

Universidad de la Ciencias Informáticas

Facultad 9



**TRABAJO DE DIPLOMA PARA OPTAR POR EL
Título de Ingeniero en Ciencias Informáticas**

Título

Sistema automatizado para la gestión de datos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales.



Autores

Lisdaynet Gómez Espinosa.

Dayris Espinosa Ronquillo.

Tutores

Ing. Eyllin Hernández Luque.

Ing. Daniel Burgos Hechavarría.

Co-tutor

Lic. Jorge Castillo Burgos.

Ciudad de La Habana, junio, 2008.

“Año 50 de la Revolución”

DEDICATORIA

Dayrís

A mis padres Leticia y Vicente, por ser esas personas maravillosas que me lo han dado todo, por su ejemplo intachable, por confiar siempre en mí, por hacer de mí la persona que soy hoy.

A mi queridísima hermanita Daylin por ser tan dulce, quererme tanto y siempre saber salir adelante en los momentos difíciles sin mi presencia física pero sí espiritual, por tener ese corazón tan noble, por ser tú esa otra parte de mí que tanto quiero.

A mi novio, amigo, compañero, mi todo, Enrique Félix por darme tanto amor y apoyo en estos años de Universidad, por ser cómplice de mis secretos y mis sueños, por ser mi primer y único amor, por ser ese hombre que es, por ser ese ejemplo intachable a seguir, por ser sólo tú.

A mi familia que tanto apoyo me han dado durante toda la carrera y han sabido cuidar a mis padres y hermana mientras estoy lejos de ellos.

Lisdaynet

A mi papá que siempre ha sido mi faro.

A mi mamá por toda la comprensión y confianza que me ha dado.

A mis abuelos por haberme entregado todo el amor del mundo.

A mi hermana Liannys para que se inspire y siga adelante en la vida.

A todos los miembros de mi familia, por todos estos años de sacrificio, para poder lograr mi sueño de graduarme, venciendo la distancia y el tiempo.

A quien me ayudó a sobrevivir lejos del amor y los brazos de mi abuela María.

A mi amiga y hermana Sarhay porque nunca ha importado la distancia para mantenernos juntas y ayudarnos.

Al amor de mi vida por todo los días especiales que vivo con él.

En fin a todos aquellos que por ser como son y formar parte de mi historia, se han ganado un pedacito dentro de mi corazón.

AGRADECIMIENTOS

Dayrís

A mis padres y hermana por siempre estar presente cuando más los he necesitado en mi vida, por siempre confiar en mí, nunca los defraudaré.

A mi novio Enrique Félix por siempre estar ahí en todo momento, por darme aliento y ánimo cuando pensaba que todo estaba perdido, por ser esa guía importante en mi carrera, por brindarme esa mano amiga que siempre se necesita, por aguantar mis malacrianzas, por quererme tanto, no se que habría sido de mí sin ti. Te amo

A mis tías y tíos (Yoli, Maira, Margarita, Macho y Emilio) por siempre brindarme su mano y contribuir a que hoy pueda discutir mi tesis.

A mis abuelas (Estelvina y Yolanda) por todo su amor, por guiarme por caminos seguros y verdaderos.

A mis primos, especialmente a Maité mi prima querida, por ser esa otra hermana que tanto quiero.

A mis suegros (Marusa y Agudo) que tanto apoyo incondicional me han brindado, convirtiéndose en mis segundos padres.

A mi cuñado (Félix Manuel) por ser amigo en todo momento y tener ese noble corazón que tanto admiro.

A mi compañera de tesis Lisdaynet por ser esa persona tan dedicada y admirable, por pasar tontas noche y horas frente a una PC junto a mí, haciendo posible que este día llegara.

A mis tutores Eylin y Daniel por todo el apoyo y conocimiento brindado, por soportar mis acelerones y comprender en cada momento la necesidad de su ayuda.

A mis compañeros de Universidad con los cuales pasé momentos inolvidables, especialmente a Yojania mi amiga del alma, Jorgito, Marcos y Jose.

A todos aquellos que de una forma u otra han contribuido a mi formación profesional.

Lisdaynet

En primer lugar a mis padres, a mis abuelos y mi tío por hacer de mi esa persona en que me he convertido, con todos sus consejos y buenas acciones pero también su severidad ante lo mal hecho.

A todos aquellos los que me han enseñado y preparado en la vida desde mi querida profesora Milagros en preescolar, que me enseñó las primeras cosas, hasta los que hoy me han dado el conocimiento para ser ingeniera informática y dar mi aporte por mi país.

A mi amiga y compañera de tesis Dayris por las incontables muestra de amistad y por todo el apoyo dado en los momentos difíciles por lo que he pasado en este año.

A mis tutores, amigos y compañeros de aula que me ayudaron a que todo saliera bien.

DECLARACIÓN DE AUTORÍA

Nosotras, Lisdaynet Gómez Espinosa y Dayris Espinosa Ronquillo, declaramos que somos las únicas autoras de este trabajo y autorizamos a la Facultad 9 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Lisdaynet Gómez Espinosa (Autor)

Dayris Espinosa Ronquillo (Autor)

Ing. Eyllin Hernández Luque (Tutor)

Ing. Daniel Burgos Hechavarría (Tutor)

DATOS DE CONTACTO

Síntesis de la Tutora: Ing. Eylin Hernández Luque

Profesión: Ingeniero en Ciencias Informáticas

Años de graduado: 1

Correo Electrónico: ehernandezl@uci.cu

Síntesis de la Tutora: Ing. Daniel Burgos Hehavarria

Profesión: Ingeniero Informático

Años de graduado: 2

Correo Electrónico: dburgos@uci.cu

OPINIÓN DEL TUTOR

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.

Las diplomantes Lisdaynet Gómez Espinosa y Dayris Espinosa Ronquillo, han demostrado compromiso y esmero en la realización de la investigación científica. Su dinamismo y laboriosidad han sido un factor fundamental en el cumplimiento exitoso del objetivo trazado en dicha investigación. Las estudiantes mostraron independencia absoluta, originalidad y autonomía en cada fase de la tarea a realizar, logrando resultados satisfactorios y buena aceptación de la información y la modelación del trabajo. Además las 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 los objetivos perseguidos con su trabajo. La modelación del análisis y diseño de la aplicación cumple con el objetivo trazado, lo cual brindan beneficios sociales y económicos para la gestión del conocimiento geológico en la Oficina Nacional de Recursos Minerales.

Las diplomantes son muy exigentes en sus labores, se esmeraron para terminar su trabajo de diploma en tiempo y forma.

Por todo lo anteriormente expresado, consideramos que las estudiantes están aptas 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

Ing. Daniel Burgos Hechavarría

Tutor

RESUMEN

En la Oficina Nacional de Recursos Minerales (ONRM), entidad minera regente en Cuba, existe el Departamento de la Dirección Técnica; donde los especialistas a cargo, tienen que realizar el proceso del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales (BNRDEAM). Este balance se realiza de forma manual, producto a la inexistencia de un software que facilitará y agilizará los procesos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales.

Como parte de todo el levantamiento de requisitos para informatizar el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales , por la importancia y novedad que tiene para el país, se hace necesaria una propuesta de software para llevar a cabo el BNRDEAM.

Este trabajo siguió el objetivo de realizar el análisis y el diseño para una aplicación Web que facilite la administración y gestión eficiente de los datos que tributan al Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales. Logrando describir detalladamente todos los procesos que se desarrollan para realizar un balance eficiente, así como la administración y recuperación de los datos, dada la situación problemática y las necesidades que tenían la Oficina Nacional de Recursos Minerales(ONRM). Se realiza y se muestran los resultados del estudio sobre las tecnologías más idóneas. Además se exponen la propuesta y la construcción de la solución a través de los diferentes diagramas de análisis y diseño para la aplicación que se propone, así como el estudio de la factibilidad para dicha propuesta.

