuesto; permitió identificar los actores y trabajadores que intervienen, así como los Casos de Uso correspondientes.
- ✚ El planteamiento de los requisitos funcionales (RF) y los no funcionales (RNF) de la aplicación, propició la modelación de la aplicación en términos de Casos de Uso del Sistema.
- ✚ Se obtuvo un enfoque más preciso del análisis y el futuro diseño de la aplicación.

CAPÍTULO 4 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.1. Introducción

En el presente capítulo se realiza el análisis y diseño (construcción) de la propuesta de solución, a través de los flujos de Análisis, Diseño e Implementación, se presenta el diagrama de clases del análisis y del diseño de los diferentes casos de usos definidos en el capítulo anterior, se muestra el modelo lógico y físico de datos. Finalmente se realiza el modelo de despliegue donde se representan los nodos en los que se distribuye la aplicación y el de componentes para una mejor descripción de la solución propuesta.

4.2. Modelo de Análisis

4.2.1. Diagrama de Clases del Análisis

El modelo de análisis ayuda a refinar y estructurar los requisitos, proporciona una estructura centrada en aspectos tales como la flexibilidad ante los cambios y la reutilización. Se representa mediante un sistema de análisis que denota el paquete de más alto nivel del modelo.

Rumbaugh plantea sobre el modelo de análisis que es descrito con el lenguaje del desarrollador, estructurado por clases y paquetes estereotipados; proporciona la estructura a la vista interna, utilizado fundamentalmente para comprender cómo debería darse forma al sistema, es decir, cómo debería ser diseñado e implementado, no debe contener redundancias, inconsistencias, etc., entre requisitos, esboza como llevar a cabo la funcionalidad dentro del sistema, incluida la funcionalidad significativa para la arquitectura; sirve como una primera aproximación al diseño, define realizaciones de casos de uso, y cada una de ellas representa el análisis de un caso de uso del modelo de casos de uso. Para un mejor entendimiento puede consultar el [ANEXO 1](#). Las clases de análisis representan abstracciones de clases y posiblemente de subsistemas del diseño del sistema. Dentro del modelo de análisis, los casos de uso se describen mediante clases de análisis llamadas generalmente "realizaciones de caso de uso-análisis".

4.2.2. Diagramas de Interacción

La secuencia de acciones en un caso de uso comienza cuando un actor invoca el caso de uso mediante el envío de algún tipo de mensaje al sistema. Si se considera el “interior” del sistema, un objeto de interfaz recibirá este mensaje del actor. El objeto de interfaz enviará a su vez un mensaje a algún otro objeto, y de esa forma los objetos implicados interactuarán para llevar a cabo el caso de uso.

Los diagramas de interacción están constituidos por dos tipos: los diagramas de Colaboración y los de Secuencia. Ambos expresan información similar, pero en una forma diferente. Ver [ANEXO 2](#).

4.3. Modelo de Diseño.

4.3.1 Diagrama de clases de Diseño.

En la fase de diseño se modela el sistema de manera que soporte todos los requisitos, tanto funcionales como no funcionales, creándose así una entrada apropiada para las actividades de implementación. En este modelo, los casos de uso son realizados por las clases del diseño y sus objetos, lo cuál se denota por la realización de casos de uso del diseño que describe cómo se realizan estos en particular. (RUMBAUHG 2000a; PRESSMAN 2002)

Sobre el modelo de diseño Rumbaugh plantea que es un modelo de objetos que describe la realización física de los casos de uso, centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar, constituyendo una entrada principal en la actividad de implementación.

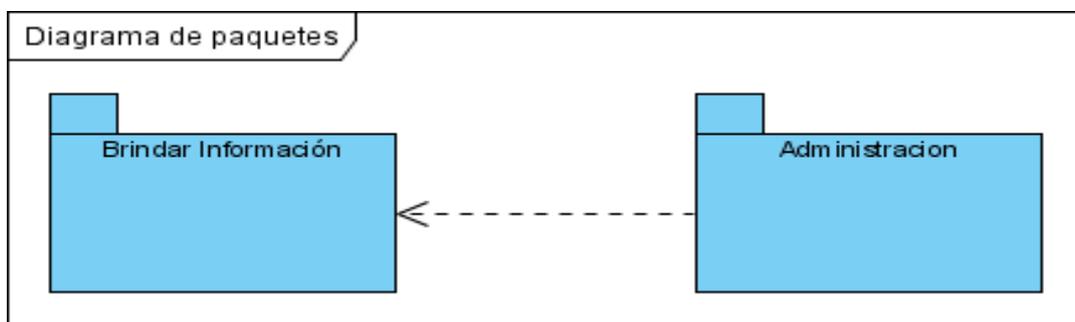


Figura 8 Diagrama de Paquetes.

En cuanto a Symfony, puntualizar que este framework está basado en un patrón clásico de diseño Web conocido como arquitectura Modelo - Vista - Controlador (MVC), que está formado por 3 niveles:(Potencier and Zaninotto 2007)

- ✚ El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- ✚ La vista transforma el modelo en una página Web que permite al usuario interactuar con ella.
- ✚ El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones.

Abstracción de la base de datos

La capa del modelo se puede dividir en la capa de acceso a los datos y en la capa de abstracción de la base de datos. De esta forma, las funciones que acceden a los datos no utilizan sentencias ni consultas que dependen de una base de datos, sino que utilizan otras funciones para realizar las consultas. Así, si se cambia de sistema gestor de bases de datos, solamente es necesario actualizar la capa de abstracción de la base de datos.

Los elementos de la vista

La capa de la vista también puede aprovechar la separación de código. Las páginas Web suelen contener elementos que se muestran de forma idéntica a lo largo de toda la aplicación: cabeceras de la página, el layout¹¹ genérico, el pie de página y la navegación global. Normalmente sólo cambia el interior de la página. Por este motivo, la vista se separa en un layout y en una plantilla. Normalmente, el layout es global en toda la aplicación o al menos en un grupo de páginas. La plantilla sólo se encarga de visualizar las variables definidas en el controlador.

Acciones y controlador frontal

En las aplicaciones Web el controlador suele tener mucho trabajo. Una parte importante de su trabajo es común a todos los controladores de la aplicación. Entre las tareas comunes se encuentran el manejo de las peticiones del usuario, el manejo de la seguridad, cargar la configuración de la aplicación y otras tareas similares. Por este motivo, el controlador normalmente se divide en un

¹¹ EL layout almacena el código HTML que es común a todas las páginas de la aplicación.

controlador frontal, que es único para cada aplicación, y las acciones, que incluyen el código específico del controlador de cada página.

Una de las principales ventajas de utilizar un controlador frontal es que ofrece un punto de entrada único para toda la aplicación. Así, en caso de que sea necesario impedir el acceso a la aplicación, solamente es necesario editar el script correspondiente al controlador frontal.

La implementación del MVC que realiza Symfony

- ✚ La capa del Modelo
 - Abstracción de la base de datos
 - Acceso a los datos
- ✚ La capa de la Vista
 - Vista
 - Plantilla
 - Layout
- ✚ La capa del Controlador
 - Controlador frontal
 - Acción

El controlador frontal y el layout son comunes para todas las acciones de la aplicación. Se pueden tener varios controladores y varios layouts, pero solamente es obligatorio tener uno de cada tipo. El controlador frontal es un componente que sólo tiene código relativo al MVC, por lo que no es necesario crear uno, ya que Symfony lo genera de forma automática.

Las clases de la capa del modelo también se generan automáticamente, en función de la estructura de datos de la aplicación. La librería Propel¹² se encarga de esta generación automática, ya que crea el *esqueleto* o estructura básica de las clases y genera automáticamente el código necesario. Cuando Propel encuentra restricciones de claves foráneas (o externas) o cuando encuentra datos de tipo fecha, crea métodos especiales para acceder y modificar esos datos. La abstracción de la base de datos es completamente invisible al programador, ya que la realiza otro componente específico llamado Creole¹³. Así, si se cambia el sistema gestor de bases de datos en cualquier momento, no se

¹² Librería de Symfony

¹³ Componente de Symfony

debe reescribir ni una línea de código, ya que tan sólo es necesario modificar un parámetro en un archivo de configuración. La lógica de la vista se puede transformar en un archivo de configuración sencillo, sin necesidad de programarla. (Potencier and Zaninotto 2007)

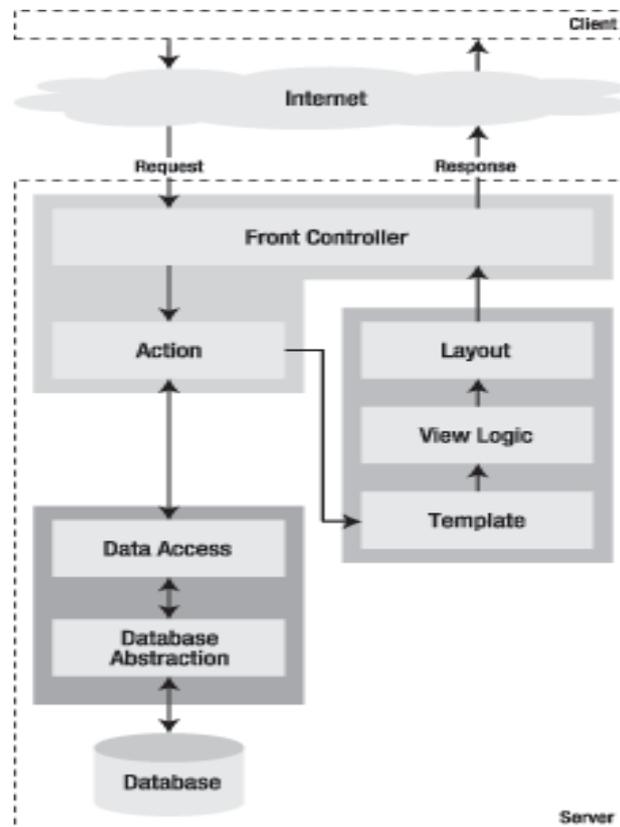


Figura 9 El flujo de trabajo de Symfony.

Se definió para formalizar el diseño del sistema propuesto realizar un diagrama por sesión de cada CU y mejorar la comprensión del mismo. Symfony en la capa de modelo genera 4 clases por cada tabla de la BD para representar la abstracción de los datos, BaseNombreTabla, NombreTabla, BaseNombreTablaPeer y NombreTablaPeer. Ver modelo genérico [ANEXO 3](#).

Se establecieron algunos mecanismos de diseño con el objetivo de simplificar los diagramas y ganar en claridad en la interpretación de estos. Se determinó representar sólo las clases BaseNombreTabla debido a que estas son las más importantes, pues son las que proveen mayor información sobre la distribución de los datos en la BD y las relaciones existentes entre dichos datos; además en el caso de las Action se relacionan solamente con la clase del modelo que tienen relación directamente. Ver [ANEXO 4](#).

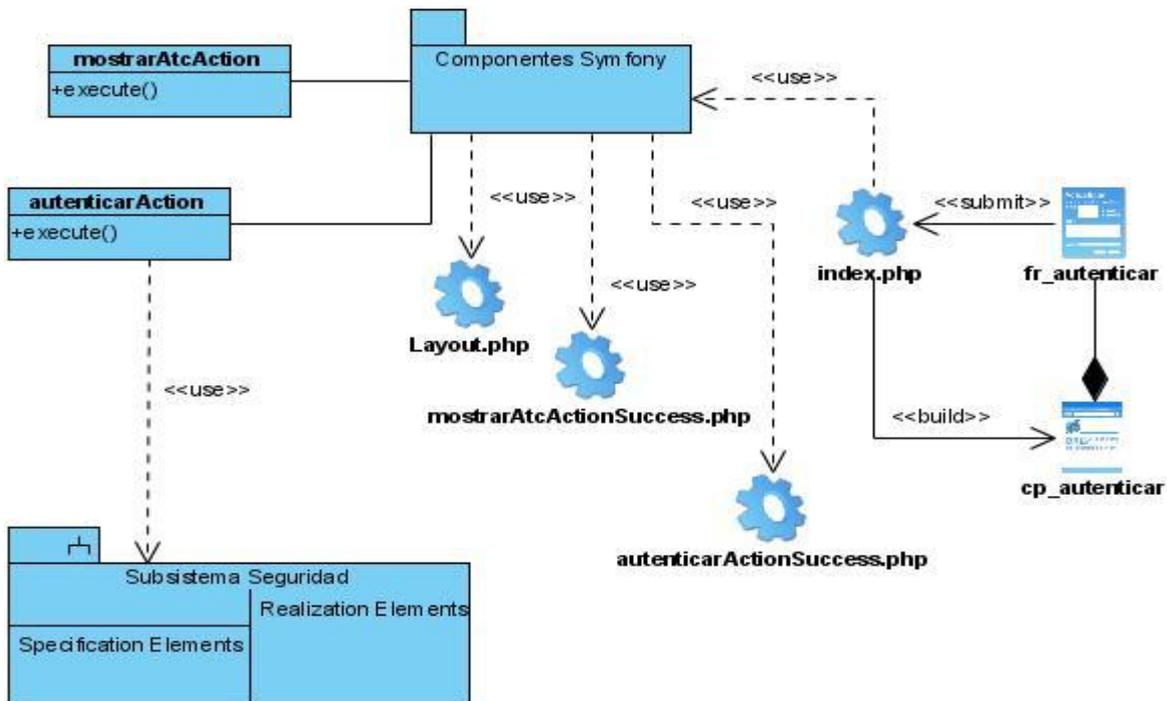


Figura 10 Paquete Brindar Información. Caso de uso " Autenticar ".

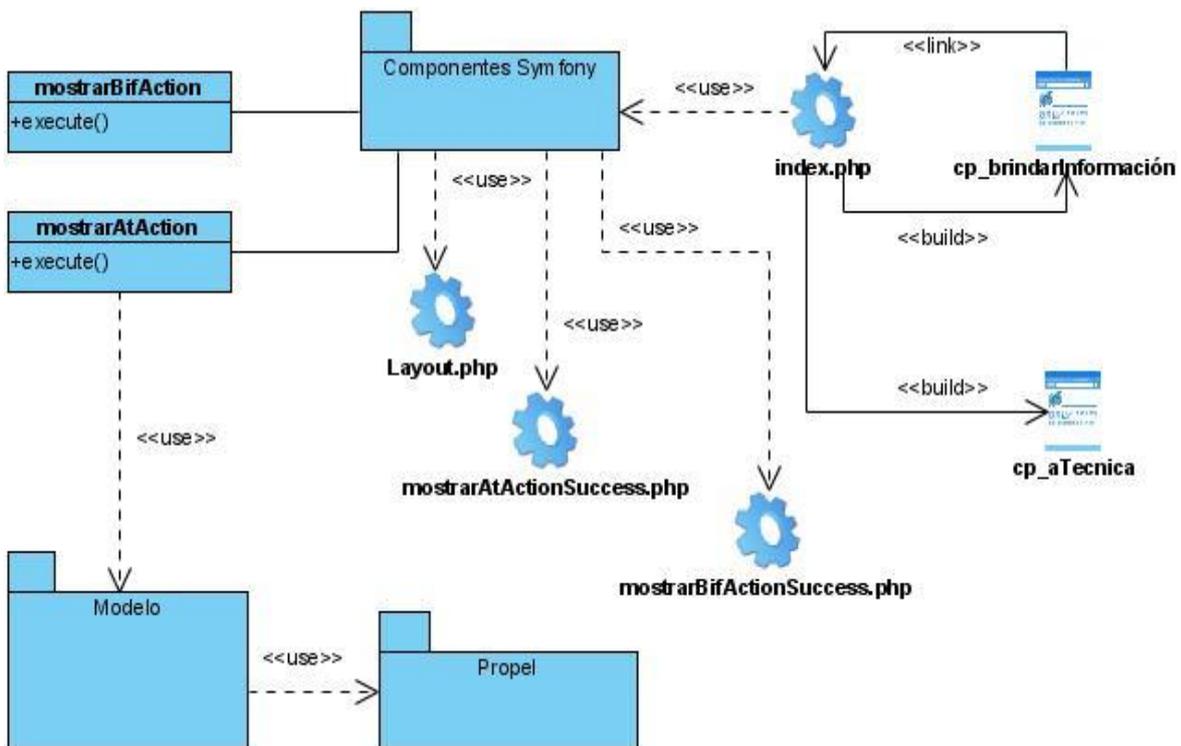


Figura 11 Paquete Brindar Información. Caso de uso " Brindar Información ". Sección " Asistencia Técnica ".

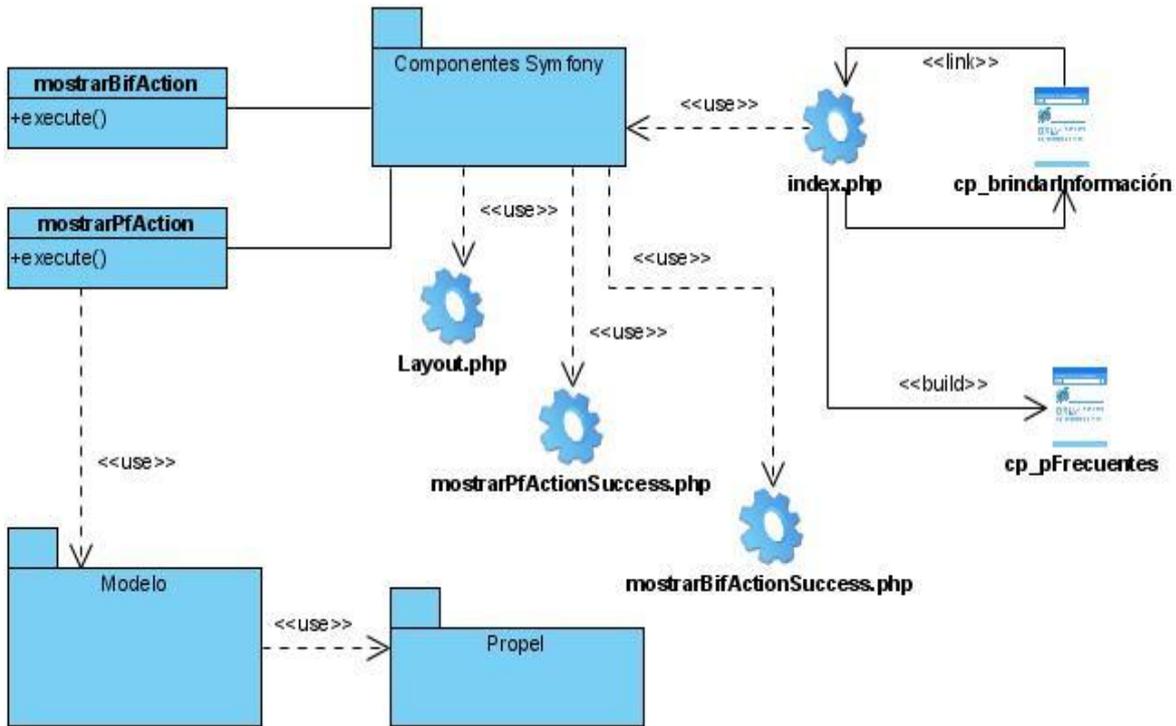


Figura 12 Paquete Brindar Información. Caso de uso " Brindar Información ". Sección " Servicio de Preguntas ".