PALABRAS CLAVES

- **BNRDEAM**
- **Gestión**
- **Aplicación Web**

TABLAS Y FIGURAS

Tabla 1 Actores del Negocio.	42
Tabla 2 Trabajadores del Negocio.	43
Tabla 3 Descripción CU_Entregar Informe de Investigación.	44
Tabla 4 Descripción CU_Entregar Proyecto de Explotación.	45
Tabla 5 Descripción CU_Entregar Balance.	46
Tabla 6 Descripción CU_Consultar Información.	47
Tabla 7 Descripción de los Actores.	50
Tabla 8 Descripción CU_Gestionar Datos Yacimiento.	54
Tabla 9 Descripción CU_Gestionar Recursos Disponibles.	58
Tabla 10 Descripción CU_Gestionar Recursos de Explotación.	62
Tabla 11 Descripción CU_Generar Balance General.	63
Tabla 12 Descripción CU_Mostrar Información Yacimiento.	65
Tabla 13 Descripción CU_Mostrar Información Materia Prima.	66
Tabla 14 Descripción CU_Mostrar Búsquedas Especializadas.	68
Figura 1 Fases y Flujos de Trabajo de RUP.	19
Figura 2 Arquitectura 3 capas.	27
Figura 3 Comparación Modelo Clásico de una aplicación Web y Modelo Ajax de una aplicación Web.	32
Figura 4 Diagrama de Casos de Uso del Negocio.	43
Figura 5 Diagrama de Objetos del Negocio.	47
Figura 6 Diagrama de Casos de Uso del Sistema.	68
Figura 7 Diagrama de Paquetes del Análisis.	71
Figura 8 Diagrama de Clases Persistentes.	80
Figura 9 Modelo de Datos.	81
Figura 10 Diagrama de Despliegue.	82
Figura 11 Diagrama de Componente.	83
Figura 12 Paquete Componente Symfony.	83
Figura 13 Paquete Actions.	84
Figura 14 Paquete Success.	84
Figura 15 Paquete Acceso a Datos.	85
Figura 16 Diagrama de Interacción del Análisis, Sección Adicionar Yacimiento, CU_Gestionar Datos Yacimiento.	95
Figura 17 Diagrama de Interacción del Análisis, Sección Mostar Yacimiento, CU_Gestionar Datos Yacimiento.	96
Figura 18 Diagrama de Interacción del Análisis, Sección Buscar Yacimiento, CU_Gestionar Datos Yacimiento.	96
Figura 19 Diagrama del Diseño, Sección Adicionar Datos Yacimiento, CU_Gestionar Datos Yacimiento.	97
Figura 20 Diagrama del Diseño, Sección Modificar Datos Yacimiento, CU_Gestionar Datos Yacimiento.	98

Figura 21 Diagrama del Diseño, Sección Buscar Datos Yacimiento, CU_Gestionar Datos Yacimiento..... 99



INDICE

INTRODUCCIÓN	1
Estructuración del contenido.....	6
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA.	7
1.1 Introducción.....	7
1.2 Conceptos asociados al dominio del problema.....	7
1.2.1 Geología.....	7
1.2.1.1 Hidrogeología.....	7
1.2.2 Balance.....	7
1.2.2.1 Recursos.....	7
1.2.2.2 Recursos Disponibles	8
1.2.2.3 Recursos Explotables o de Explotación.....	8
1.2.2.4 Recursos Asegurados o Caudal Seguro.....	8
1.2.3 Aguas.....	8
1.2.3.1 Aguas Minerales	9
1.3 Objeto de estudio	9
1.3.1 Descripción General.....	9
1.3.2 Descripción Actual del Dominio del Problema	10
1.3.3 Situación Problemática	11
1.4 Análisis de otras soluciones existentes	12
1.5 Conclusiones	14
2.1. Introducción.....	15
2.2. Aplicaciones Web.	15
2.3. Software Libre.....	16
2.3.1. Distribución de Linux.....	17
2.4. Metodologías de Desarrollo de Software.....	17
2.4.1 Proceso Unificado de Desarrollo de Software (RUP)	18
2.4.2 Programación Extrema (XP)	19
2.4.3 Desarrollo guiado por funcionalidad (FDD)	20

2.4.4 Selección de la metodología a utilizar	21
2.5. El Lenguaje Unificado de Modelado (UML) como soporte de lenguaje orientado a objeto para el modelado de aplicaciones Web.....	21
2.6. Herramientas CASE de Desarrollo de Software	23
2.6.1. Rational Rose	23
2.6.2. Visual Paradigm	24
2.6.3. Selección de la Herramienta CASE a utilizar	25
2.7. Arquitectura en Capas.....	25
2.7.1 Arquitectura en 3 capas.....	26
2.8. Sistema Gestor de Base de Datos	27
2.8.1. MySQL.....	28
2.8.2. Postgre.	28
2.9. Lenguajes de programación.....	30
2.9.1. Lenguajes de programación del lado del cliente. Java Script	31
2.9.2. Lenguajes de programación del lado del cliente. Ajax	31
2.9.3. Lenguajes de programación del lado del servidor. ASP.NET	33
2.9.4. Lenguajes de programación del lado del servidor. PHP	34
2.10. IDE de desarrollo	35
2.10.1. Eclipse.....	36
2.10.2. Zend Studio	37
2.10.3. Selección del IDE de Desarrollo a utilizar.	38
2.11. Framework.....	39
2.11.1. Symfony	39
2.12. Conclusiones	40
CAPÍTULO 3 PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.....	41
3.1. Introducción	41
3.2. Modelo de Negocio.....	41
3.2.1. Reglas del negocio	41
3.2.2. Actores del negocio	42
3.2.4 Diagrama de Casos de Uso del Negocio.....	43
3.2.5. Descripción de los Casos de Uso del Negocio	43

3.2.5.1. Descripción CU_Entregar Informe de Investigación.....	43
3.2.5.2 Descripción CU_Entregar Proyecto de Explotación.....	44
3.2.5.3. Descripción CU_Entregar Balance.....	45
3.2.5.4. Descripción CU_Consultar Información.....	46
3.2.6. Diagrama de Objetos	47
3.3 Requerimientos del sistema	48
3.4.1. Descripción de los actores	50
3.4.2. Descripción CU_Gestionar datos de Yacimientos.....	51
3.4.3. Descripción CU_Gestionar Recursos Disponibles	54
3.4.4 Descripción CU _ Gestionar Recursos de Explotación	58
3.4.5. Descripción CU_Generar Balance General.....	62
3.4.6. Descripción CU_Mostrar información Yacimiento	63
3.4.7. Descripción CU_Mostrar información Materia Prima.....	65
3.4.8. Descripción CU_Mostrar Búsquedas Especializadas.....	67
3.4.9 Casos de Uso del Sistema	68
CAPÍTULO 4	69
CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	69
4.1 Introducción.....	69
4.2 Modelo de análisis	69
4.2.1 Diagramas de clases del análisis	69
4.2.2 División por paquetes.....	70
4.3 Diagramas de interacción.....	71
4.3.1 Diagrama de Colaboración.....	71
4.4 Modelo de Diseño	71
4.4.1 Diagrama de Clases del Diseño	72
4.5 Principios de Diseño	77
4.6 Diseño de la base de datos	79
4.6.1 Diagrama de clases persistentes	80
4.6.2 Modelo de datos.....	81
4.6 Diagrama de Despliegue	82

4.7 Modelo de Implementación.	82
4.7.1 Diagrama de componentes.	82
4.7.1.1 Diagrama de componente.....	83
4.7.1.2 Paquete Componente Symfony.	83
4.7.1.3 Paquete Actions.....	84
4.7.1.4 Paquete Success.....	84
4.7.1.5 Paquete Acceso a Datos.....	85
4.8 Conclusiones parciales.	85
Conclusiones Generales.....	86
Recomendaciones.....	87
Referencias Bibliográficas.....	88
Bibliografía Consultada.....	90
Glosario de Términos.....	92
Anexos.....	94
Anexo#1: Tabla que tributa al Balance Nacional de Recursos Disponibles y de Explotación.	94
a) Tabla Resumen General de los Yacimientos de Aguas Minerales.....	94
Anexo#2: Diagramas de clases del análisis.....	94
a) Del CU_Gestionar Datos Yacimiento.....	94
Anexo#3: Diagramas de iteración del Análisis.....	95
a) Del CU_Gestionar Datos Yacimiento, sección Adicionar Yacimiento.....	95
b) Del CU_Gestionar Datos Yacimiento, sección Modificar Yacimiento.....	95
c) Del CU_Gestionar Datos Yacimiento, sección Buscar Yacimiento.	96
Anexo#4: Diagramas del Diseño.....	97
a) Del CU_Gestionar Datos Yacimiento. Sección Adicionar Datos Yacimiento.	97
b) Del CU_Gestionar Datos Yacimiento. Sección Modificar Datos Yacimiento.....	98
c) Del CU_Gestionar Datos Yacimiento. Sección Buscar Datos Yacimiento.....	99

INTRODUCCIÓN

Desde tiempos muy remotos el hombre siempre ha estado observando la naturaleza, aprendiendo de ella y utilizándola para satisfacer sus necesidades a medida que la va conociendo mejor.

En Europa, específicamente en Italia, la explotación de manantiales calientes tiene orígenes antiguos. Los etruscos utilizaron los abundantes manantiales termales en su territorio con fines terapéuticos, aprendieron los elementos de la medicina sagrada, que tenía el carácter de las artes mágicas, de los griegos. Las curaciones se realizaban en los templos dedicados a Esculapio, erigidos en sitios considerados como convenientes para la curación en virtud de su ubicación y belleza natural. Uno de los prerrequisitos de estos lugares era contar con un manantial. [1]

En el avance de la historia, se han realizado grandiosos aportes científicos, tecnológicos, económicos, culturales y sociales, producto del ahínco de los hombres por desarrollar la sociedad, dedicando muchos de ellos a la investigación geológica, desmesurados en cada una de sus ramas.