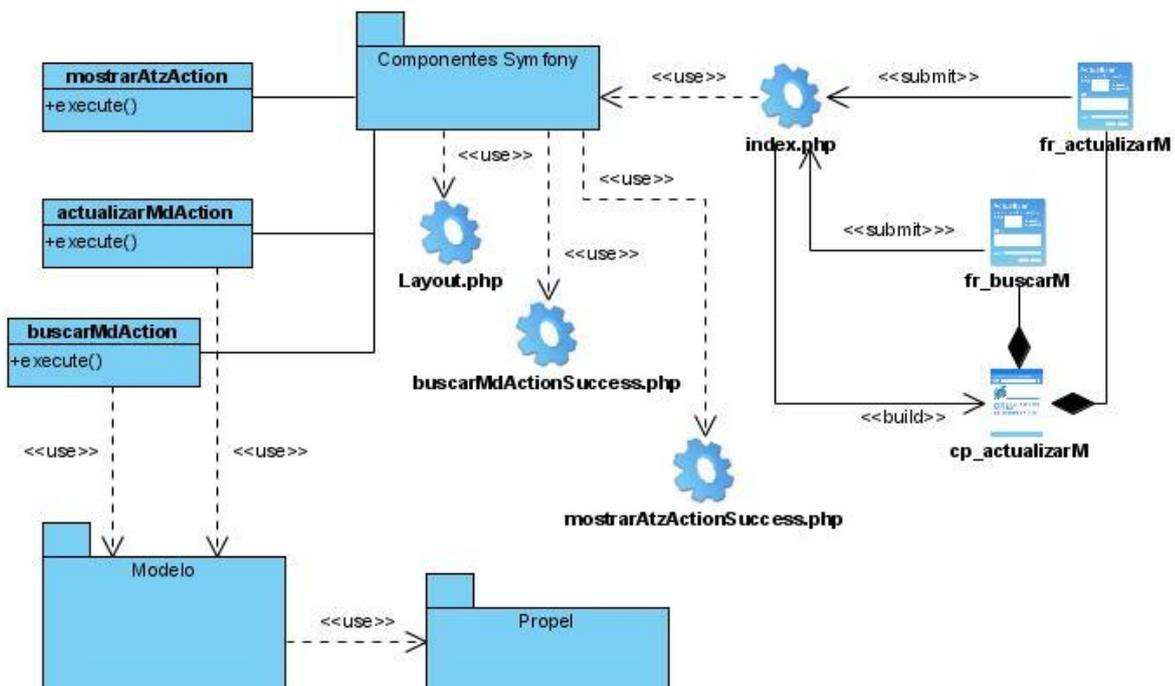


Figura 13 Paquete Administración. Caso de uso " Gestionar metadato ". Sección " Actualizar Metadato ".

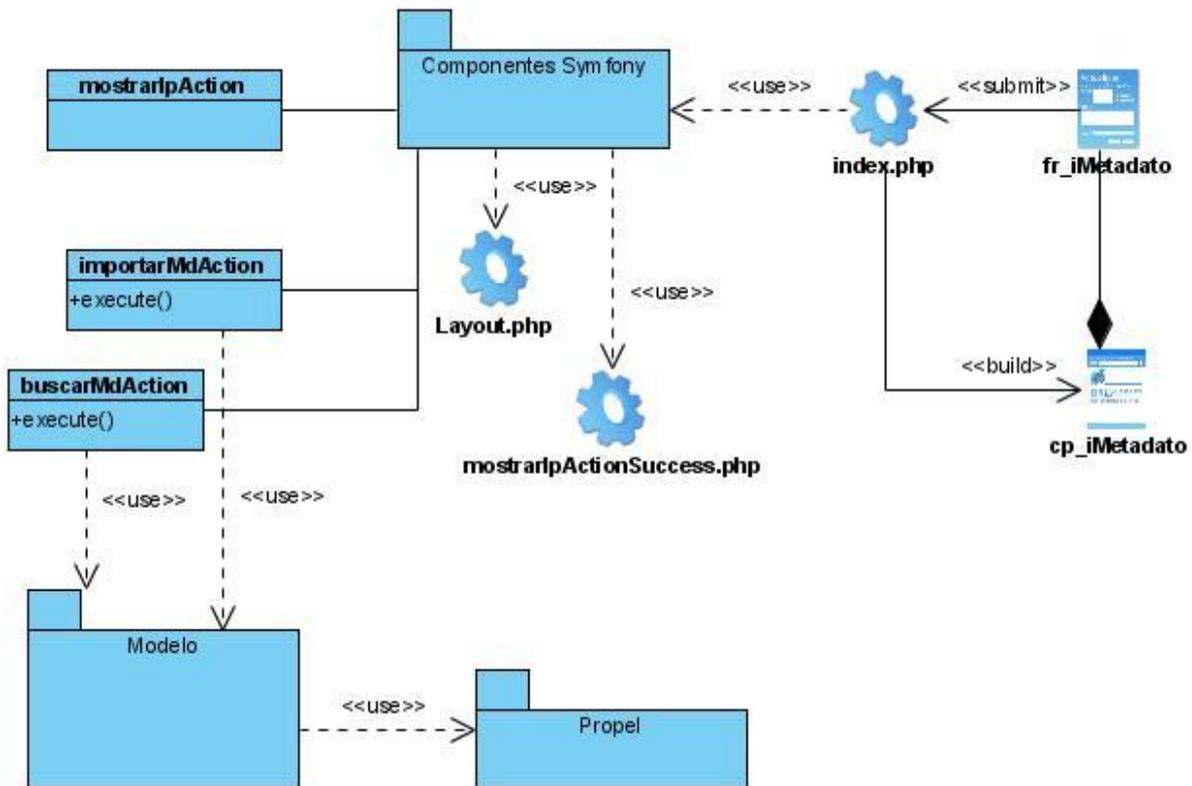


Figura 14 Paquete Administración. Caso de uso " Gestionar metadato ". Sección " Importar Archivo ".

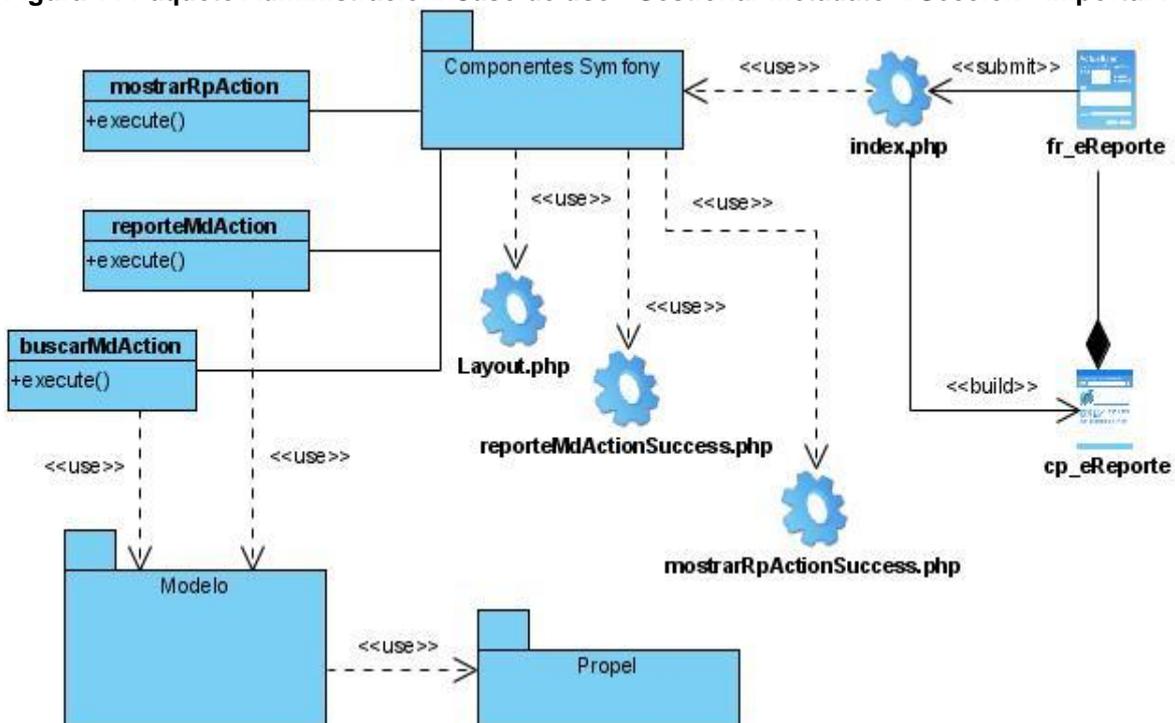


Figura 15 Paquete Administración. Caso de uso " Gestionar metadato ". Sección " Emitir Reporte ".

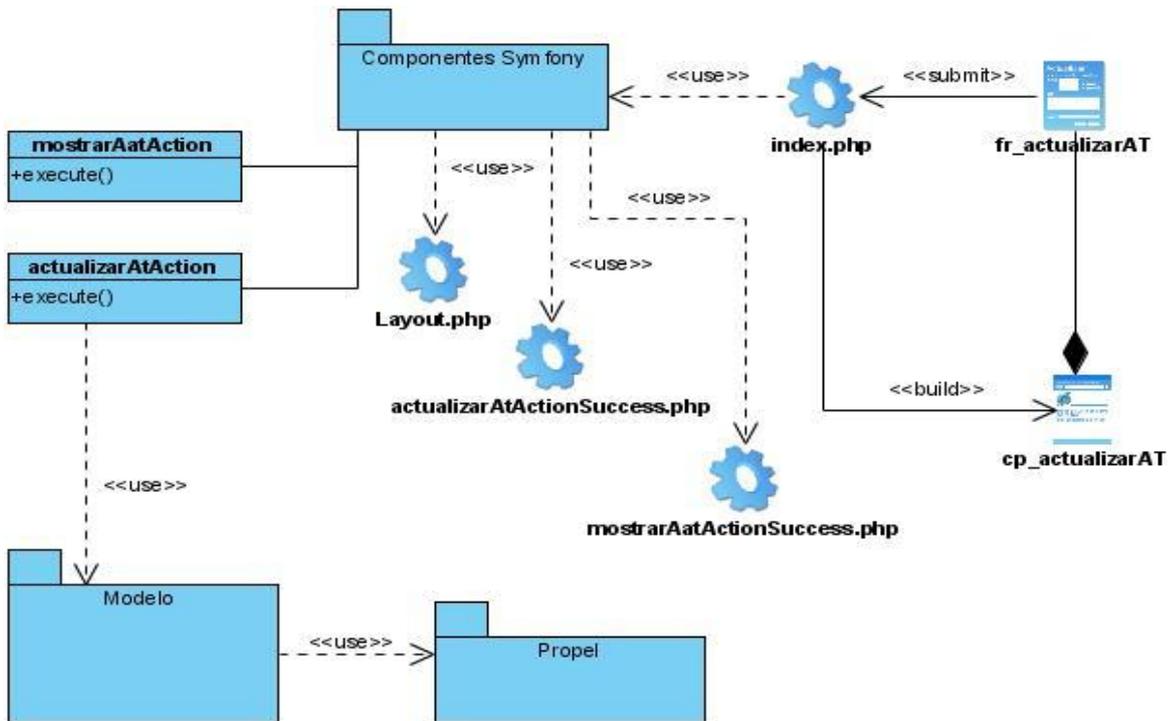


Figura 16 Paquete Administración. Caso de uso " Actualizar servicio de información ". Sección "Actualizar Asistencia Técnica."

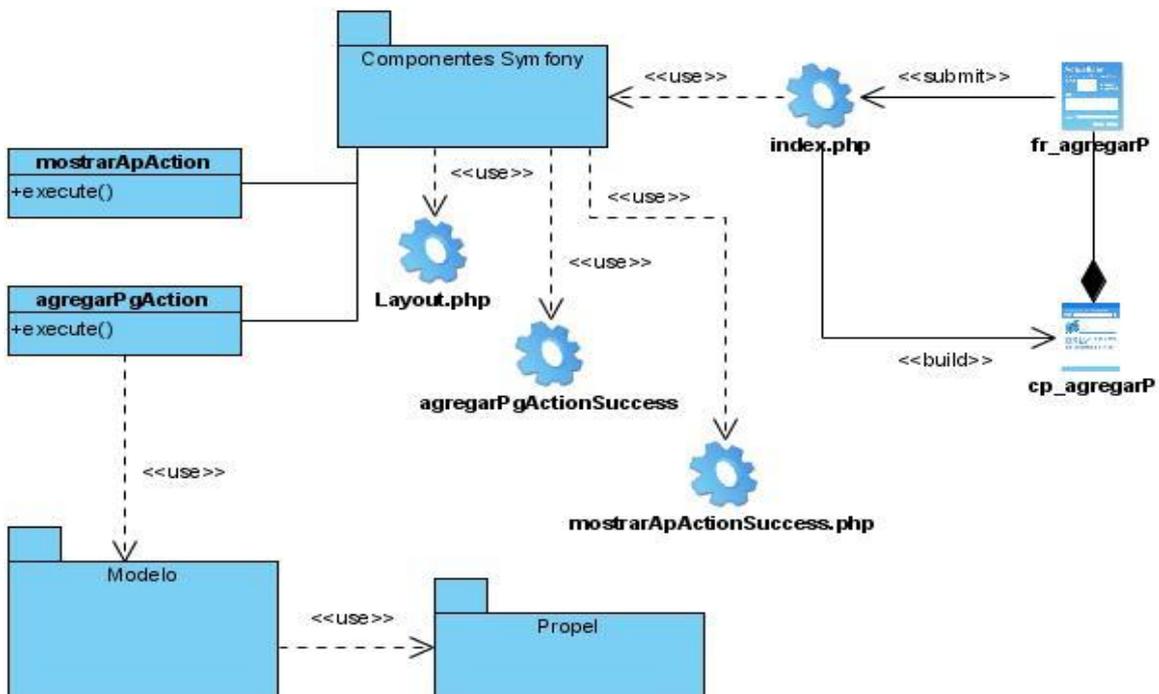


Figura 17 Paquete Administración. Caso de uso " Actualizar servicio de información ". Sección "Agregar Preguntas".

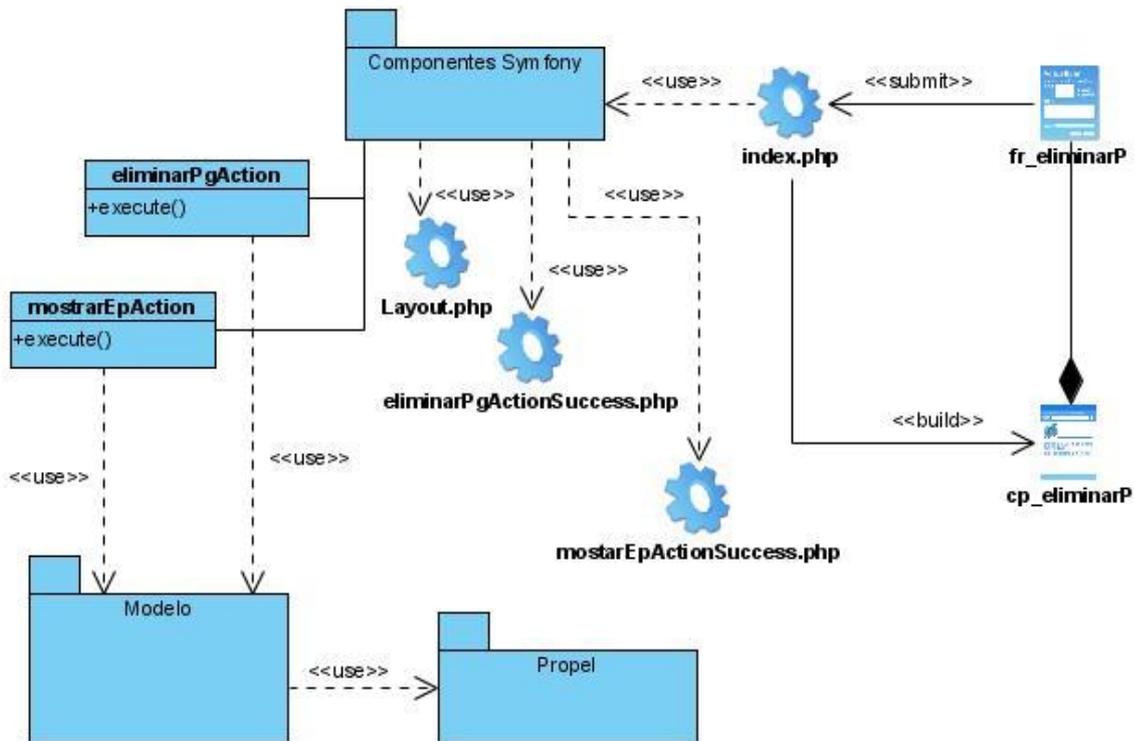


Figura 18 Paquete Administración. Caso de uso " Actualizar servicio de información ". Sección "Eliminar Preguntas".

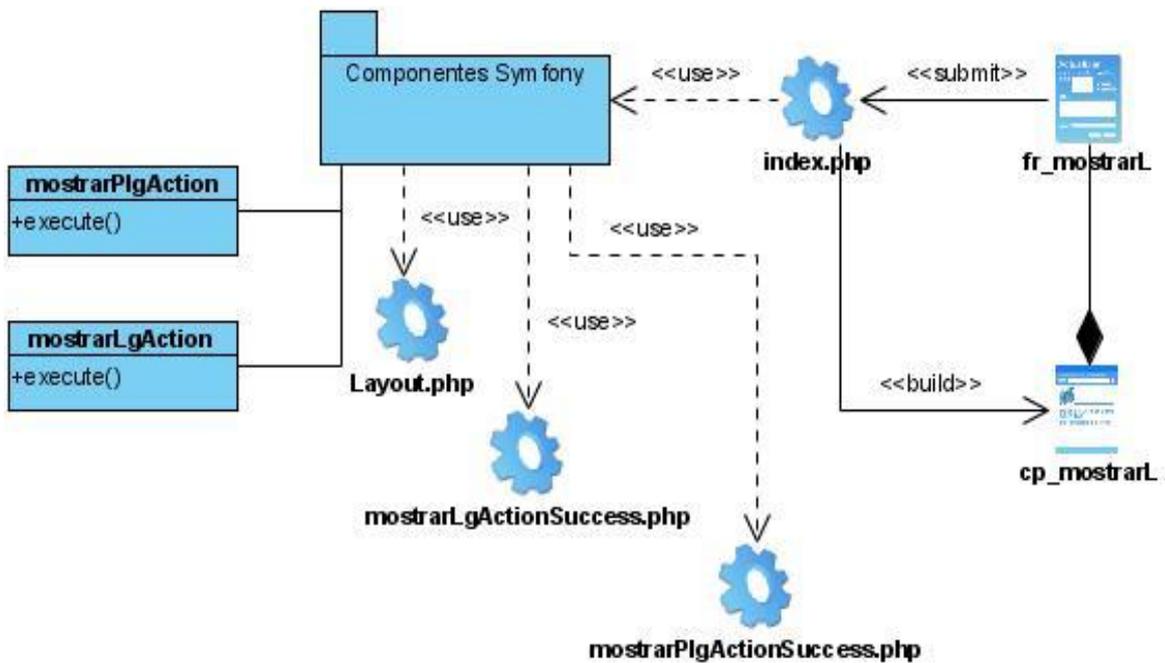


Figura 19 Paquete Administración. Caso de uso " Gestionar Logs ". Sección "Mostrar Logs".

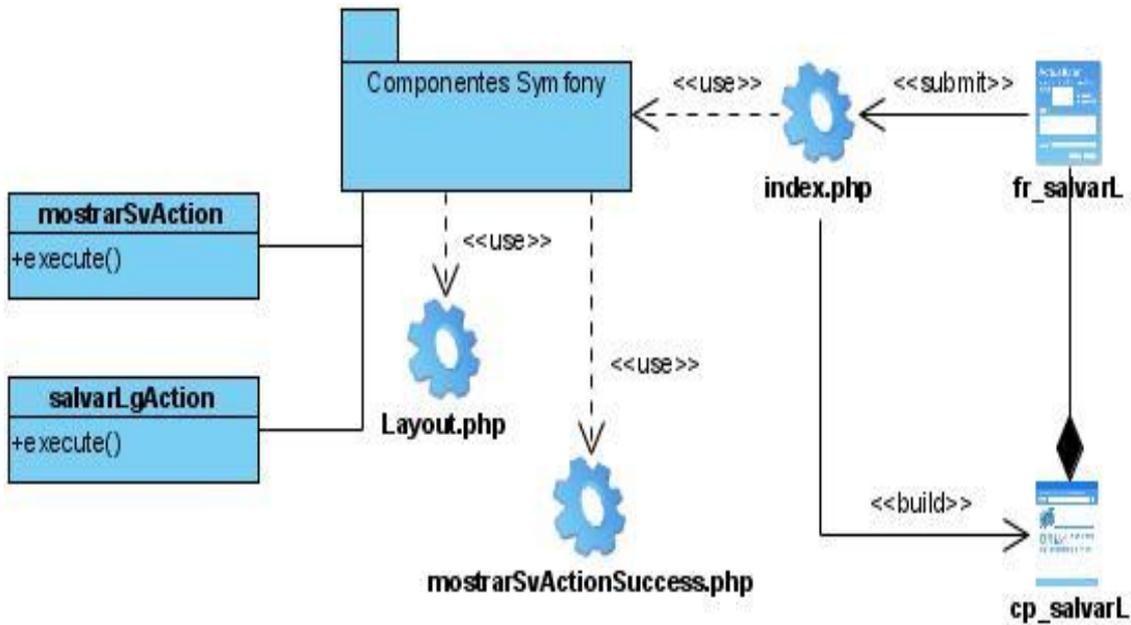


Figura 20 Paquete Administración. Caso de uso "Gestionar Logs". Sección "Salvar Logs".

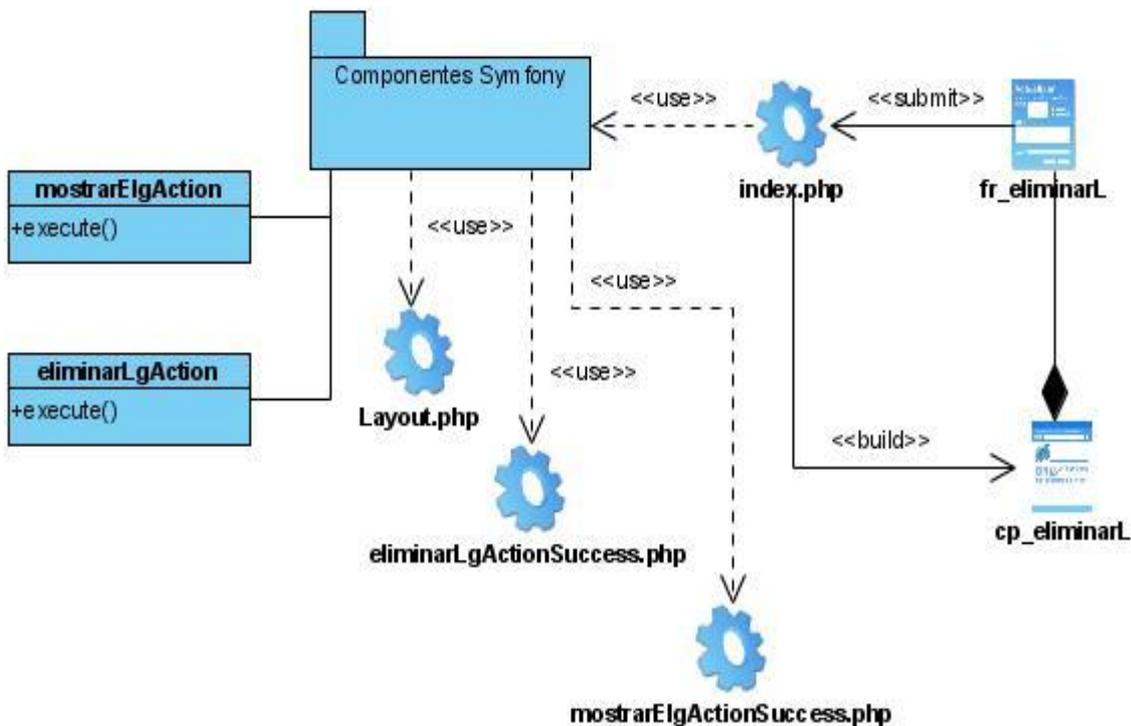


Figura 21 Paquete Administración. Caso de uso "Gestionar Logs". Sección "Eliminar Logs".

4.4. Principios de Diseño

4.4.1. Diseño de la interfaz de la aplicación.

El diseño Web no se limita a la apariencia estética, la combinación de colores, o a un logo más o menos acertado. De él depende que la información sea útil, que los servicios se puedan usar, es decir, el diseño convierte a una aplicación Web en algo atractivo para el usuario, por su estética y su utilidad.

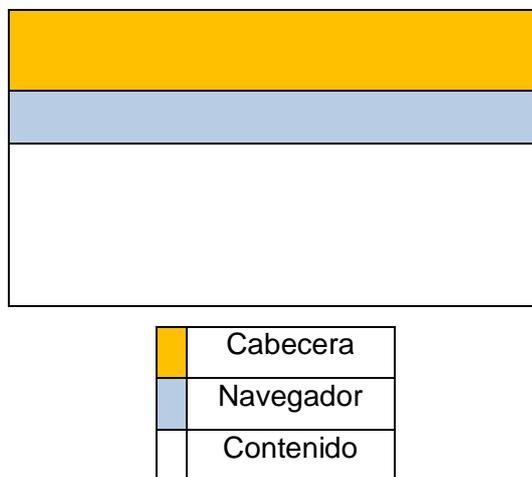
Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos. Para ello, este sistema utiliza ciertos principios generales que garantizan la usabilidad en los diseños para aplicaciones Web.

- ✚ Requerir un mínimo proceso de aprendizaje y permitir su utilización desde el primer momento, por cualquier persona que tenga un mínimo dominio de la computación.
- ✚ Requerir de los usuarios un mínimo esfuerzo para alcanzar sus objetivos.
- ✚ Garantizar la legibilidad, el color de los textos debe contrastar con el del fondo, y el tamaño de fuente debe ser suficientemente grande.
- ✚ Evitar elementos invisibles de navegación que han de ser inferidos por los usuarios, menús desplegados, indicaciones ocultas, etc.
- ✚ Evitar las caídas inesperadas de la aplicación y los enlaces rotos.
- ✚ Mostrar al usuario solamente aquellas opciones a las que, dado su rol en el negocio, tiene derecho a acceder.
- ✚ Mostrar al usuario, siempre que vaya a realizar una acción relevante sobre el sistema, un mensaje de confirmación que le permita asegurarse que es correcta la opción seleccionada.
- ✚ Mostrar la mayor cantidad de información acerca de las opciones brindadas en un momento dado, de modo que el usuario siempre sepa cuáles son las operaciones a las que puede acceder y en qué consiste exactamente cada una.

4.4.2. Estándares de la interfaz de la aplicación.

La interfaz gráfica es el medio mediante el cual interactúa el usuario con el sistema, pues esta debe ser lo más amigable posible y lograr que se sienta identificado con la misma.

Con vistas a lograr un diseño consistente de la interfaz de la aplicación, en todas las páginas se ha seguido el esquema Cabecera-Navegador-Contenido, uno de los más usados actualmente. La cabecera contiene el logotipo de la aplicación en la esquina superior izquierda y una barra de navegación con enlaces de texto para cada uno de los módulos de la aplicación. A continuación se muestra el esquema de una página.



Se utilizan para el diseño las tablas y plantillas, dado que son 100% compatibles con todos los navegadores, hasta en sus versiones más antiguas, a diferencia de los marcos. Se utiliza también una hoja de estilos para guardar la configuración del diseño de los menús de todas las páginas. Esta hoja de estilos establece el tipo y tamaño de fuente de los distintos elementos de cada página. Se utiliza en general las familias de fuentes Arial y Sans-Serif, de tamaños entre 9 y 14 píxeles respectivamente, según la importancia de la información mostrada.

4.6. Modelo de Despliegue.

El modelo de despliegue describe la distribución física del sistema, muestra como están distribuidos los componentes de software entre los distintos nodos de cómputo. Permite comprender la correspondencia entre la arquitectura software y la arquitectura hardware. Además este modelo muestra la topología del hardware sobre el que se ejecuta el sistema.

Un nodo es un objeto físico en tiempo de ejecución, es decir una máquina que se compone habitualmente de, por lo menos, memoria y capacidad de procesamiento, a su vez puede estar formada por otros componentes.



Figura 24 Modelo de Despliegue.

4.7. Modelo de Implementación.

El modelo de implementación describe cómo los elementos del modelo de diseño, se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc. Las relaciones de dependencia se utilizan en los diagramas de componentes para indicar que un componente utiliza los servicios ofrecidos por otro componente.

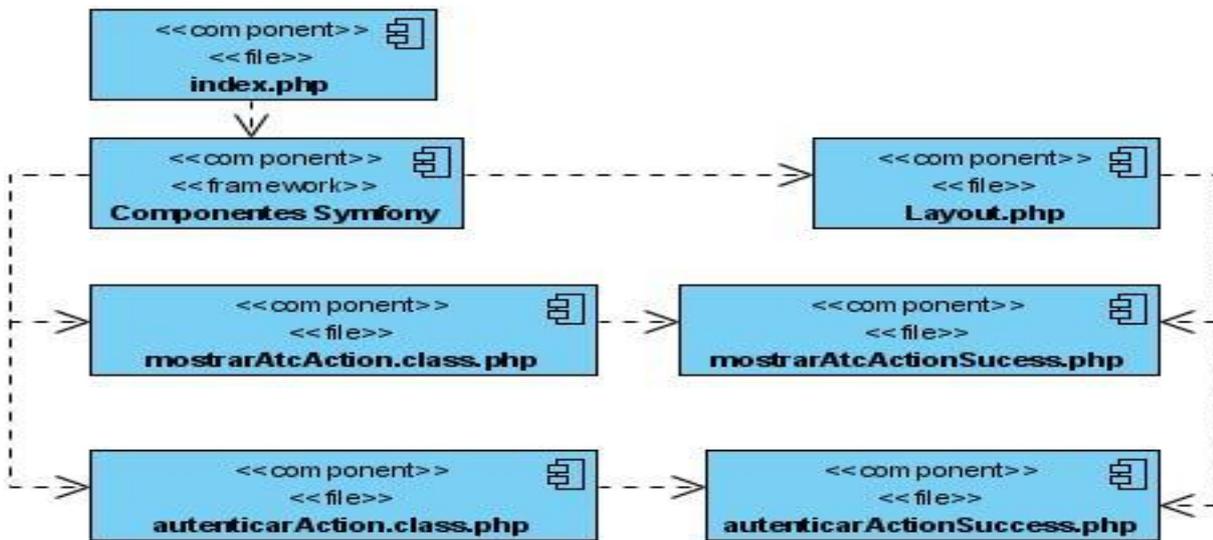


Figura 25 Modelo de Componentes. CU "Autenticar".

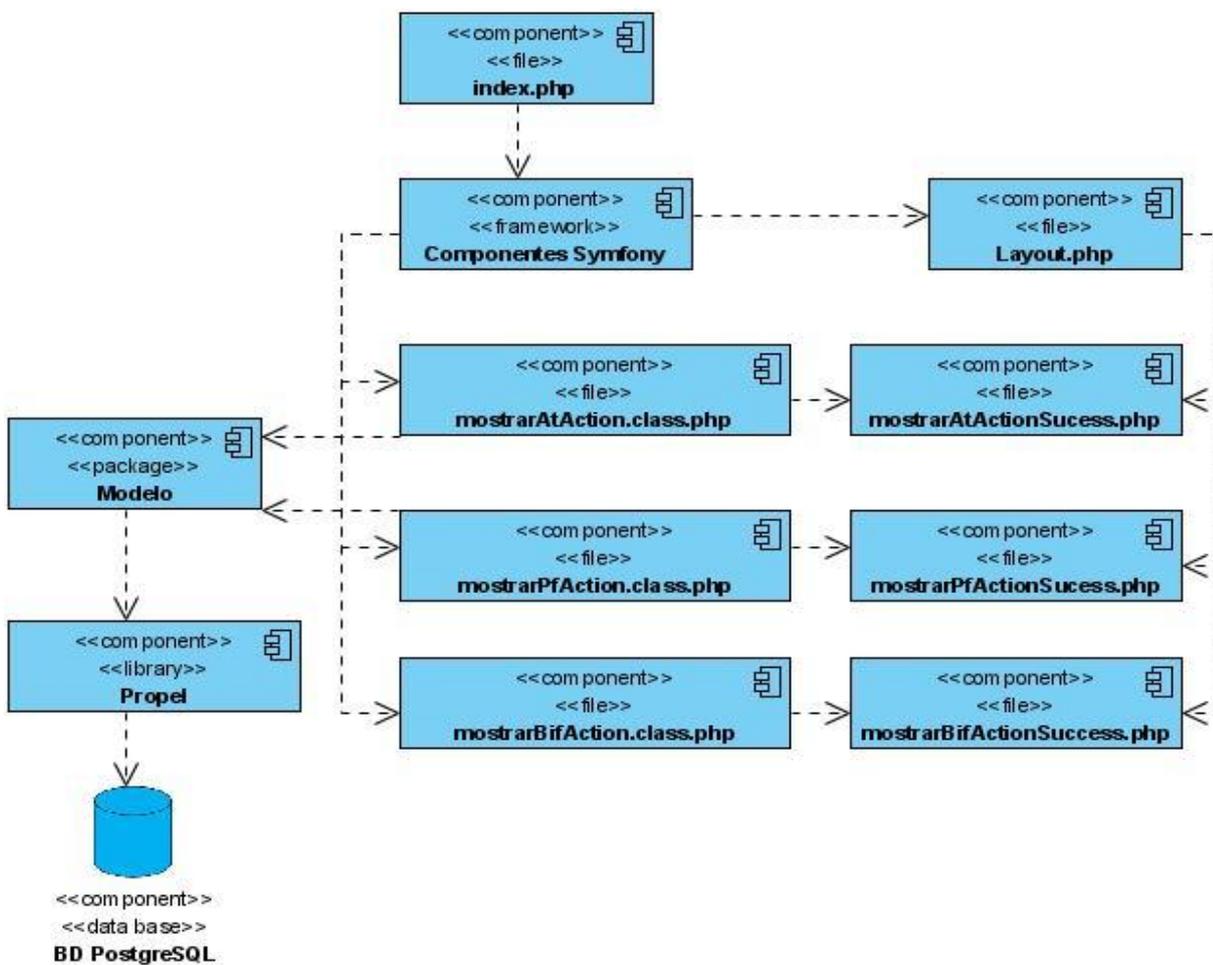


Figura 26 Modelo de Componentes. CU "Brindar Información".