El estudio de los microorganismos del agua comenzó en el siglo XIX, siendo su principal objeto las bacterias patógenas, con el fin de controlar las enfermedades transmitidas por este medio. Los primeros estudios microbiológicos de aguas minerales, realizados en España, se remontan a 1838 cuando el farmacéutico Pablo Prolongo describe la naturaleza orgánica de los copos que flotan en las aguas del balneario de Carratraca. En los años siguientes comenzaron a realizarse estudios microbiológicos más completos como el efectuado por el Dr. Santiago García Fernández en el Balneario de Arbieta (Vizcaya) en 1897, en el que además de las observaciones microscópicas incluyó recuentos y cultivos de los microorganismos.[2]

En la actualidad, las investigaciones relacionadas con la acción terapéutica de las aguas mineromedicinales, se realizan principalmente en los Centros Termales y van dirigidas a relacionar la composición química y la acción de las aguas sobre el organismo o sobre algún órgano determinado, así como al estudio de las propiedades farmacocinéticas de estas aguas, con el fin de conocer la absorción y distribución y eliminación de las mismas por el organismo.[3]

El agua es imprescindible para la vida, siendo el componente inorgánico más abundante de los seres vivos. En el hombre llega a constituir más del 60 % del peso corporal, y en algunos animales, como por ejemplo la medusa, el 98 %. Sin el agua no es posible la realización de los procesos metabólicos de

las células. El agua interviene químicamente en estos procesos y fuera de las células, actúa como medio de transporte en la circulación sanguínea.

En el mundo hay diversas entidades que se encargan de llevar un control de todos los recursos que la naturaleza le brinda y las aplicaciones. En Cuba también existen estas entidades. Un ejemplo real es el Ministerio de la Industria Básica (MINBAS).

Este ministerio para el cumplimiento de sus objetivos estratégicos, mantiene un amplio programa de preparación, superación y entrenamiento de sus directivos y técnicos, misión que corresponde garantizar a la Escuela Superior de la Industria Básica, para lo cual dispone de una adecuada y moderna infraestructura.

Dentro de la misma (siendo una dependencia funcional) se encuentra la Oficina Nacional de Recursos Minerales (ORMN) que es la Autoridad Minera en Cuba. 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. [4]

La ONRM está compuesta por más de 100 especialistas que laboran en sus diferentes áreas. Dicha Institución es responsable directa de la salvaguarda de todo el conocimiento geológico de la República de Cuba que actualmente está constituido por los fondos documentales de Informes, Proyectos, Estudios Técnico-Económicos, Bases de Datos y otros. Entre sus misiones fundamentales se encuentran garantizar el aprovechamiento racional de los recursos minerales del país y ejercer con eficiencia, experiencia y responsabilidad el control estatal sobre las actividades de la geología, minería, petróleo y aguas minerales. En la actualidad, aún cuando por su importancia es necesario garantizar con un soporte automático el buen desempeño de los procesos, no existe un sistema automatizado que gestione las principales funciones que allí se realizan y que hoy se están ejecutando de forma manual.

De manera general la **situación problemática** se enuncia de la siguiente forma:

En la ONRM hasta el momento no existe ningún sistema automatizado que gestione los procesos relacionados con las aguas minerales para la gestión del balance o estado anual de las mismas.

Hasta la fecha se lleva el control sobre las aguas minerales de forma manual pero por su importancia se hace necesario comenzar a realizarlo de forma diferente a través de una aplicación que gestione y guarde estos datos.

Los concesionarios (que son las entidades que trabajan ligadas a la ONRM, en el caso Ciego Montero o los Portales por citar un ejemplo) tienen que dirigirse a dicha oficina para entregar la información en copia dura o digital en CD, soportada en diferentes formatos como Word o Excel.

En la ONRM el especialista a cargo se estudia y analiza dicha información. El cual después de un trabajo exhaustivo si todo está correcto les otorga el permiso para la explotación del recurso. De no estar correcta la información, comienza un trabajo engorroso de pasarla para los concesionarios y comenzar todo el proceso nuevamente.

Al no existir algún software que guardara estos datos se llega a la deficiencia y lentitud del proceso, así como en la repercusión de no tener constancia de cómo se mantuvo la estabilidad del agua en un período de tiempo mayor a un año puesto que los registros eran instantáneos sin un historial. Como el control de las aguas minerales repercute en la toma de decisiones de las diferentes empresas que comercializan con ella dígase Ciego Montero o los Portales, hacen que aumente su urgencia dada su importancia.

Por el interés que existe, la gestión de los datos sobre dichas aguas se acordó una nueva metodología para llevar a cabo este proceso. Por lo que no sólo se trabajaría en realizar de forma automática un balance de las aguas minerales sino todo lo relacionado con el control y análisis de las aguas naturales minerales, termales, industriales, entre otras, que enmarca esta metodología.

Hoy día es evidente, que con la nueva metodología para realizar el balance, se hace necesario un sistema, para evitar la consecuencia de un retraso en el tiempo de respuesta y la eficiencia de los principales servicios que brinda esta dependencia ministerial.

De esta manera queda evidenciado que el **problema a resolver** es ¿Cómo contribuir a perfeccionar la gestión de datos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales en la ONRM para que sea más eficiente?

Asimismo el **objeto de estudio** lo constituye el Proceso de gestión de datos para el Balance Nacional de Recursos Disponibles y de Explotación de Aguas Minerales y el **campo de acción** Análisis e

informatización de los procesos de gestión para el Balance de Recursos Disponibles y de Explotación de Aguas Minerales en la Oficina Nacional de Recursos Minerales.

Se persigue con esta investigación lograr el siguiente **objetivo general**: Realizar el análisis y el diseño para una aplicación Web que facilite la administración y recuperación eficiente de los datos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales en la ONRM.

Teniendo en cuentas varios **objetivos específicos** los cuales se enuncian en los siguientes puntos:

- Valorar las características fundamentales del proceso de gestión asociados al Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales dentro de la Oficina Nacional de Recursos Minerales.
- Preparar el personal para el análisis y el diseño de la aplicación.
- Aplicar la metodología adecuada para realizar el análisis y el diseño de la aplicación.
- Verificar la correspondencia entre los requisitos, el análisis y el diseño de la aplicación.

Para lo que se proponen las siguientes **tareas a desarrollar**:

1. Comprobación de la validez del problema a resolver.
2. Estudio de datos geológicos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales.
3. Estudio sobre la gestión de datos sobre las Aguas Minerales a nivel internacional.
4. Análisis de la base datos dada a Cuba a través del convenio con Brasil.
5. Estudio de las herramientas a utilizar.
6. Análisis y diseño de una aplicación informática que garantice la eficiencia en la gestión de información relacionada con el Balance de las Aguas Minerales.
7. Validación de la propuesta.

Se plantea la siguiente **idea a defender**:

Con el análisis y diseño de una aplicación fácil de utilizar basada en nuevas tecnologías, se contribuirá a perfeccionar el proceso de gestión de datos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales en la ONRM.

Como **métodos de investigación científica** se utilizaron:

Métodos Teóricos:

- ✚ Histórico lógico: Posibilita el análisis histórico del proceso de gestión de datos.
- ✚ Análisis y la síntesis: Se analiza la bibliografía y se realiza síntesis de la misma.
- ✚ Modelación: Se modelan diagramas para el análisis y diseño de la aplicación Web.

Métodos Empíricos:

- ✚ Entrevistas: Se realizan entrevistas al especialista del Balance de Recursos de Aguas Minerales en la ONRM con el fin de precisar la idea a defender.
- ✚ Observación: Se realizaron visitas a la ONRM para observar los procesos manuales que se llevan a cabo en la actualidad.

Estructuración del contenido

La investigación consta de 5 capítulos donde en el **Capítulo 1** se abordará en detalle todo lo relacionado con la fundamentación teórica de la investigación, y se especificarán algunos conceptos asociados a la misma. Se trata de forma detallada el Objeto de Estudio y la Situación Poblémica para mejor entendimiento del propósito de la investigación.

En el **Capítulo 2** se hará referencia a las Tecnologías de la Información y las Comunicaciones (TIC), además de tratar las tendencias y tecnologías actuales para el desarrollo de aplicaciones Web, para de esta forma fundamentar la elección del Lenguaje de programación que se escoja así como de la Metodología de Desarrollo, el Lenguaje de Modelado y la Herramienta CASE.