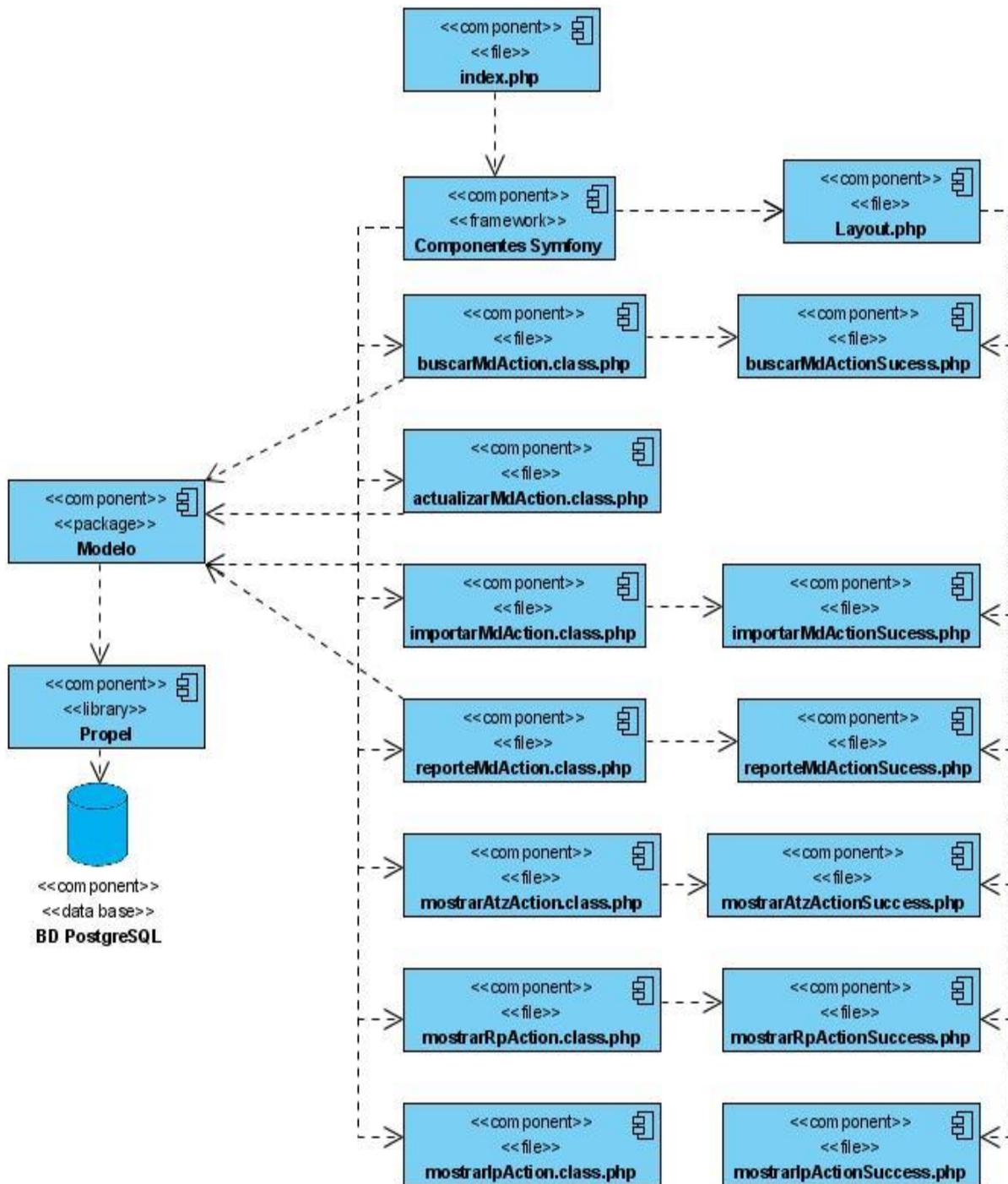


Figura 27 Modelo de Componentes. CU "Gestionar Metadato".

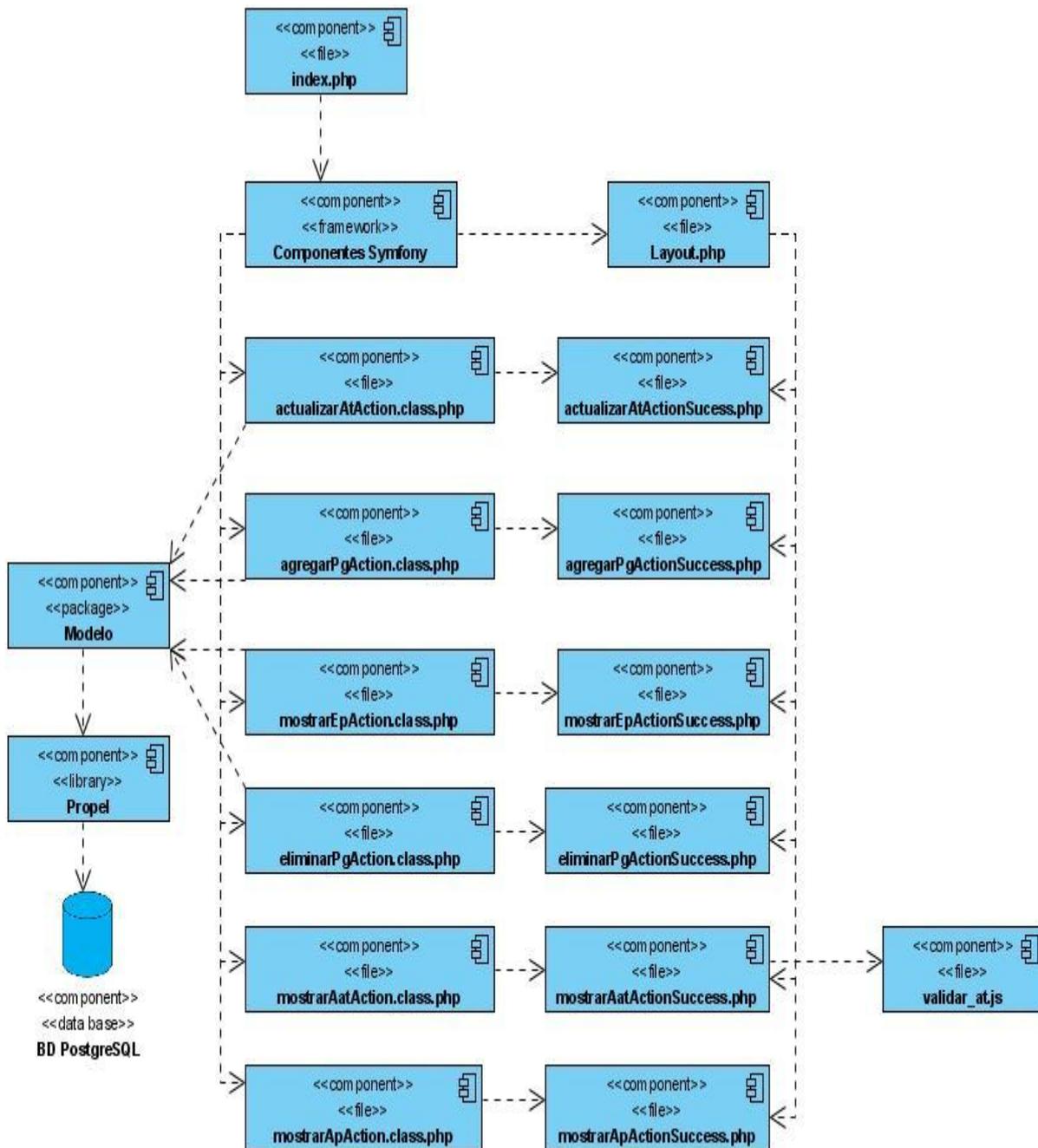


Figura 28 Modelo de Componentes. CU "Actualizar servicio de Información".

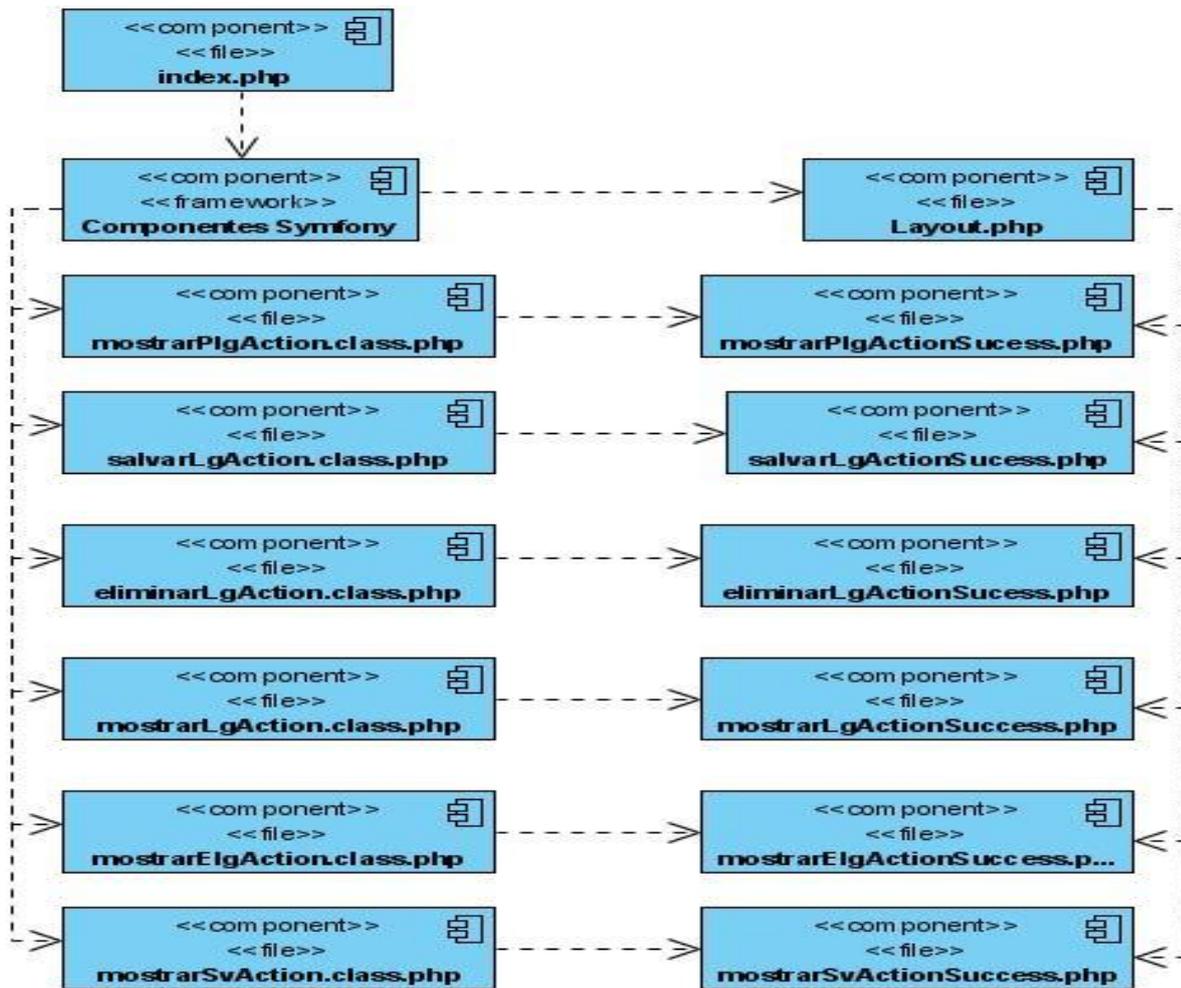


Figura 29 Modelo de Componentes. CU "Gestionar Logs".

4.8. Conclusiones parciales.

- ✚ Se obtuvo la propuesta de diseño de una aplicación Web de fácil uso, entendible y desarrollada con nuevas herramientas para eliminar el trabajo manual en la ONRM.
- ✚ El uso de mecanismos de diseño simplificó la complejidad de los diagramas, facilitando la comprensión de los mismos.
- ✚ Con la realización de la implementación de la aplicación propuesta se conseguirá:
 - Proveer los servicios que necesita brindar la ONRM
 - Facilitar el trabajo a los especialistas.

CONCLUSIONES

Una vez elaborada la investigación se evidencia que con el diseño del Subsistema para la Administración de Metadatos geológicos en la ONRM, se arriban a las siguientes conclusiones:

- ✚ El objetivo elaborado en el diseño de la investigación fue cumplido, comprobándose la idea a defender como respuesta del problema a resolver que originó esta investigación, tributando directamente a la captación de ficheros de metadatos geológicos.
- ✚ El empleo de los métodos teóricos y empíricos facilitó conocer el estado del objeto de estudio.
- ✚ Las tareas investigativas facilitaron una adecuada organización de la investigación.
- ✚ Se garantizarán los niveles de seguridad en la información, mayor agilidad en los resultados, por lo que reducirá el tiempo de respuesta y minimización de los errores en la información geológica.
- ✚ Se logrará la integridad, disponibilidad, confidencialidad y autenticidad de los datos, realizando un control de las acciones llevadas a cabo por los usuarios en el sistema y estableciendo niveles de acceso dada las responsabilidades de los usuarios según sus roles.
- ✚ Con el uso de las herramientas de software libre y multiplataforma se provee a la ONRM de un mínimo de gastos en lo referido al pago de licencias de software.
- ✚ Se obtuvo la propuesta de diseño de una aplicación Web de fácil uso, entendible y desarrollada con nuevas herramientas para eliminar el trabajo manual en la ONRM.

RECOMENDACIONES

Debido a los resultados de todo el proceso de investigación realizado y basado en la experiencia acumulada se proponen las siguientes recomendaciones:

- ✚ Realizar un estudio detallado de la norma ISO 19139 para crear una versión de la representación de la información geológica en ficheros XML basada en dicha norma y ajustable a las necesidades particulares de nuestro país.
- ✚ Incluir en próximas iteraciones, el uso de la técnica de desarrollo Ajax.
- ✚ Realizar la implementación de la propuesta de diseño de la aplicación Web planteada en la investigación.

REFERENCIAS BIBLIOGRÁFICAS

1. (2005) Tema 4 Ficheros. Disponible en: <http://campusvirtual.uma.es/emetprog/METODOLOGIA/matdocMP/teoria/doc/TEMA4-Curso0405.pdf>
2. GARRET, J. J. (2006) Ajax: Un nuevo acercamiento a las aplicaciones web. Disponible en: http://gpi.uci.cu/index.php?option=com_remository&Itemid=31&func=select&id=339.
3. (2006) Visual Paradigm: Disponible en: http://wiki.prod.uci.cu/index.php/Visual_Paradigm
4. Ahumada, J. A. G. I. (2007). "V. SENSORES REMOTOS Y MANEJO DE INFORMACIÓN GEOGRÁFICA." Disponible en: <http://www.ine.gob.mx/ueajei/publicaciones/libros/429/sensores.pdf>
5. Alfaro, F. M. "Tecnología Cliente Servidor."
6. Álvarez, M. A. (2007) DotProject.
7. BERDASCO BLANCO, A. (2007) Sistema Informático. Disponible en: <http://www.ciencia-ficcion.com/glosario/>
8. C. G. Hernandez (2006). Módulo Alojamiento del Sistema Automatizado para la Gestión de Información de la Misión Milagro, Universidad de las Ciencias Informáticas. Ciudad de la Habana.
9. C.Mateu (2004). Desarrollo de Aplicaciones Web. SL.
10. Cabrera, A. O. and Y. P. Velázquez (2007). SISTEMA INFORMATIVO DE PARTES DE PRODUCCIÓN Y DE VENTAS DE PRODUCTOS DEL GRUPO DE LA ELECTRÓNICA DEL MINISTERIO DE INFORMÁTICA Y LAS COMUNICACIONES. Ciudad de La Habana, Universidad de las Ciencias Informáticas.
11. Cantero, J. (2006). Un Vistazo a PHP5. castellano., P. e. (2007) Visual Paradigm for UML. Disponible en: <http://www.programacion.net/noticia/1363/>
12. Cornejo, J. E. G. (2001). "Arquitectura en Capas. Un camino hacia los procesos distribuidos."
13. Corporation, M. (2006). E. base de Datos.
14. Cuba, G. O. d. I. R. d. (1995). RESOLUCION No. 374/99. Disponible en: <http://www.medioambiente.cu/legislacion/resoluciones/R-374-99MINBAS.htm>
15. Diseño, E. d. (2006). Arquitectura del Proyecto de Informatización del Conocimiento Geológico v.1.0.
16. E, H. O. (2007). El lenguaje Unificado de Modelado. Disponible en: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>
17. Garret, J. J. (18 febrero 2005). Ajax: A New Approach to Web Applications.
18. Graells, D. P. M. (2000). Las TIC y sus aportaciones a la sociedad.
19. Instituto Nacional de Estadística, G. e. I. d. M. (2003) Metadatos Geográficos. Disponible en: <http://antares.inegi.gob.mx/metadatos/index.html>
20. Interbusca (2008). "Definición de ficheros." Disponible en: <http://antivirus.interbusca.com/glosario/fichero.html>
21. Kicillof, C. R. a. N. (Marzo 2004). Estilos y Patrones en la Estrategia de Arquitectura de Microsoft.

22. Larman, C. (1999). UML y Patrones. Introducción al análisis y diseño orientado a objetos. Prentice Hall. Disponible en: <http://bibliodoc.uci.cu/pdf/reg00061.pdf>
23. Latina, P. (2007) Desarrollan en Cuba Proyecto de Informatización de Geología. Disponible en: <http://www.bibliociencias.cu/mensaje.php?group=bvcitma.noticias&msg=72>
24. Maganto, A. S. and D. Ballari (2008). Normas sobre Metadatos.
25. Mandrake (2004). Revisión Rápida de PHP5 integrado con Zend.
26. MINBAS (2007). "MINISTERIO DE LA INDUSTRIA BASICA." Disponible en: http://www.cubagob.cu/des_eco/minbas.htm
27. Minera, C. d. I. G. (2007). Servicio Geológico Minero Argentino (SEGEMAR). Disponible en: <http://www.segemar.gov.ar/db/>
28. Minerales, C. d. P. d. R. (2007). Servicio Geológico de Brasil (CPRM). Disponible en: <http://www.cprm.gov.br/>
29. Minería, C. G. d. (2007). Servicio Geológico Mexicano. Disponible en: <http://www.coremisgm.gob.mx/inicio.html>
30. MOLPECERES, A. (2003) Procesos de desarrollo: RUP, XP y FDD. Disponible en: <http://www.javahispano.org/articles.article.action?id=76>
31. Oriente, G. d. (Oct/2007). Procedimientos para la Documentación del Metadato. Santiago de Cuba.
32. Potencier, F. and F. Zaninotto (2007). Symfony, la guía definitiva. Disponible en: <http://www.librosweb.es>
33. PRESSMAN, R. S. (2002). Ingeniería del software. Un enfoque práctico. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02689.pdf>
34. Rafael Oliva Santos, E. Q. O. (2006) LOS METADATOS GEOGRÁFICOS: ACTUALIDAD Y ESTÁNDARES
35. RUMBAUHG, J. I. J. (2000a). El Lenguaje Unificado de Modelado. Manual de referencia. Madrid, Pearson Educación.
36. RUMBAUHG, J. I. J. (2000b). El Proceso Unificado de Desarrollo de Software Madrid, Pearson Educación. . Disponible en: <http://bibliodoc.uci.cu/pdf/reg00060.pdf>
37. Survey, U. S. G. (2007). Science for a Changing World (USGS) Disponible en: <http://www.usgs.gov/>
38. W3C (2006) Guía breve de CSS.
Disponible en: <http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>
39. Paradigm, V (2007) Visual Paradigm for UML Disponible en: <http://www.visualparadigm.com/product/vpuml>

ANEXOS

Anexo 1. Diagrama de clases del Análisis.