En el **Capítulo 3** comprende las características del sistema, realizándose la presentación de la solución propuesta.

En el **Capítulo 4** es donde se efectúa la construcción de dicha solución, la cual incluye el diagrama de clases del análisis y del diseño, al igual que el diseño de la base de datos para el sistema propuesto.

En el **Capítulo 5** se maneja el estudio de la factibilidad, la planificación, los costos y los beneficios.

CAPÍTULO 1

FUNDAMENTACIÓN TEÓRICA.

1.1 Introducción

En el presente capítulo se analizan y clasifican los conceptos necesarios para lograr un mejor entendimiento de la investigación del tema, se describen los procesos actuales de la ONRM, quedando claramente expuestos los principales problemas que motivan la investigación. Se analizan soluciones existentes que pudieran contribuir con una posible variante de solución del problema.

1.2 Conceptos asociados al dominio del problema.

1.2.1 Geología

La geología (del griego geo, tierra, y logos, estudio) es la ciencia que estudia la forma interior del globo terrestre, la materia que lo compone, su mecanismo de formación, los cambios o alteraciones que éstas han experimentado desde su origen, y la colocación que tienen en su actual estado.[5]

1.2.1.1 Hidrogeología

La hidrogeología es una rama de las ciencias geológicas que estudia las aguas subterráneas en lo relacionado con su origen, su circulación, sus condicionamientos geológicos, su interacción con los suelos y rocas, su estado (líquido, sólido y gaseoso) y propiedades (físicas, químicas, bacteriológicas y radiactivas) y su captación.[5]

1.2.2 Balance

Estado anual de los parámetros hidrogeológicos preestablecidos cuantitativamente y cualitativamente como caudal y calidad así como comportamiento en el tiempo de las aguas.

1.2.2.1 Recursos

Elemento de la naturaleza que la industria, con su tecnología, es capaz de transformar.

Volumen de agua disponible para la explotación de un acuífero, expresados en unidad de caudal, con carácter dinámico.[5]

1.2.2.2 Recursos Disponibles

Volumen de agua que puede extraerse permanentemente a largo plazo de un acuífero, sin que experimente una reducción de la reserva permanente, expresados en unidad de caudal. Estos recursos dependen fundamentalmente de la alimentación neta del acuífero y se refieren al caudal de escurrimiento subterráneo bajo condiciones de equilibrio natural.[5]

1.2.2.3 Recursos Explotables o de Explotación

Pueden ser parte o la totalidad de los Recursos Disponibles, y constituyen el volumen de agua que se puede captar de un acuífero a largo plazo sin originar alteraciones indeseables en el régimen de las aguas subterráneas, teniendo en cuenta condiciones técnicas y económicas, expresado en unidad de caudal. En la concepción en estos recursos influyen notablemente la obra de captación y el propio efecto de la explotación.[5]

1.2.2.4 Recursos Asegurados o Caudal Seguro

Es el límite del caudal que puede extraerse regular y de forma permanente de un acuífero, sin una disminución riesgosa de las reservas del mismo, de forma tal que su extracción anual no sea mayor que un nivel del agua subterránea, que provoque intrusión de agua no condicionada.[5]

1.2.3 Aguas

El agua es un compuesto formado por dos átomos de hidrógeno y uno de oxígeno. Su fórmula molecular es H₂O.

Se halla en forma líquida en los mares, ríos, lagos y océanos; en forma sólida, nieve o hielo, en los casquetes polares, en las cumbres de las montañas y en los lugares de la Tierra donde la temperatura es inferior a cero grados Celsius; y en forma gaseosa se halla formando parte de la atmósfera terrestre como vapor de agua.[5]

1.2.3.1 Aguas Minerales

Según la NC 93-01-218 de 1995. Agua Mineral, es el agua de origen natural, cuyas características físicas, químicas, microbiológicas, con caudal determinado, estable durante un ciclo hidrológico, cumple los requisitos higiénicos sanitarios establecidos. Las Aguas Minerales según sus usos pueden ser Minerales Naturales (para ingestión), Minerales Medicinales (para aplicación e ingestión), Termales (para aplicación o aprovechamiento energético) o Industriales (para el aprovechamiento de los minerales que la componen).

Los principales atributos de las aguas minerales son; la estabilidad de su calidad y del caudal estimado y la pureza bacteriológica de la fuente. [5]

1.3 Objeto de estudio

El objeto de estudio de la investigación es el Proceso de gestión de datos para el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales que se realiza en al ONRM.

1.3.1 Descripción General

El proceso de gestión de datos para el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales, consistirá en un sitio Web con una base de datos que hará más eficaz y eficiente el cumplimiento de las funciones de la Dirección Técnica de la ONRM, dirigido a la fiscalización, control y protección de las aguas minerales, ya que permitirá conocer en tiempo real el comportamiento del acuífero tanto cuantitativa como cualitativamente. Este proceso aporta además los elementos necesarios para una correcta planificación en la explotación de los yacimientos y para la toma de decisiones.

Para un mejor entendimiento del objeto de estudio se mencionan los objetivos del proceso:

Llevar un correcto control de que todos los datos bases que tributan al balance ya sea los informes de la última fase de investigación geológica (explotación), los de actualizaciones de sus recursos, como los que conforman el balance generados por el concesionarios. Permitir que la oficina guarde el estado de los parámetros hidrogeológicos preestablecidos en dichos documentos en una base de datos bien segura con los diferentes permisos dado a cada usuario para el momento de realizar cualquier consulta.

El cumplimiento de estos objetivos arrojan grandes resultados como:

- El control del comportamiento en tiempo real, durante el período que se informa (1 año) y el acumulado a la fecha de los parámetros hidrogeológicos preestablecidos.
- El control de los recursos que realmente se han extraído del acuífero.
- La perfección del cumplimiento de las funciones de la ONRM relativa a la fiscalización.
- La mejora de la comunicación entre los concesionarios y la Dirección Técnica.
- Más rapidez y precisión en el desarrollo del proceso.

Proporcionando varias ventajas al proceso actual, como:

- Se cuenta con una aplicación de escritorio para la recogida de los datos en el concesionario.
- Se controla el comportamiento en tiempo real, durante el período que se informa (1 año) y el acumulado a la fecha de los parámetros hidrogeológicos preestablecidos.
- Se controla los recursos que realmente se han extraído del acuífero.
- Mayor rapidez y precisión en el desarrollo del proceso (la aplicación de campo generará un XML con la información necesaria dado que el concesionario cuenta con el acceso a la Web, este podrá cargar dicho archivo, donde el especialista en la Dirección Técnica podrá leer, revisar y aprobar los datos enviados por los concesionarios).
- Aumento eficaz de la eficiencia en el desarrollo del proceso.
- Mejora la comunicación entre ambas entidades Concesionario y la ONRM.

De esta forma se puede decir que existen varias desventajas las cuales pueden influir en el desarrollo del proceso:

- El concesionario que no cuente con la tecnología necesaria para este proceso no podrá beneficiarse con este hasta que la haya adquirido.
- La comprobación de la eficiencia del proceso en parte se ve limitada con la conclusión de la aplicación de campo.

1.3.2 Descripción Actual del Dominio del Problema

Dentro de la Oficina Nacional de Recursos Minerales (ONRM), la Dirección Técnica se encarga de revisar y dar los diferentes certificados de aprobación a los concesionarios para su correcto funcionamiento. Al surgimiento de un nuevo yacimiento el concesionario a cargo debe realizar un

“Informe de la Investigación Geológica e Hidrogeológica”. Este es enviado a la Dirección Técnica la cual se encarga de revisar y de dar el “Certificado de Aprobación” en caso que este correcto, de lo contrario emite un “Acta de Corrección”. Luego de aprobado el informe, el concesionario puede presentar o no ,el “Proyecto de Explotación” en dependencia de si desea explotar o no. Este pasa por el mismo proceso de revisión y aprobación.

A lo largo de la vida de explotación, el concesionario debe mantener un control activo sobre los diferentes parámetros que se miden en el pozo. Este registro debe estar disponible en cualquier momento en que la Dirección Técnica desee chequear su funcionamiento. Estos datos son recogidos en una tabla que lleva por nombre “Movimiento o Comportamiento de los Recursos Disponible y de Explotación de Aguas Minerales”. Esta tabla, que tiene que estar disponible en todo momento, también forma parte del balance pues independientemente que se pueda chequear los datos en cualquier momento, dadas las características de la recogidas de los datos en el concesionario, esta debe ir como parte del balance analizando todo el período comprendido. Los otros documentos que forman el balance es la tabla “Resumen General de los Yacimientos de Aguas Minerales”, el documento “Proyecto de Resolución de Balance de Agua” y las diferentes gráficas que muestran el “Comportamiento de los diferentes parámetros hidrogeológicos y variables o elementos del período analizado(1 año)”. El documento integrado por estos anteriormente explicados es “El Balance o Estado Anual de los Recursos Disponibles y de Explotación de las Aguas Minerales de la Nación” y este también pasa por un proceso de revisión y aprobación en la Dirección Técnica.

1.3.3 Situación Problemática

En la ONRM hasta el momento no existe ningún sistema automatizado que gestione los procesos relacionados con las aguas minerales para la gestión del balance o estado anual de las mismas.

Hasta la fecha se lleva el control sobre las aguas minerales de forma manual pero por su importancia se hace necesario comenzar a realizarlo de forma diferente a través de una aplicación que gestione y guarde estos datos.

Los concesionarios (que son las entidades que trabajan ligadas a la ONRM, en el caso Ciego Montero o los Portales por citar un ejemplo) tienen que dirigirse a dicha oficina para entregar la información en copia dura o digital en CD, soportada en diferentes formatos como Word o Excel , los cuales son revisados y aprobados por el especialista de la ONRM.

Al no existir algún software que guarde estos datos existe la deficiencia y lentitud del proceso, así como en la repercusión de no tener constancia de cómo se mantuvo la estabilidad del agua en un período de tiempo mayor a un año puesto que los registros son instantáneos sin un historial. Como el control de las aguas minerales repercute en la toma de decisiones de las diferentes empresas que comercializan con ella dígase Ciego Montero o los Portales, hacen que aumente su urgencia dada su importancia.

Por el interés que existe la gestión de los datos sobre dichas aguas se acordó una nueva metodología para llevar a cabo este proceso. Por lo que no sólo se trabajaría en realizar de forma automática un balance de las aguas minerales sino todo lo relacionado con el control y análisis de las aguas naturales minerales, termales, industriales, entre otras, que enmarca esta metodología.

Se hace evidente, que con la nueva metodología para realizar el balance, se hace necesario un sistema, para evitar la consecuencia de un retraso en el tiempo de respuesta y la eficiencia de los principales servicios que brinda esta dependencia ministerial.

1.4 Análisis de otras soluciones existentes

En el mundo no hay evidencia de que se lleve un control sistemático sobre la información cualitativa y cuantitativa de las aguas minerales que son el objetivo del trabajo. Hay constancia de registros sobre controles realizados a las aguas subterráneas pero no específicamente sobre las aguas minerales que tienen sus exigencias específicas como caudal, calidad y estabilidad.

Después de un análisis de cómo se lleva este proceso en otros países como España y Brasil. Se toman algunos elementos que se puedan adaptar y hacerlos parte del proceso. Hay que tener en cuenta que las empresas relacionadas en estos procesos son privadas por lo que el control de estos datos no se puede llevar de igual forma en un lugar como en otro dado que las características son diferentes.

De España del Instituto Geominero se toma el modelo de captura de información y del CPRM Servicios Geológicos de Brasil por un convenio entre países, el producto SIAGAS para aguas subterráneas. De SIAGAS se toma el sistema gestión de Base Datos y de las empresas la captura de la información.

Esta solución dada a Cuba aunque no satisfacen totalmente las necesidades del cliente ya que no se centra en las especificidades que tiene el proceso de gestión para realizar el balance, es la que se ha puesto en marcha temporalmente.

Esta aplicación surge a raíz de los acuerdos suscritos entre Cuba y Brasil en ocasión de la visita oficial del Presidente Luiz Inácio Lula da Silva en el mes de enero en Cuba.

El acuerdo complementario al acuerdo básico de cooperación científica, técnica y tecnológica entre el gobierno de la República de Cuba y el gobierno de la República Federal de Brasil, para la implementación del proyecto “Soporte Técnico y Extensión Institucional del Sistema de Informatización de Aguas Subterráneas SIAGAS – Cuba.”

Este proyecto tiene como objetivo implementar y desarrollar la asistencia técnica al Sistema de Informaciones de Aguas Subterráneas – SIAGAS – CUBA, incluyendo el soporte técnico para la correcta consistencia y alimentación a la base de datos, y promover el intercambio de experiencias y la actualización para la elaboración de los estudios con vistas al reconocimiento y el control de los recursos de aguas minerales y subterráneas de Cuba, haciéndolo extensivo a otras instituciones del país.

Comprende la realización de evaluaciones anuales, a través de visitas técnicas, de la ejecución del SIAGAS – CUBA, promoviendo el entrenamiento técnico para el uso del sistema, y la concesión de subsidios para la elaboración de inventarios de pozos e implantación del SIAGAS en otras instituciones.[6]

Este producto como otro cualquiera presenta ventajas y desventajas las cuales se mencionan respectivamente:

Ventajas:

- Es un producto que está probado y funciona muy bien para aguas subterráneas.
- Permite la interpretación y el análisis de la información variada sobre aguas subterráneas.

Desventajas:

- No está probada en Cuba.
- No es una salida para el Balance o Estado Anual de las Aguas Minerales.

- La usabilidad en la actualidad de esta Base de Datos es mínima pues en estos momentos está en la fase de cargar los datos de Cuba para una pronta funcionalidad en la Dirección Técnica de la ONRM.

1.5 Conclusiones

En este capítulo se realizó una descripción general del objeto de estudio de la presente investigación. Se enunciaron los problemas y necesidades que presenta la Dirección Técnica de la ONRM, así como los diferentes conceptos con los que se trabaja. De ahí que surja la idea de desarrollar un sistema como el que se propone, con el objetivo de solucionar la situación problemática que genera el problema científico, razón de esta investigación.

Para la solución a los problemas que presenta la informatización de los procesos y para la satisfacción de las necesidades reales que existen, se trazaron los objetivos generales y específicos con los que se da respuesta a estas problemáticas.

Además con el análisis de cada uno de estos aspectos mencionados se permite arribar de manera parcial a las siguientes conclusiones:

- El estudio del proceso manual de gestión de la informatización de datos para el balance, facilitó el aprendizaje de la información necesaria e imprescindible, que será necesaria para el proceso que se quiere implementar.
- El sistema informático existente (SIAGAS) en la Dirección Técnica de la ONRM no es óptimo para el problema que se quiere satisfacer.

CAPÍTULO 2

TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.

2.1. Introducción

En el presente capítulo se describen todas las tecnologías que se utilizarán en la construcción de la aplicación en cuestión, se analizan las principales tendencias actuales en cuanto a tipos de aplicaciones, Metodologías de Desarrollo de Software, Herramientas CASE y de programación, arquitectura en aplicaciones Web, y sistemas gestores de bases de datos. El capítulo incluye la justificación de todos los usos de tecnologías o tendencias a utilizar, evidenciando claramente sus principales características que hacen posible su selección.

2.2. 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.

Estructura de las aplicaciones Web.

Las aplicaciones Web pueden ser estructuradas de diversas formas, todo depende de los desarrolladores de la misma, pero hoy en días las que más se desarrollan son las aplicaciones por capas de todas las más conocidas son aquellas que son desarrolladas en tres capas; las cuales se describen a continuación:

Capa I: Interfaz o Navegador Web

Capa II: Lógica o de Negocio

Capa III: Base de Datos

Entre estas capas existe una gran relación y comunicación. Su funcionamiento consiste en efectuar peticiones desde la capa interfaz o Navegador Web hacia la capa de acceso a datos a través de la capa Lógica o de Negocio, pues la capa de Presentación no puede comunicarse directamente con la de almacenamiento ya que esta arquitectura consiste en aislar la lógica de la aplicación y en convertirla en una capa intermedia bien definida y lógica del software. En la capa de presentación se realiza relativamente poco procesamiento de la aplicación; las ventanas envían a la capa intermedia peticiones de trabajo. Y éste se comunica con la capa de almacenamiento del extremo posterior. [7]

Asumiendo las características y facilidades de un sitio Web, se planea desarrollar una aplicación Web sencilla y de fácil utilización por los usuarios que les facilite con mayor seguridad una gestión de datos relacionados con las informaciones referidas al Balance de Aguas Minerales, permitiéndole a los usuarios una gran eficiencia en las respuestas según sus solicitudes, esto posibilitará tener una base de datos mucho más segura. El software al tener características específicas, como que los usuarios que acceden a ella están ubicados en distintos puntos geográficos, hace que sea más factible el uso de una aplicación Web, dado que si ocurre un cambio sólo se afectaría el servidor donde esté corriendo. Este no es el caso de una aplicación de escritorio producto que debe instalarse en todas las estaciones de trabajo que a su vez tienen que contar con la tecnología que esta exija para su correcto funcionamiento.