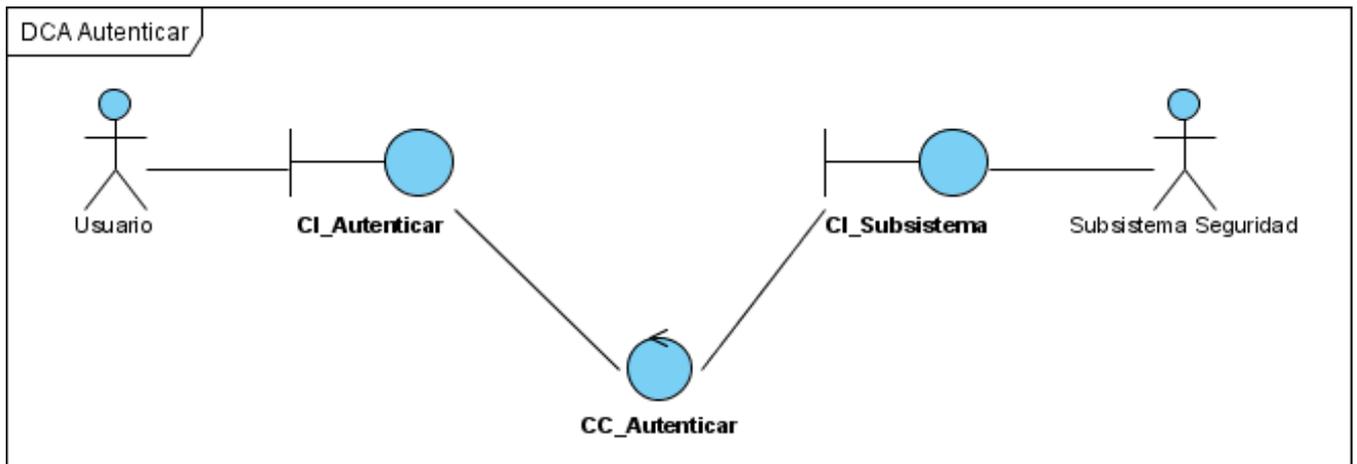


Figura 30 DCA. Caso de uso "Autenticar".

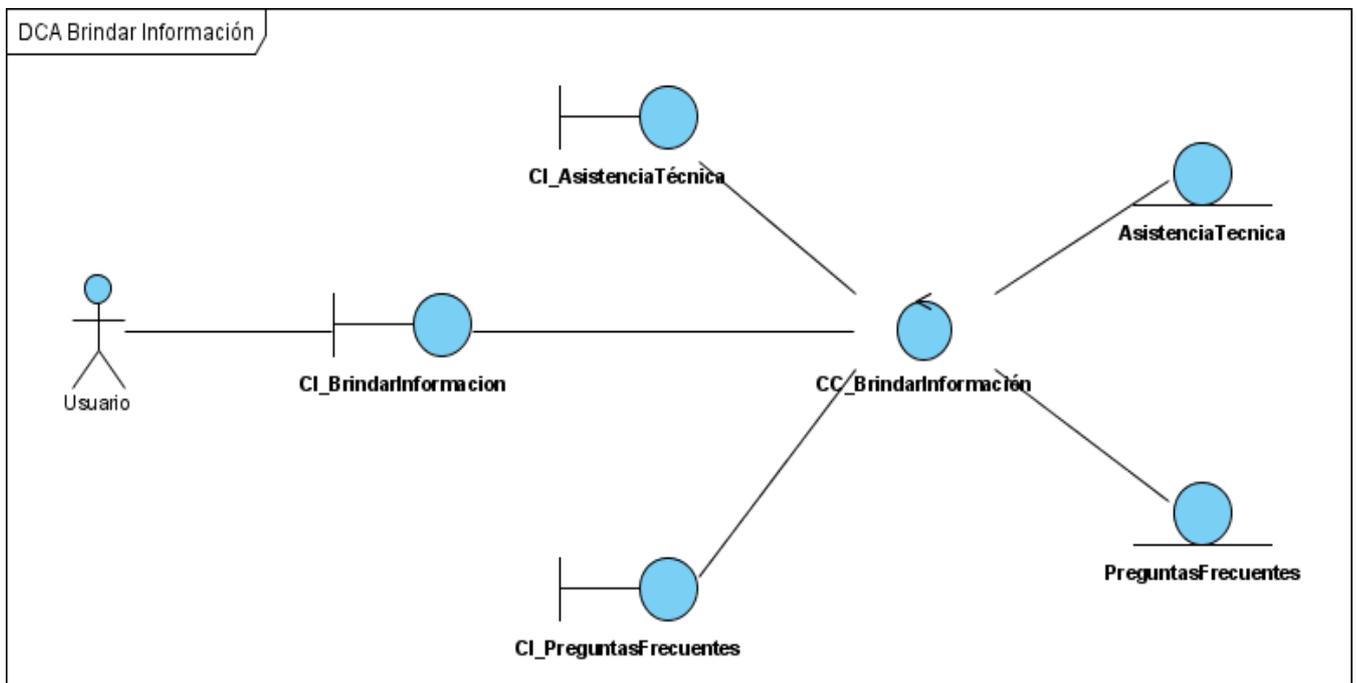


Figura 31 DCA. Caso de uso "Brindar Información".

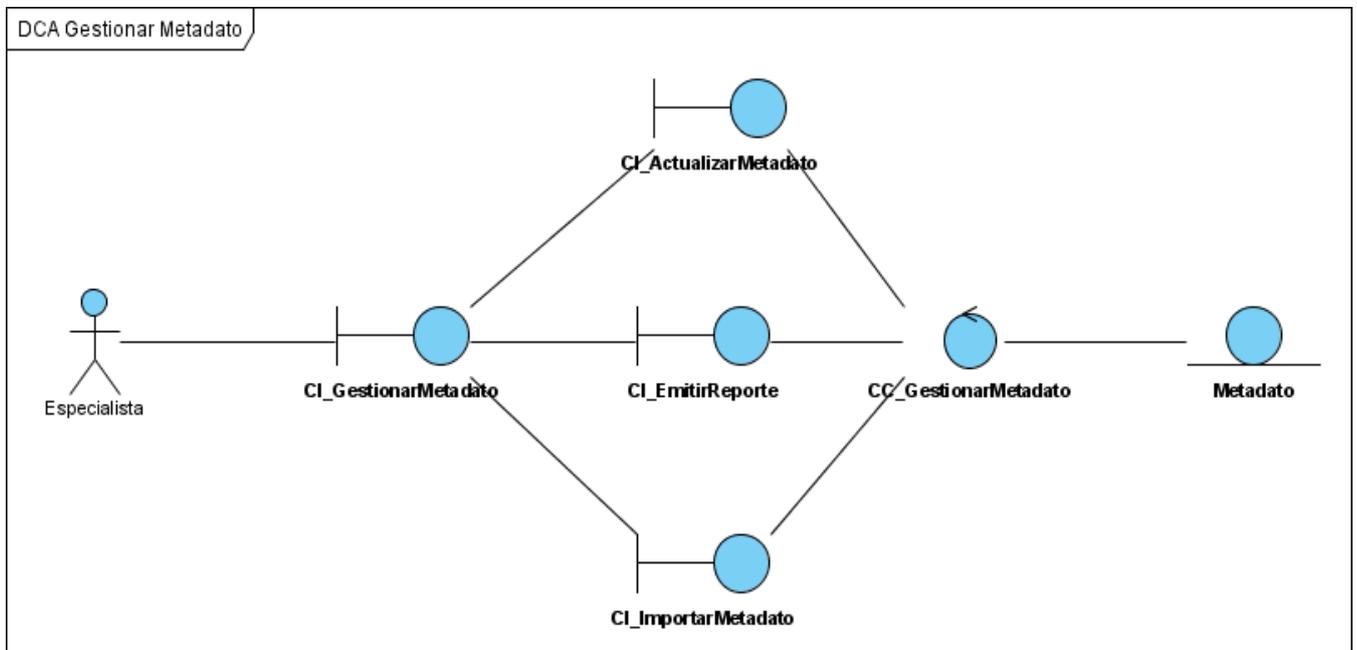


Figura 32 DCA. Caso de uso "Gestionar Metadato".

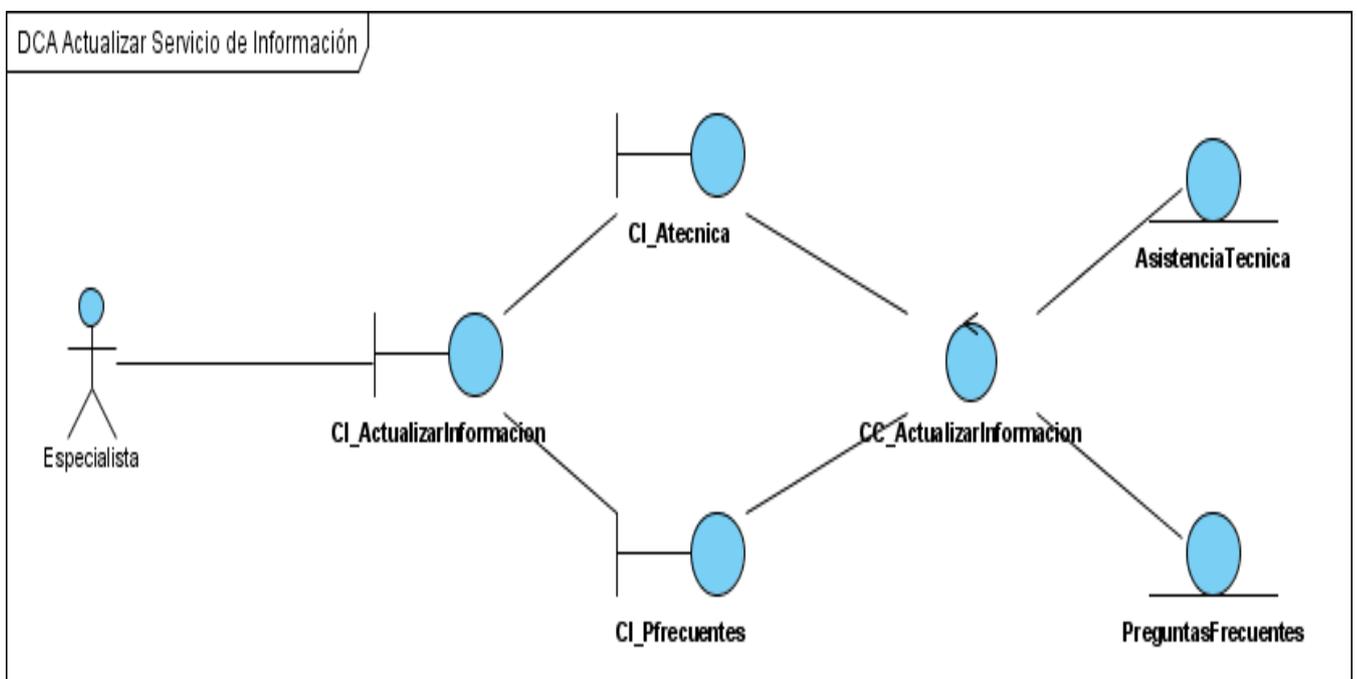


Figura 33 DCA. Caso de uso "Actualizar servicio de Información".

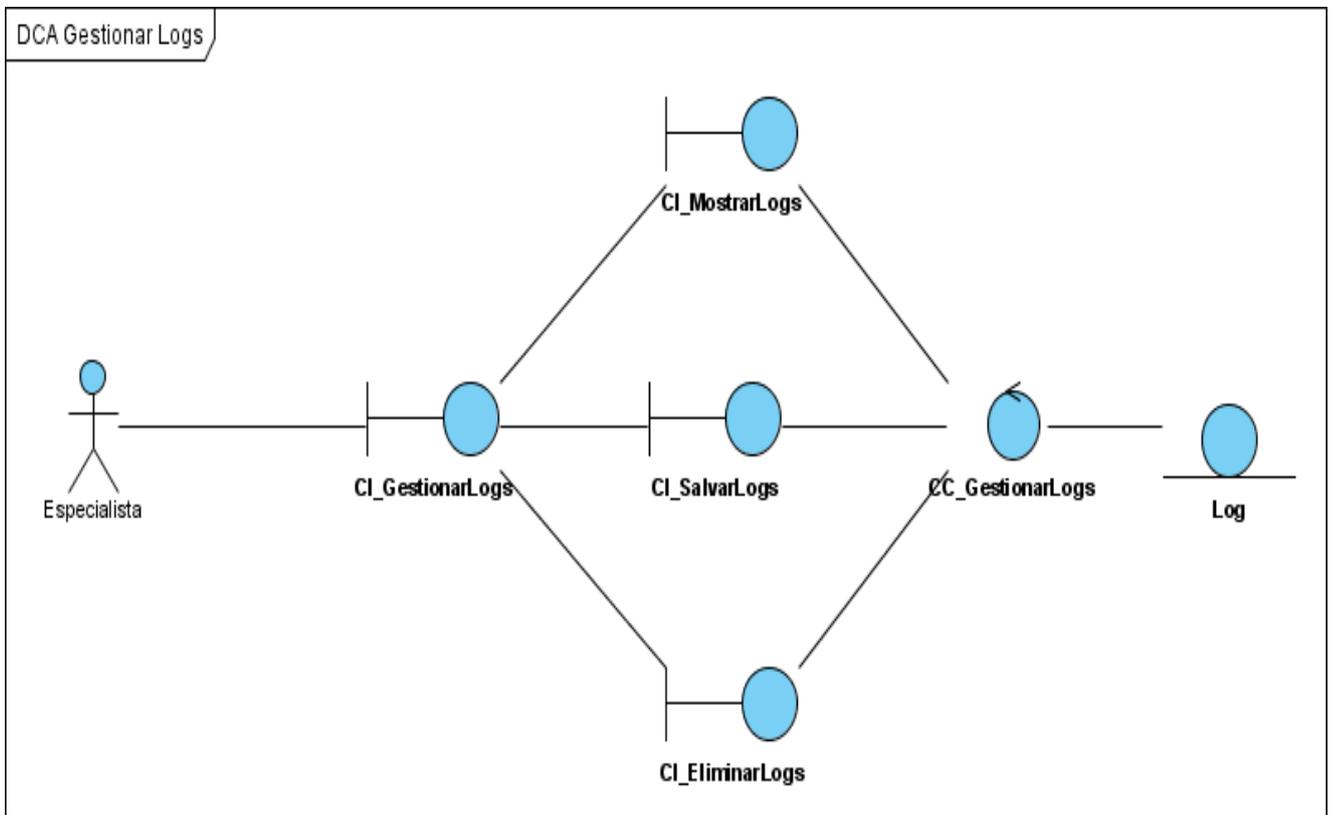


Figura 34 DCA. Caso de uso "Gestionar Logs".

Anexo 2. Diagramas de Interacción del Análisis. Colaboración y Secuencia.

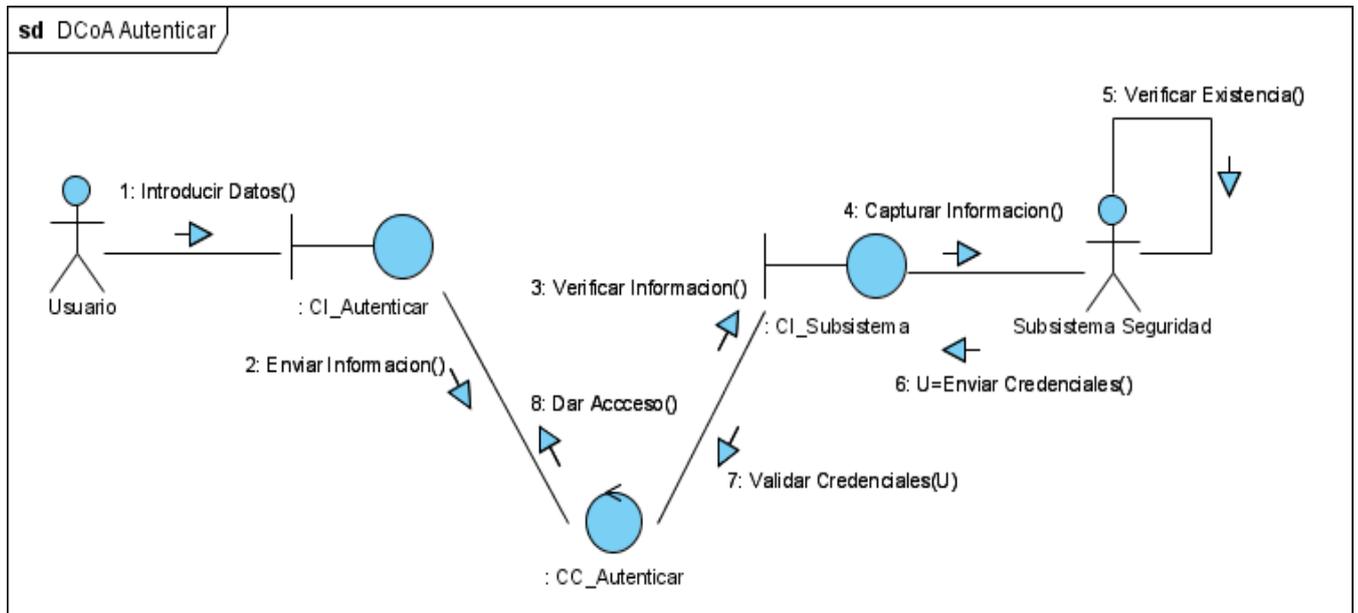


Figura 35 Diagrama de Colaboración. CU "Autenticar".