2.3. Software Libre

Desde un tiempo atrás hasta la actualidad en Cuba se ha dado un proceso que periódicamente ha ido evolucionando, el proceso de la migración hacia el software libre. Actualmente un gran número de usuarios de la red cubana están migrando para esta nueva tendencia, debido a sus facilidades y comodidades. La ONRM no está ajena a este cambio y también se inclina por aquellos productos que sean libres y que cumplan con las libertades del proyecto GNU (Acrónimo recursivo que significa "GNU no es Unix"):

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a las necesidades.
- La libertad de distribuir copias, con lo que puedes ayudar al vecino.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.

Con la licencia del proyecto GNU GPL se hace obligatorio la palabra “libertad” en los software llamados “libres”, la cual, que quede claro, es un asunto de libertad de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software y no precisamente de precio. GNU posee una serie de software libre los cuales brindan una gran ventaja a la hora de desarrollar un producto. Uno de ellos es el sistema operativo GNU/Linux o Sistema Operativo Linux el cual es muy estable en su rendimiento y se ha vuelto muy eficaz en los últimos años para la comunidad informática cubana y del mundo en general. Por todas las ventajas que posee este sistema operativo y por petición del cliente se hará uso del mismo en el software ha realizar, haciendo uso además de herramientas y tecnologías libres.

2.3.1. Distribución de Linux

Una distribución no es otra cosa, que una recopilación de programas y ficheros, organizados y preparados para su instalación. Estas distribuciones se pueden obtener a través de Internet, o comprando los CDs de las mismas, los cuales contendrán todo lo necesario para instalar un sistema Linux bastante completo y en la mayoría de los casos un programa de instalación que ayudará en la tarea de una primera instalación (Martínez 1998). Existen muchas y variadas distribuciones de GNU/Linux, una de ellas es UBUNTU, la cual será la que se utilizará como SO en el desarrollo del trabajo.

UBUNTU es un sistema operativo de código abierto desarrollado en torno al Kernel Linux. La filosofía UBUNTU se basa en los siguientes principios: que el software debe ser gratuito, que la gente debe poder usar el software en su lengua materna y debe poder hacerlo independientemente de cualquiera sean sus limitaciones; además, la gente debe ser libre de personalizar o modificar el software del modo que crea más conveniente. UBUNTU incluye una cuidadosa selección de los paquetes de Debian, y mantiene su poderoso sistema de gestión de paquetes que nos permite instalar y desinstalar programas de una forma fácil y limpia. Este emplea las mejores herramientas de traducción y accesibilidad que la comunidad del Software Libre es capaz de ofrecer.

2.4. Metodologías de Desarrollo de Software.

Uno de los principales problemas en la actualidad en el desarrollo de software es seleccionar la metodología 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.

No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable. Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo. Los ciclos de vida se clasifican en Cascada, usando Prototipos, Basado en Componentes, en Espiral (Incremental, Iterativo), Programación Automática. Las metodologías se basan en alguna combinación de estos enfoques.

Las metodologías (tanto comerciales como en el ámbito académico y de investigación) pueden ser agrupadas en dos grandes corrientes: Metodologías Estructuradas y Metodologías Orientadas a Objetos.

Actualmente se cuenta con una adecuada cantidad de propuestas que definen este marco de trabajo, se escucha hablar de metodologías tradicionales o robustas como Rational Unified Process (RUP), Microsoft Solutions Framework (MSF) y Métrica 3.0 o metodologías ágiles como Extreme Programming, Scrum, Cristal Methods y Feature Driven Development, pero las experiencias indican que no existe una receta mágica y no se trata de sólo seleccionar una de las propuestas y seguirla simplemente, es seleccionar la correcta para solucionar el problema.

2.4.1 Proceso Unificado de Desarrollo de Software (RUP)

El Proceso Unificado está basado en componentes. Utiliza como estándar de modelado visual el UML y se sostiene sobre tres características fundamentales:[8]

Guiado por CU: Los casos de uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.

Centrado en la Arquitectura: Los modelos proyectados del análisis y el diseño constituyen la arquitectura del producto a desarrollar.

Iterativo e Incremental: Durante todo el proceso de desarrollo se producen versiones incrementales (que se acercan al producto terminado) del producto en desarrollo.

RUP se repite a lo largo de una serie de ciclos de desarrollo que constituyen la vida de un sistema, donde cada ciclo concluye con una versión del producto. Cada ciclo consta de cuatro fases: Inicio, Elaboración, Construcción y Transición. Todas las fases se subdividen en n iteraciones y terminan con un hito.

RUP está dividido en nueve flujos de trabajo, los cuales se pueden ver claramente en la siguiente figura:

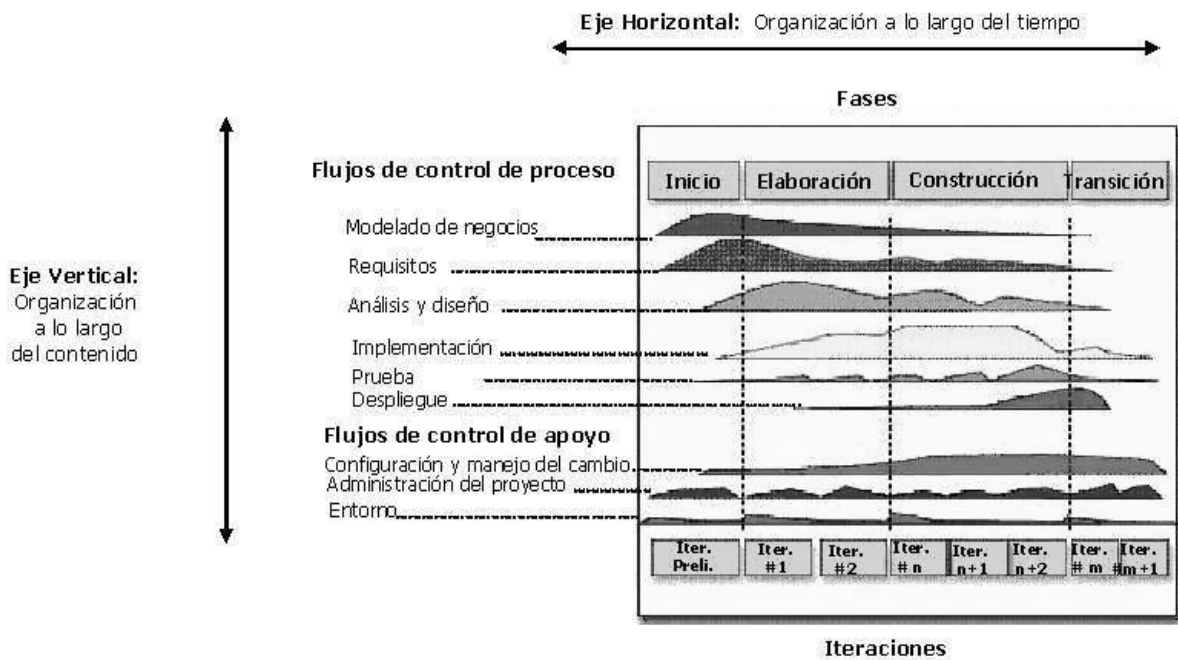


Figura 1 Fases y Flujos de Trabajo de RUP

2.4.2 Programación Extrema (XP)

Completamente diferente al proceso anteriormente mencionado, XP es una metodología ágil, preocupándose principalmente del objetivo, mejorando las relaciones interpersonales y velocidad de reacción, cosa que RUP no hace.

Se intenta minimizar el riesgo de fallar en el proceso usando a un representante del cliente a tiempo completo para el equipo de desarrollo. La función de este es de contestar rápidamente a cualquier duda del equipo y corregir de la misma manera, sin tener que retrasarse en 'absurdas' reuniones de tomas de decisiones. [9]

XP define UserStories como base del software a desarrollar. A partir de los UserStories y de la arquitectura perseguida se crea un plan de release entre el equipo de desarrollo y el cliente. Para cada release se discutirán los objetivos de la misma con el representante del cliente y se definirán las iteraciones (de pocas semanas de duración).[9]

A diferencia de otros métodos, en XP la codificación pertenece al equipo completo, de forma que el conocimiento de la aplicación completa lo posea el equipo entero y no unos pocos miembros. [9]

En XP se sigue un diseño evolutivo con la siguiente premisa: Conseguir la funcionalidad deseada de la forma más sencilla posible. Este diseño hace que apenas se le de importancia al análisis como fase independiente, debido a que se trabaja exclusivamente en función de las necesidades del momento.[9]

XP tiene como objetivo primario el funcionamiento del sistema, y cumplir los planes de releases, para luego trabajar en mejorar los algoritmos. Es decir se ocupa la filosofía KISS (Keep It Simple Stupid), un diseño evolutivo de "conseguir la funcionalidad de la manera más simple posible". Esto permite ahorrar mucho tiempo en la fase de análisis, dándole menos importancia como fase individual, pero integrándola al final de obtener funcionalidad.[9]

2.4.3 Desarrollo guiado por funcionalidad (FDD)

FDD es un proceso diseñado por Peter Coad, Erich Lefebvre y Jeff De Luca. Se basa en un proceso iterativo con iteraciones cortas. Está pensado para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Esta metodología propone tener etapas de cierre cada dos semanas, lo cual implica que los desarrolladores tendrán nuevas actividades que realizar en dicho período de tiempo. Esto hace que la motivación del equipo se mantenga durante todo el proyecto debido a que se ven los resultados periódicamente.[9]

Cada iteración se basa en funcionalidades, que son partes del sistema que tiene significado para el cliente. Esto quiere decir que se evita explicarle al cliente lo que significa construir el sistema de persistencia, para preguntar cosas más concretas, como 'enviar pedido por e-mail'. [9]

Las fases de desarrollo de FDD son:

- Desarrollar un modelo general.
- Construir una lista de funcionalidades.
- Realizar plan de releases en base a las funcionalidades a implementar.
- Diseñar en base a las funcionalidades.
- Implementar en base a las funcionalidades.

Ocupar en las primeras tres fases gran parte del tiempo, al inicio del proyecto, pero a medida que se avanza en las iteraciones las otras dos van ocupando más tiempo, y las primeras sólo son para el

refinamiento del release siguiente.

A diferencia de XP, siempre hay un responsable último con mayor experiencia, que tendrá la última palabra en el caso de problemas sin resolver o sin acuerdo, lo cual además permite asignar responsabilidades que todas las empresas exigen, respetando la jerarquía. Ahora, al igual que XP, en FDD se trabaja como grupo, por lo que los menos inexpertos se ven beneficiados de la experiencia de otros. [9]

2.4.4 Selección de la metodología a utilizar

Después de realizar un estudio de las tres metodologías más usadas actualmente en el mundo, se llega a la conclusión, dada las características de este Trabajo Investigativo que está fuertemente vinculado con el Programa Nacional de Informatización del Conocimiento Geológico (PNICG), que la metodología a utilizar para el desarrollo de la aplicación Web propuesta es RUP. Esta presenta características muy beneficiosas permitiendo un mejor control, entendimiento y seguimiento de todos los artefactos generados, así como la documentación que exige, ganando en organización y claridad para el desarrollo de un mejor trabajo en equipo, siendo la más completa y abarcadora en comparación con las antes mencionadas, como señalan algunos autores, las otras metodologías son casos particulares de esta; además se ajusta perfectamente a las características del proyecto pues este es de gran tamaño y larga duración, con un personal de trabajo inestable y cambiante debido a que son estudiantes y profesores por lo que se necesita suficiente documentación para la continuidad del mismo sin que sea afectado por los factores anteriormente mencionados.

2.5. El Lenguaje Unificado de Modelado (UML) como soporte de lenguaje orientado a objeto para el modelado de aplicaciones Web.

En todas las disciplinas de la ingeniería se hace evidente la importancia de los modelos ya que describen el aspecto y la conducta de “algo”. Ese “algo”, puede existir, estar en un estado de desarrollo o estar, todavía, en un estado de planeación.

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 COBOL10, 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.[10]

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. [7]

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. [11]

El Lenguaje Unificado de Modelado (UML) ha ganado en utilización actualmente, por ser la mezcla eficiente de una gran cantidad de estándares internacionales. Su base esta soportada por tres metodologías procedentes de la unión de tres grandes creadores, James. Rumbaugh, Grady. Boosh e Ivar. Jacobson. Logrando así un lenguaje de excelencia para modelar, que es el procesamiento que realizan los ingenieros para el diseño de software previo a su construcción. [8]

Tiene una gran cantidad de propiedades que han sido las que, realmente, ha contribuido a hacer de UML el estándar de la industria en la actualidad.

Algunas de las propiedades de UML como lenguaje de modelado son:[8]

- 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.

El modelar sirve, no solamente para los grandes sistemas, sino en aplicaciones de pequeño tamaño se obtienen beneficios del modelado, sin embargo es un hecho que entre más grande y complejo es el

sistema, más importante es el papel que juega el modelado por una simple razón: "El hombre hace modelos de sistemas complejos porque no puede entenderlos en su totalidad".[11]

Se utilizará como notación el Lenguaje Unificado de Modelado para lograr un mayor entendimiento ya que se logra modelar y describir secuencialmente por pasos todos los procesos que se lleva a cabo según la problemática planteada. Sirve porque es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

2.6. Herramientas CASE de Desarrollo de Software

Las herramientas CASE (Computer Aided Software Engineering), alcanzaron su techo a principios de los años 90. En la época en la que IBM había conseguido una alianza con la empresa de software AD/Cycle para trabajar con sus mainframes, estos dos gigantes trabajaban con herramientas CASE que abarcaban todo el ciclo de vida del software.

CASE es la Ingeniería de Software Asistida por Ordenador son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

Existen diversas herramientas CASE, de estas se pueden mencionar el Rational Rose y el Visual Paradigm.

2.6.1. Rational Rose

Rational Rose Enterprise es el producto más completo de la familia Rational Rose. Siendo la mejor elección para el ambiente de modelado que soporte la generación de código a partir de modelos en Ada, ANSI C++, C++, CORBA, Java™/J2EE™, Visual C++® y Visual Basic®. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente.

Rational Rose posee una serie de características fundamentales:

- Capacidad de Análisis de calidad de código.
- Característica de control por separado de componentes modelo que permite una administración más granular y el uso de modelos.
 - Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
 - Integración con otras herramientas de desarrollo de Rational.
 - Capacidad para integrarse con cualquier sistema de control de versiones SCC-compliant, incluyendo a Rational ClearCase. [12]

Destacar que permite a los arquitectos y diseñadores producir modelos independientes de lenguaje en Unified Modeling Language (UML) de arquitectura de software, necesidades de negocios, activos re-utilizables y comunicaciones en nivel de gerencia, posee una experiencia de diseño única, su desarrollo es dirigido por modelo con soporte de UML, permite el soporte a múltiples modelos para Model-Driven Architecture (MDA), se ejecuta tanto stand-alone o integrado a Microsoft® Visual Studio .NET. Crea arquitecturas independientes de plataforma que se pueden implementar en plataformas Java™ y .NET. Sus estándares son definibles para usuarios para crear, personalizar y apalancar estándares de diseño arquitectónico. Las referencias de modelo cruzado y de realización de versiones hasta el nivel de diagrama y clase permiten estructuración para adecuarse a cualquier proyecto. Mantiene la posibilidad de rastreo entre modelos de análisis, diseño e implementación. La modelación es de forma libre y permite publicación e informes en Web.[12]

2.6.2. Visual Paradigm

Visual Paradigm es una herramienta CASE que utiliza UML como lenguaje de modelado. Está diseñada para una amplia gama de usuarios interesados en construir sistemas de software fiables con el uso del paradigma orientado a objetos, incluyendo actividades como ingeniería de software, análisis de sistemas y análisis de negocios.[13]

Visual Paradigm es una herramienta fácil de utilizar que emplea las últimas notaciones de UML, ingeniería inversa, generación del código, importación de Rational Rose, exportación/importación XML, generador de impresos, integración del ms Visio, IDE Integration de UML con Eclipse, NetBeans.

Por lo que Visual Paradigm ofrece un entorno de creación de diagramas para UML. Con un diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad. Usa un lenguaje estándar, común a todo el equipo de desarrollo que facilita la comunicación. [14]

2.6.3. Selección de la Herramienta CASE a utilizar

Estas dos herramientas CASE son muy valiosas y ambas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software pero hay que seleccionar cual es la más apropiada según las características del proyecto al cual esta vinculado este trabajo de diploma., llegando a la conclusión de que la más idónea es Visual Paradigm debido a todas las facilidades que ofrece:

- Entorno de creación de diagramas para UML 2.0.
- Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad en múltiples plataformas.
- Software con licencia gratuita y comercial.

Siendo esta última característica un factor principal al realizar la selección, producto que el Visual Paradigm es una herramienta libre y el Rational es propietario. Esta es una característica negativa pues Cuba no tiene permitido usar software de origen norteamericano. Esto forma parte de las restricciones del bloqueo económico al que esta sometido el país.

2.7. Arquitectura en Capas

Con el transcurso del tiempo las aplicaciones de n-tier (n-capas) han emergido como la arquitectura predominante para la construcción de aplicaciones multiplataforma en la mayor parte de las empresas. En este tipo de arquitectura se definen un conjunto de niveles o capas, cada nivel interno que se atraviesa se aproxima más al nivel del conjunto de instrucciones de máquina, cada capa sólo puede comunicarse con las vecinas.