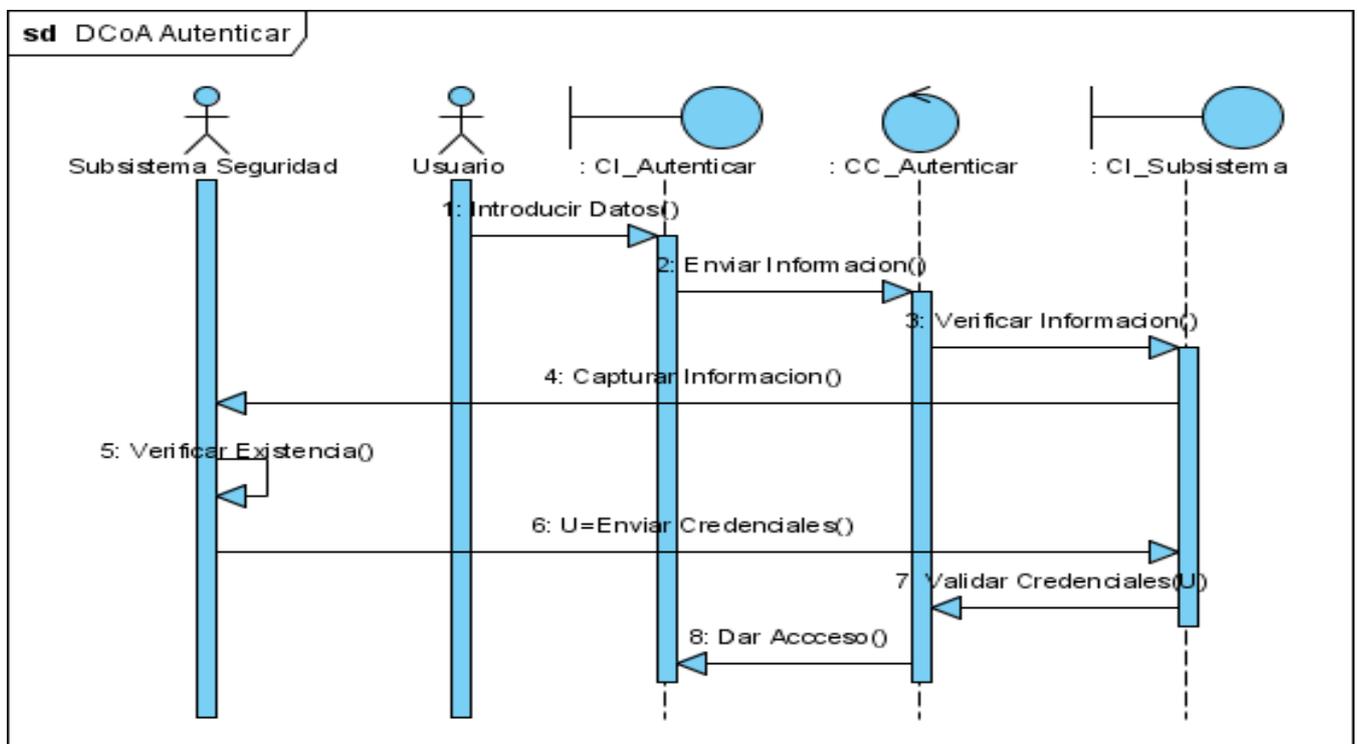


Figura 36 Diagrama de Secuencia. CU "Autenticar".

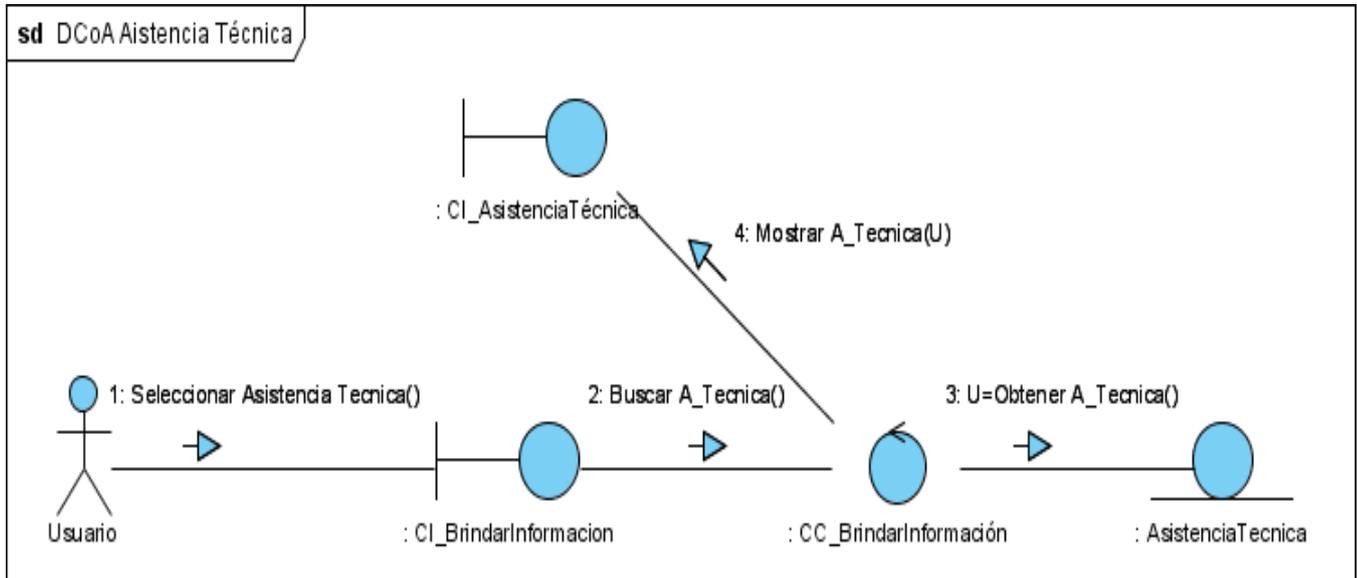


Figura 37 Diagrama de Colaboración. CU "Brindar Información". Asistencia técnica.

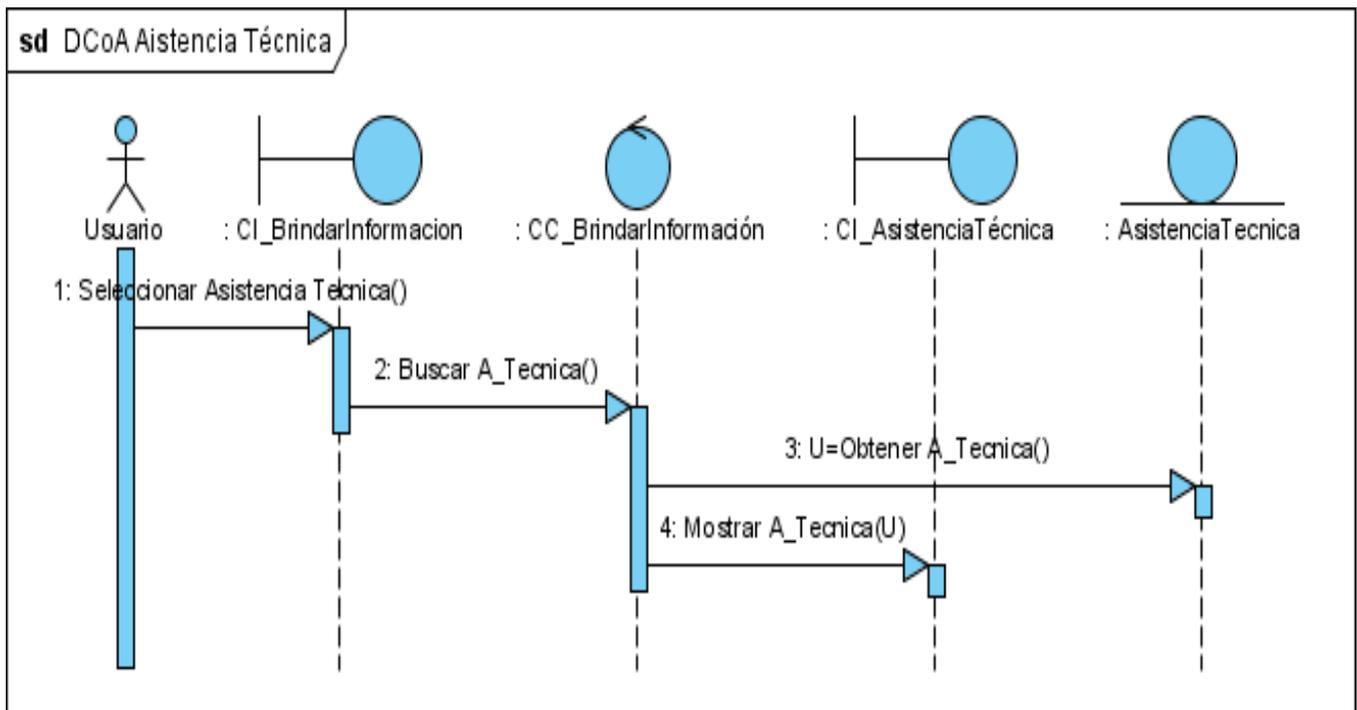


Figura 38 Diagrama de Secuencia. CU "Brindar Información". Asistencia técnica.

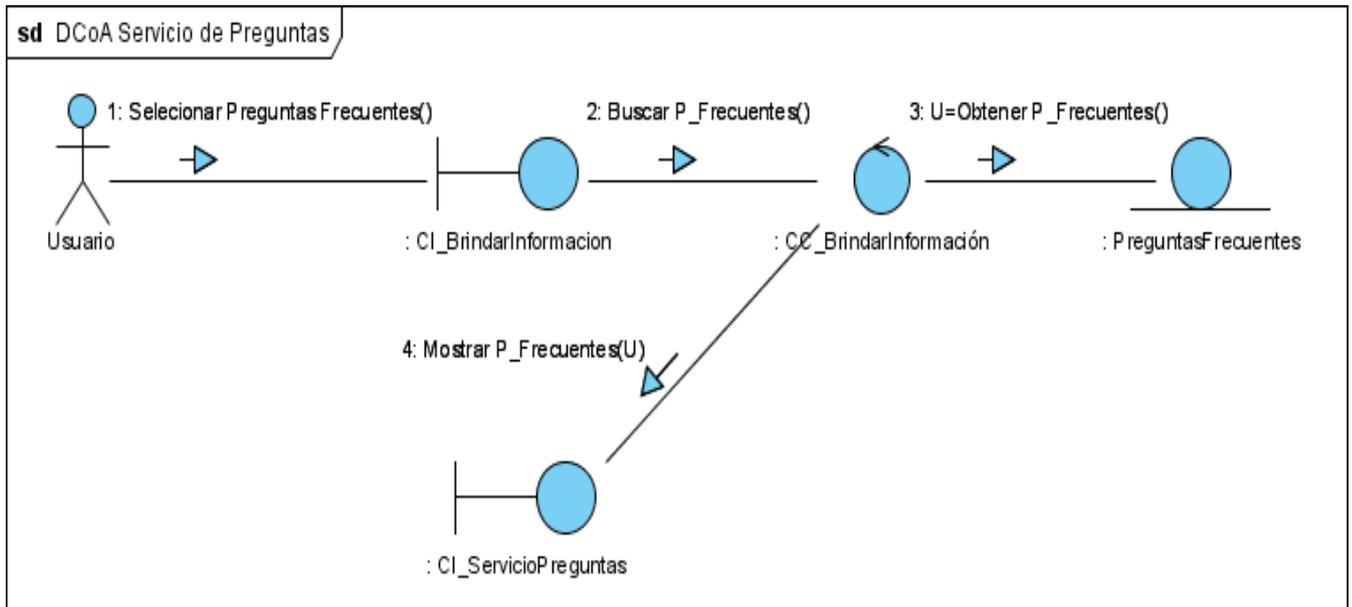


Figura 39 Diagrama de Colaboración. CU "Brindar Información". Servicio de preguntas.

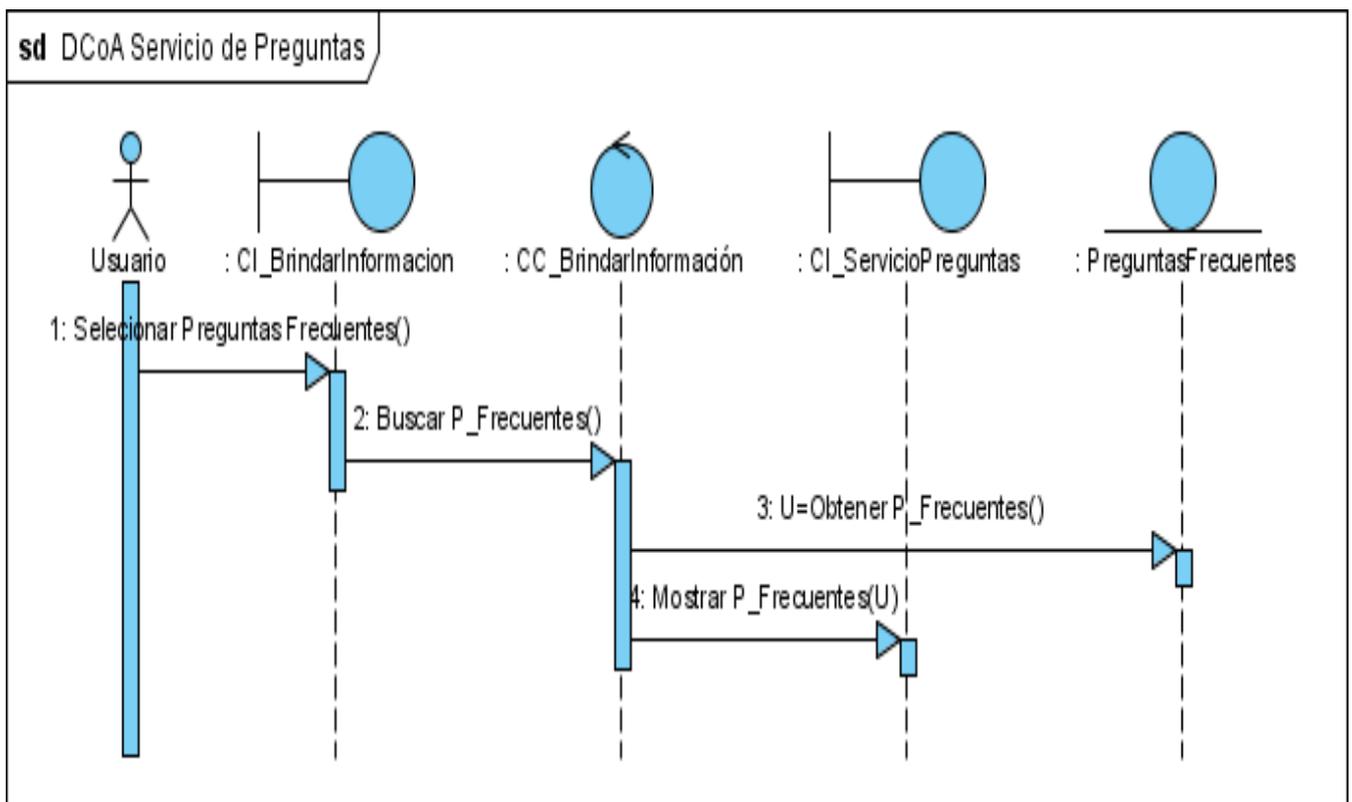


Figura 40 Diagrama de Secuencia. CU "Brindar Información". Servicio de preguntas.

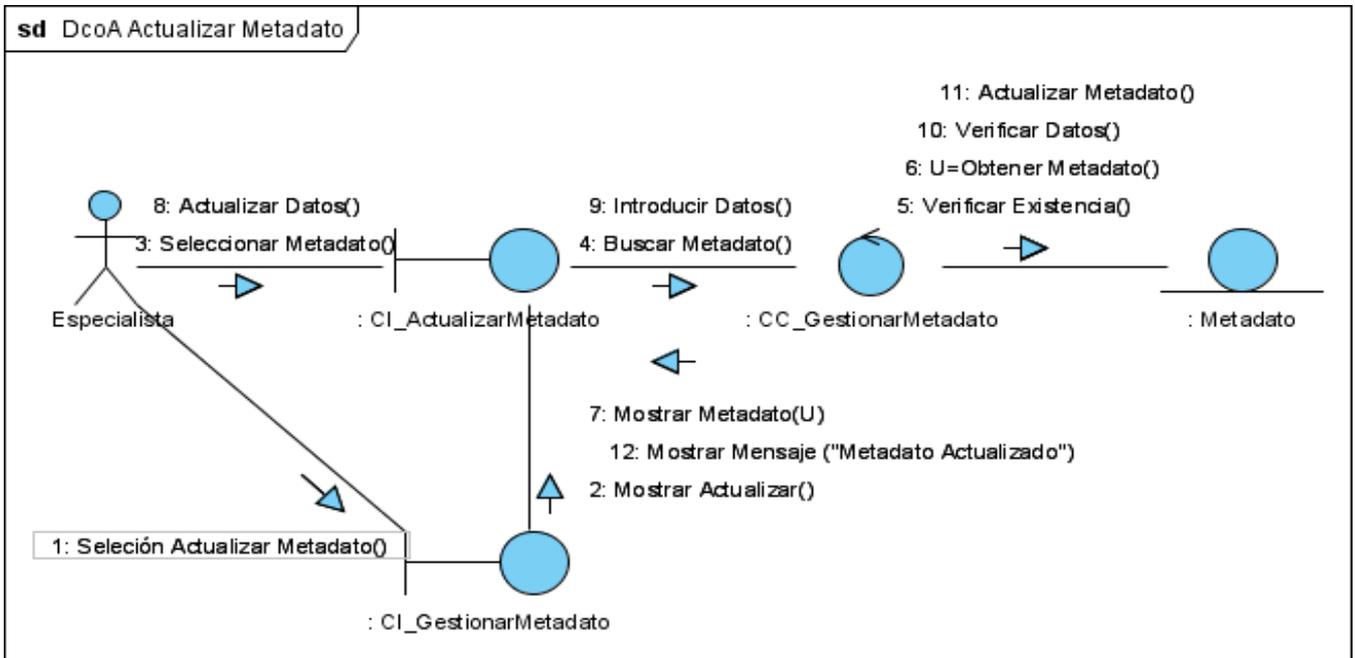


Figura 41 Diagrama de Colaboración. CU "Gestionar Metadato". Actualizar Metadato.

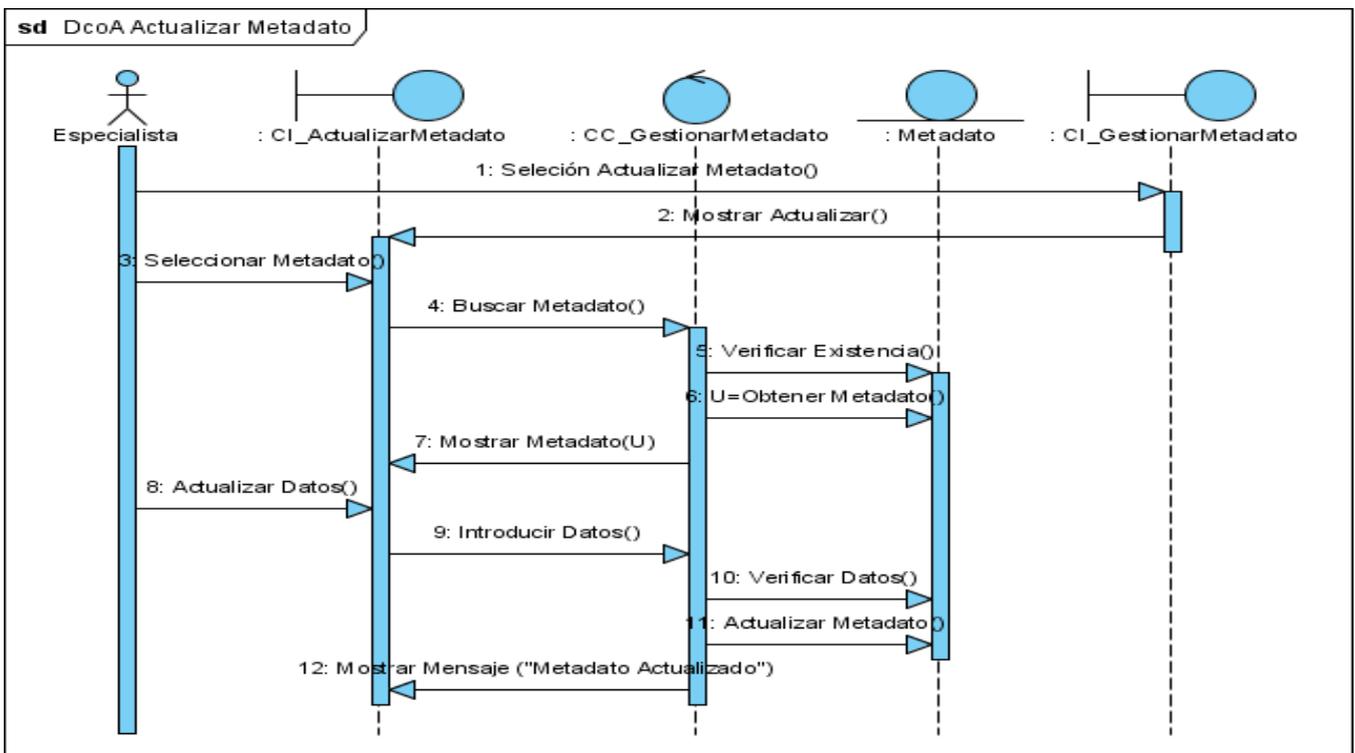


Figura 42 Diagrama de Secuencia. CU "Gestionar Metadato". Actualizar Metadato.

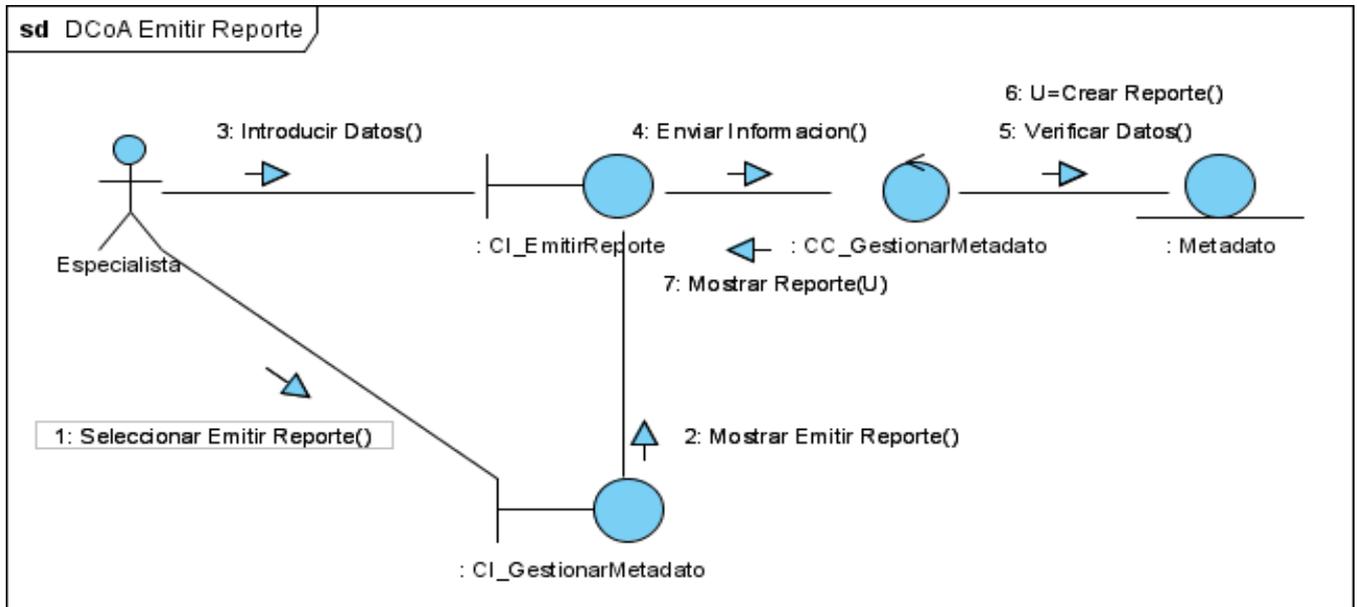


Figura 43 Diagrama de Colaboración. CU "Gestionar Metadato". Emitir Reporte.

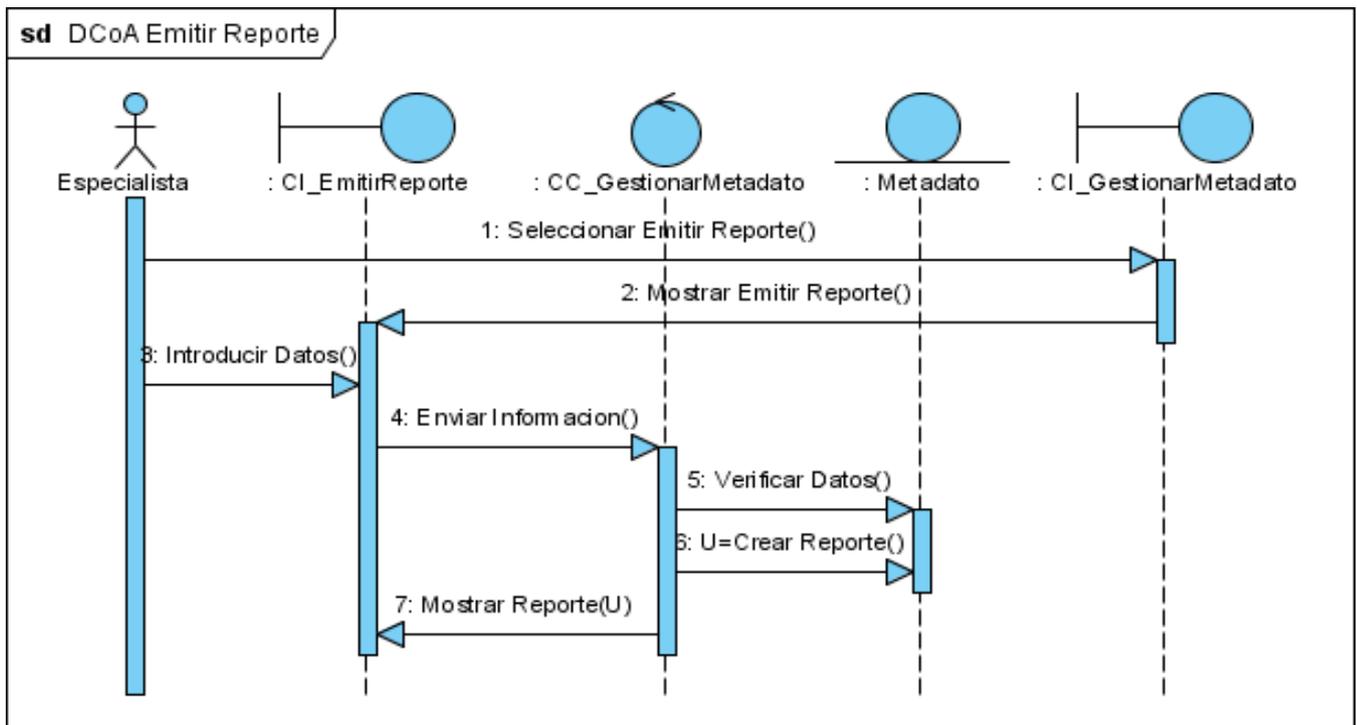


Figura 44 Diagrama de Secuencia. CU "Gestionar Metadato". Emitir Reporte.

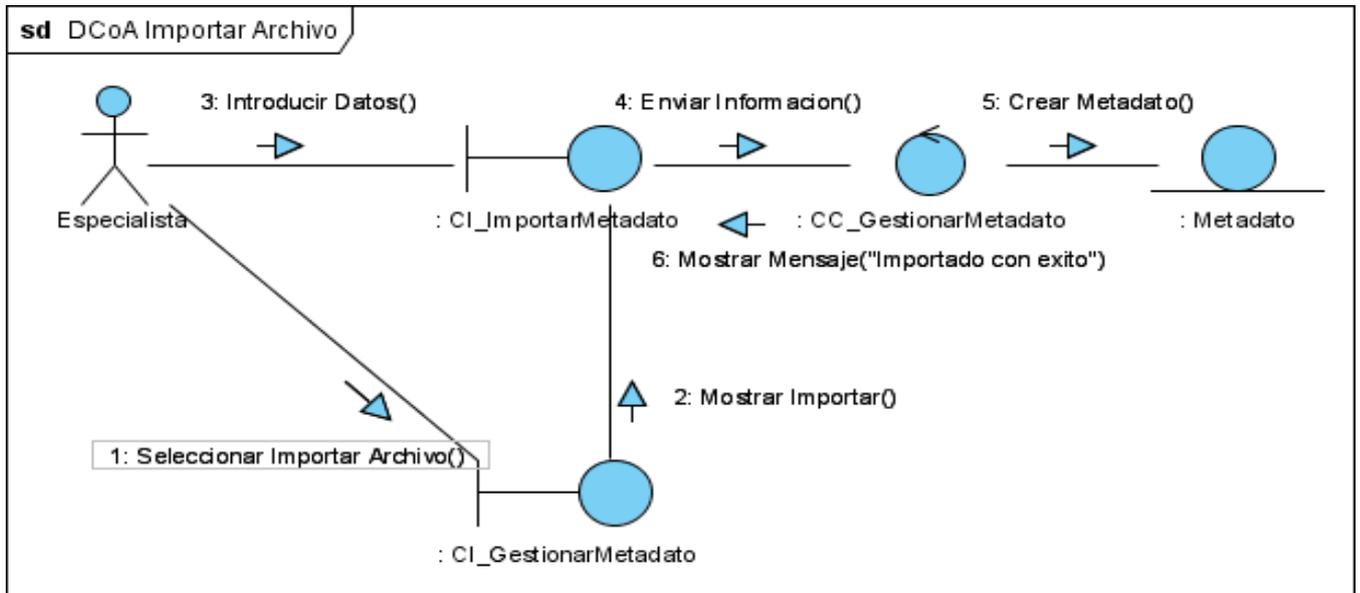


Figura 45 Diagrama de Colaboración. CU "Gestionar Metadato". Importar archivo.

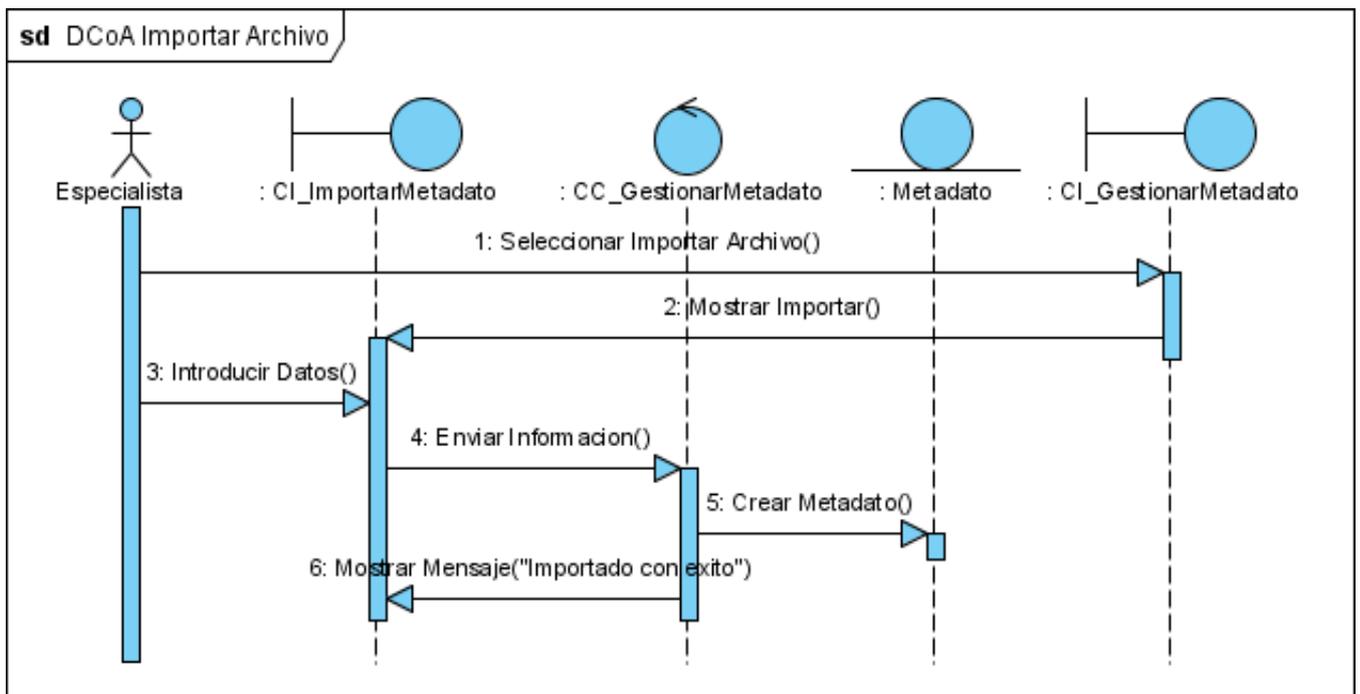


Figura 46 Diagrama de Secuencia. CU "Gestionar Metadato". Importar archivo.

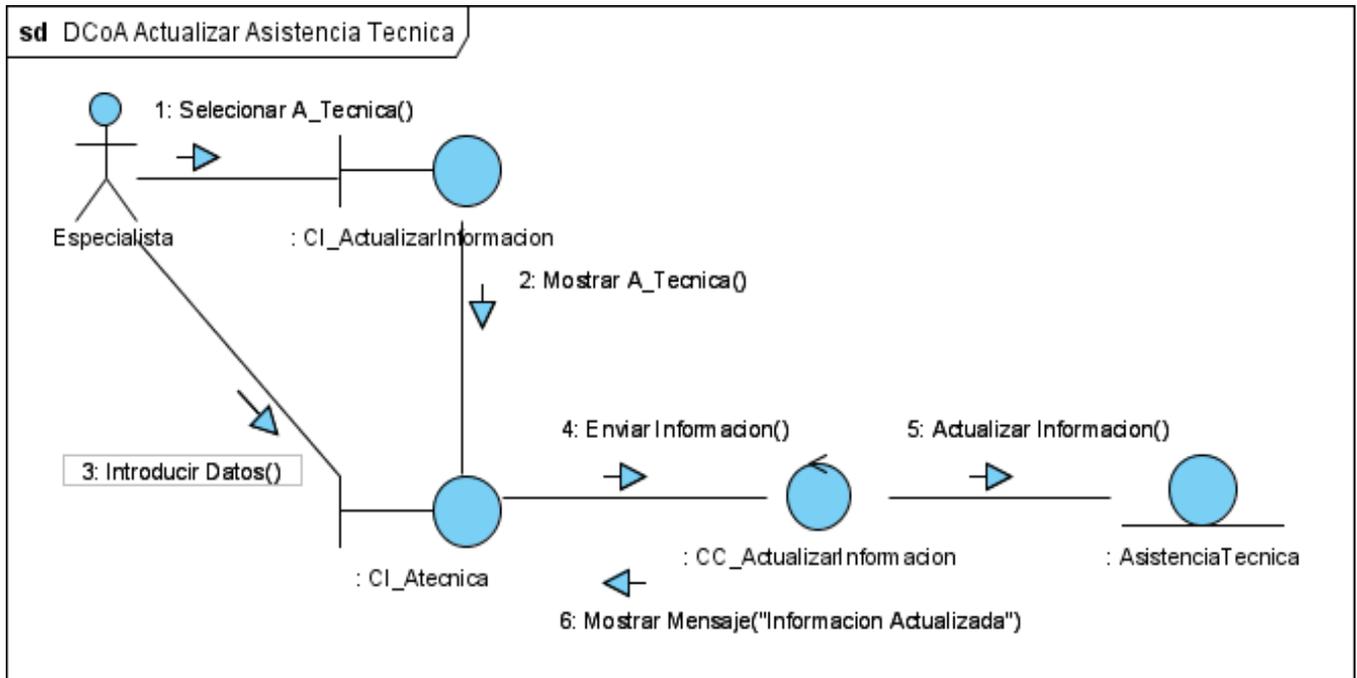


Figura 47 Diagrama de Colaboración. CU " Actualizar servicio de Información ". Actualizar Asistencia Técnica.

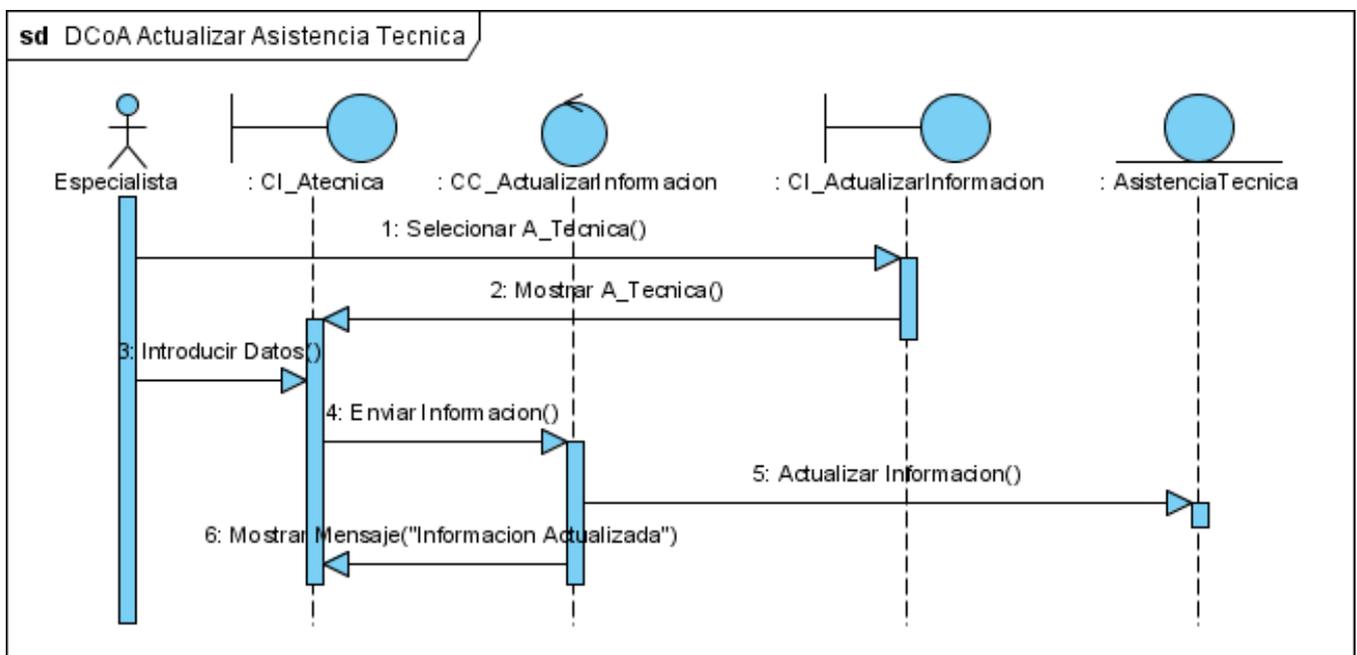


Figura 48 Diagrama de Secuencia. CU " Actualizar servicio de Información ". Actualizar Asistencia Técnica.

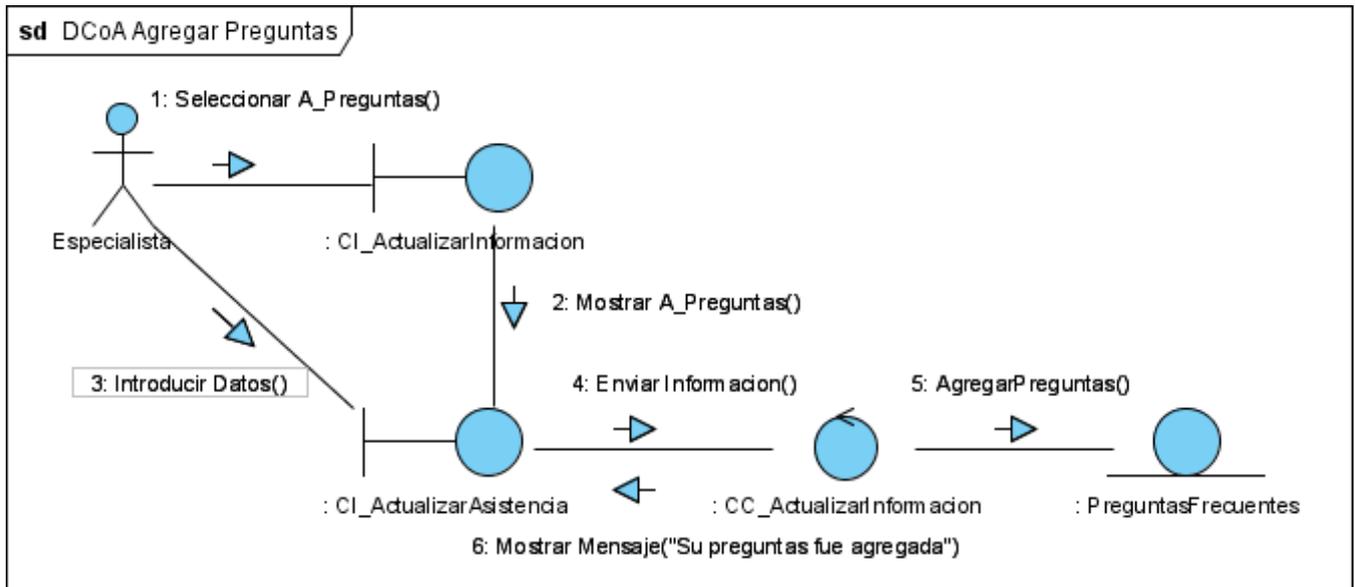


Figura 49 Diagrama de Colaboración. CU " Actualizar servicio de Información". Agregar Preguntas.

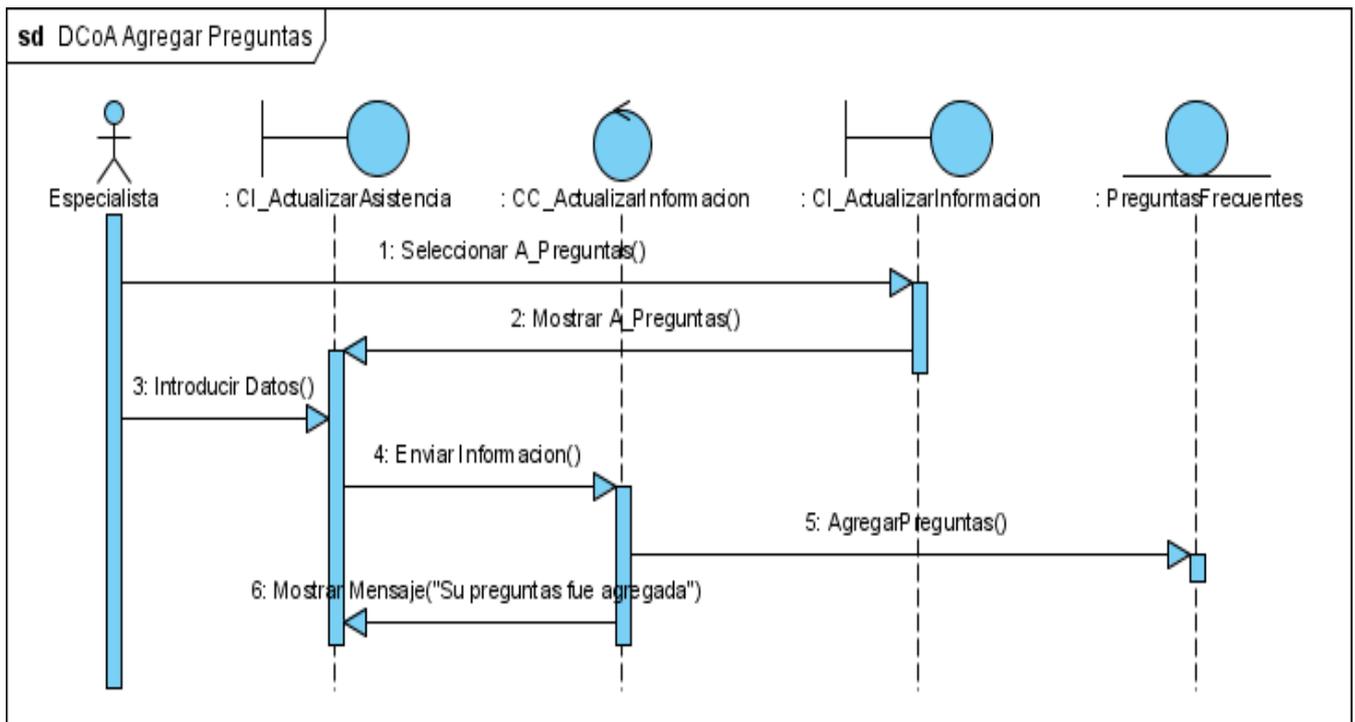


Figura 50 Diagrama de Secuencia. CU " Actualizar servicio de Información". Agregar Preguntas.

Anexo 3. Estructura de clases generadas por Symfony. Modelo genérico.

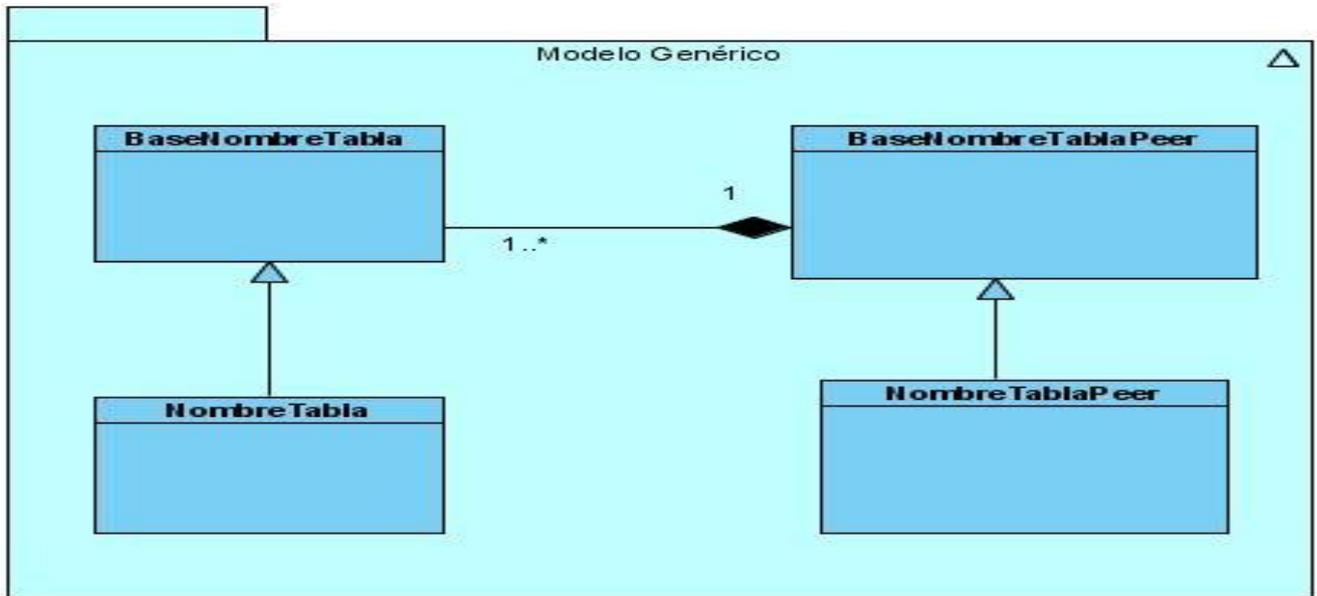


Figura 51 Estructura de clases generadas por Symfony. Modelo genérico.

Anexo 4. Generalidades sobre la relación entre las Action y el paquete Modelo.

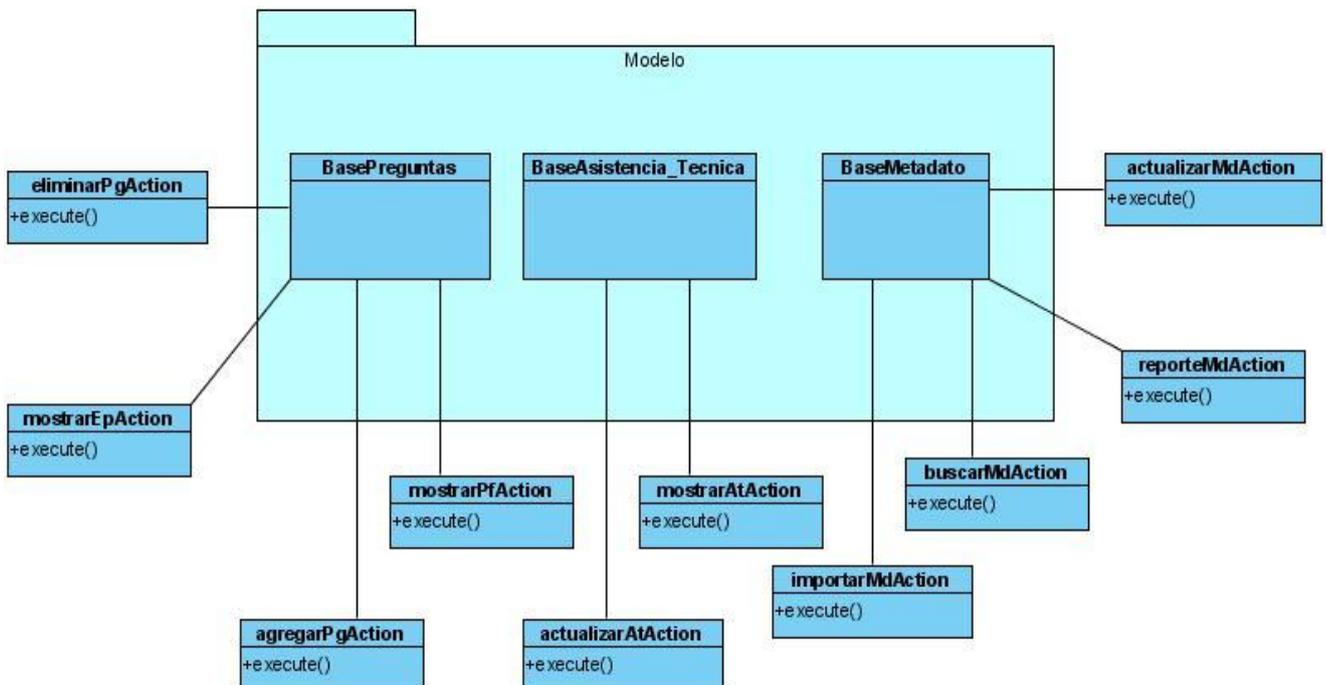


Figura 52 Action relacionadas con el paquete Modelo.

Anexo 5. Entrevista realizada.

Estimado trabajador de la ONRM, somos estudiantes de 5to año de la Universidad de las Ciencias Informática y estamos realizando una investigación referida a implementar una Aplicación Web capaz de captar ficheros de metadatos geológicos. Su experiencia y criterios serán de gran ayuda para la investigación.

Datos personales del entrevistado

Nombre _____

Ocupación dentro de la institución: _____

1. Actualmente, ¿Cómo documentan los datos geológicos?
2. ¿En que dispositivo de almacenamiento, almacenan los datos geológicos documentados?
3. ¿Qué formato o estándar, utilizan para documentar los datos almacenados?
4. ¿Cómo se encuentra distribuido esos datos o conocimiento geológicos en el país? ¿Existe algún centro principal de almacenamiento de la información?
5. ¿Existe una persona en específico encargada de realizar la documentación de los datos geológicos? ¿Quién se encarga de asignar esa persona?
6. Una vez documentada esa información, ¿alguien se encarga de revisar si esta tarea se ha realizado correctamente?
7. Luego de estar documentado un dato geológico determinado, ¿Quiénes son los que tienen acceso a esta información? Y ¿Qué nivel de acceso tendría sobre esta?
8. Al ser documentado el dato geológico, a través de metadatos geológicos, la persona que tiene acceso a esta información, ¿puede modificarla?
9. Cuando se generan los ficheros de metadatos geológicos, ¿para realizar esta tarea se debe seguir algún formato o estándar internacional?
10. ¿Cuál es el principal objetivo de generar ficheros de metadatos geológicos?
11. Además de la utilidad mencionada, ¿estos ficheros se generan con algún otro propósito específico?
12. ¿Qué ventajas traería a la ONRM contar con una aplicación Web capaz de captar ficheros de metadatos geológicos?

GLOSARIO

MINBAS: Ministerio de la Industria Básica.

ONRM: Oficina Nacional de Recursos Minerales.

PNICG: Programa Nacional de Informatización del Conocimiento Geológico de la Oficina Nacional de Recursos Minerales.

RUP: Proceso Unificado de Desarrollo.

UML: Lenguaje Unificado de Modelado.

SGBD: Sistema gestor de base de datos.

RF: Requisitos Funcionales.

RNF: Requisitos no Funcionales.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos, asociación estadounidense dedicada a la estandarización. Es una asociación internacional sin fines de lucro formada por profesionales de las nuevas tecnologías, como ingenieros de telecomunicaciones, ingenieros electrónicos, ingenieros en informática y otros (The Institute of Electrical and Electronics Engineers).

CU: Caso de Uso.

CUN: Término usado para referirse a Caso de Uso del Negocio.

CUS: Caso de Uso del Sistema.

HTTP: Protocolo para transferir archivos o documentos hipertexto a través de la red. (Hyper Text Transmisión protocolo).

UNIX: Sistema operativo portable, flexible, potente, con entorno programable, multiusuario y multitarea, muy difundido.

PHP: Es acrónimo de Hipertext Pre-processor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación (Personal Home Page).

www: Es el universo de información accesible a través de Internet, una fuente inagotable del conocimiento humano. Es un sistema de información global, interactiva, dinámica, distribuida, gráfica, basada en hipertexto, con plataforma de enlaces cruzados, que se ejecutan en Internet (Word Wide Web).

C++: Es un lenguaje híbrido, que se puede compilar y resulta más sencillo de aprender para los programadores que ya conocen C. Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Las principales características son abstracción (encapsulación), el soporte para programación orientada a objetos

(polimorfismo) y el soporte de plantillas o programación genérica (templates). Es un lenguaje que abarca tres paradigmas de la programación: La programación estructurada, la programación genérica y la programación orientada a objetos.

Java: Lenguaje de programación orientado a objetos con el que se puede realizar cualquier tipo de programa, es un lenguaje muy extendido, es un lenguaje independiente de la plataforma, es compilado en un bytecode que es interpretado desarrollado por la compañía Sun Microsystems a principios de los 90.

C#: Lenguaje de programación orientado a objetos, evolución del lenguaje C++, desarrollado por Microsoft.

GNU: Conjunto de programas desarrollados por miembros de la Fundación por el Software Libre, son de uso gratuito (FSF- Free Software Foundation).

GNU/LINUX: Es un sistema operativo, es una implementación de libre distribución UNIX para computadoras personales (PC), servidores, y estaciones de trabajo. Es multitarea, multiusuario, multiplataforma y multiprocesador.

GPL: Es una licencia creada por la Free Software Foundation y orientada principalmente a los términos de distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software Libre (General Public License).

OOP: Programación Orientada a Objetos.

MVC: Modelo Vista Controlador.

PC: Personal Computer.

XML: Extensible Markup Language. Es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium. Orientado principalmente al almacenamiento, procesamiento y transmisión de mensajes.