Como tecnología, las arquitecturas de n-capas proporcionan una gran cantidad de beneficios para las empresas que necesitan soluciones flexibles y fiables para resolver complejos problemas inmersos en cambios constantes.

Todas las aplicaciones basadas en n-capas permitirán trabajar con clientes ligeros, tal como navegadores de Internet, WebTV, Teléfonos Inteligentes, PDAs (Personal Digital Assistants o Asistentes Personales Digitales) y muchos otros dispositivos preparados para conectarse a Internet.

2.7.1 Arquitectura en 3 capas.

Una aplicación de tres capas es una aplicación cuya funcionalidad puede ser segmentada en tres niveles lógicos (capas):

- Los servicios de presentación.
- Los servicios de negocios (Lógica de Negocios).
- Los servicios de datos.

La capa de servicios de presentación es responsable de:

- Obtener información del usuario.
- Enviar la información del usuario a los servicios de negocios para su procesamiento.
- Recibir los resultados del procesamiento de los servicios de negocios.
- Presentar estos resultados al usuario.

El nivel de servicios de negocios es responsable de:

- Recibir la entrada del nivel de presentación.
- Interactuar con los servicios de datos para ejecutar las operaciones de negocios para los que la aplicación fue diseñada a automatizar (por ejemplo, la preparación de impuestos por ingresos, el procesamiento de ordenes y así sucesivamente).
- Enviar el resultado procesado al nivel de presentación.

El nivel de servicios de datos es responsable de:

- Almacenar los datos.
- Recuperar los datos.
- Mantener los datos.
- La integridad de los datos.

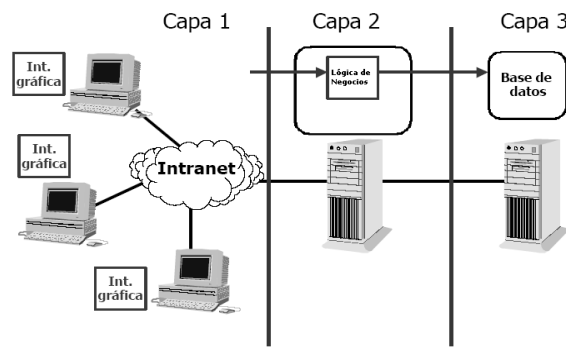


Figura 2 Arquitectura 3 capas

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

- Soportar un diseño basado en niveles de abstracción crecientes.
- Permitir a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
- Admitir optimizaciones y refinamientos.
- Proporcionar amplia reutilización. Al igual que los tipos de datos abstractos.
- 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.

2.8. Sistema Gestor de Base de Datos

Un Sistema Gestor o Manejador de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una BD, por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. Pueden ser de propósito general o específico.[15]

Una BD es un conjunto de datos que tiene las siguientes propiedades implícitas:

- Representa algún aspecto del mundo real, llamado minimundo o universo de discurso. Las modificaciones del minimundo se reflejan en la BD.
- Es un conjunto de datos lógicamente coherentes, con un cierto significado inherente. Una colección aleatoria de datos no puede considerarse propiamente una BD.

Es decir que una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), que ofrece un amplio rango de soluciones al problema de almacenar datos.

Una BD se diseña, construye y puebla con datos para propósito específico. Está dirigida a un grupo de usuarios y tiene ciertas aplicaciones preconcebidas que interesan a distintos usuarios. [15]

2.8.1. MySQL

Es un sistema de administración de base de datos, multi - hilo, multiusuario, con más de seis millones de instalaciones por todo el planeta. MySQL es propietario y patrocinado a partir de la versión 5 por la compañía Sun Microsystems¹ que lo adquirió recientemente.

MySQL es muy popular en aplicaciones Web y actúa como un componente de bases de datos para las plataformas LAMP², MAMP y WAMP (Linux/MAC/Windows-Apache-MySQL-PHP/Perl/Chitón) y para herramientas de búsqueda e identificación de errores.

MySQL trabaja en numerosas plataformas como AIX³, GNU/Linux, Mac OS X y todas las versiones de Windows. Su mayor desempeño se logra cuando se combina con el lenguaje de programación PHP.

2.8.2. Postgre.

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde 1977. Comenzó como un proyecto denominado Ingres en la

¹ Es una empresa informática de Silicon Valley, fabricante de semiconductores y software. Las siglas SUN se derivan de «Stanford University Network», proyecto que se había creado para interconectar en red las bibliotecas de la Universidad de Stanford.

² El acrónimo LAMP, MAMP y WAMP se refiere a un conjunto de subsistemas software necesario para alcanzar una solución global, en este caso configurar sitios web o Servidores dinámicos con un esfuerzo reducido.

³ AIX (*Advanced Interactive eXecutive*) es un sistema operativo UNIX System V propietario de IBM. Inicialmente significaba "Advanced IBM Unix" pero probablemente el nombre no fue aprobado por el departamento legal y fue cambiado a "Advanced Interactive eXecutive"

Universidad Berkeley de California. Ingres fue más tarde desarrollado comercialmente por la Relational Technologies/Ingres Corporation.⁴

Como muchos otros proyectos Open Source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional.[16]

Entre las características más importantes de este gestor se encuentra:

- 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 arrays.⁵
- Altamente Extensible. PostgreSQL soporta operadores, funciona con 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.
- API Flexible. La flexibilidad del API⁶ de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, Java/JDBC y C/C++.
- 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. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

⁴ Compañía que brinda soluciones empresariales de código abierto para la gestión de la información.

⁵ Un array es un medio de guardar un conjunto de objetos de la misma clase.

⁶ Una API (Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

- MVCC. MVCC, o Control de Concurrencia Multi-Versión (Multi-Versión Concurrency Control), es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios. Si alguna vez ha usado algún DBMS con capacidades SQL, tal como MySQL o Access, probablemente habrá notado que hay ocasiones en las que una lectura tiene que esperar para acceder a información de la base de datos. La espera está provocada por usuarios que están escribiendo en la base de datos. Resumiendo, el lector está bloqueado por los escritores que están actualizando registros.
- Write Ahead Logging (WAL). La característica de PostgreSQL conocida como Write Ahead Logging incrementa la dependencia de la base de datos al registro de cambios antes de que estos sean escritos en la base de datos. Esto garantiza que en el hipotético caso de que la base de datos se caiga, existirá un registro de las transacciones a partir del cual se podrá restaurar la base de datos. Esto puede ser enormemente beneficioso en el caso de caída, ya que cualesquiera cambios que no fueron escritos en la base de datos pueden ser recuperados usando el dato que fue previamente registrado. Una vez el sistema ha quedado restaurado, un usuario puede continuar trabajando desde el punto en que lo dejó cuando cayó la base de datos.[13]

Cuando se comparan los Gestores de Base de Datos, MySQL y Postgre se puede llegar a la conclusión que por sus características anteriormente mencionadas Postgre es altamente confiable para el almacenamiento de datos y con gran integridad relacional, además de ser libre diferencia notable con MySQL el cual no cumple con esta condición.

2.9. Lenguajes de programación.

Un lenguaje de programación es una construcción mental del ser humano para expresar programas. Está constituido por un grupo de reglas gramaticales, un grupo de símbolos utilizables, un grupo de términos monosémicos (es decir, con sentido único) y una regla principal que resume las demás. Para que ésta construcción mental sea operable en un computador debe existir otro programa que controle la validez o no de lo escrito. A éste se le llama traductor.

Los lenguajes pueden ser de alto o bajo nivel. En los de bajo nivel las instrucciones son simples y cercanas al funcionamiento de la máquina, como por ejemplo el código máquina y el ensamblador. En

los lenguajes de alto nivel hay un alto grado de abstracción y el lenguaje es más próximo a los humanos, como por ejemplo L^{éxico}⁷, PASCAL, Cobol o Java. [17]

2.9.1. Lenguajes de programación del lado del cliente. Java Script

Java Script es el lenguaje de programación del lado del cliente desarrollado por Netscape⁸ para incrementar las funcionalidades del lenguaje HTML muy utilizado por ser compatible con la mayoría de los navegadores modernos. Es un lenguaje que no esta orientado a objetos, porque no dispone de Herencia por lo que es correcto decir que es un lenguaje basado en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad, pero si es un lenguaje interpretado, ya que no requiere compilación, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C.

Con Java Script se pueden construir aplicaciones que varían desde la gestión de la información corporativa interna y su publicación en Intranets hasta la gestión masiva de transacciones de comercio electrónico.

Java Script es muy fácil de aprender para quien ya conoce lenguajes similares como el C++ o Java, pero, dada su simplicidad sintáctica y su manejabilidad, no es tampoco difícil para quien se acerca por primera vez a este lenguaje. Sin embargo, esto puede ser un arme de doble filo porque la simplicidad se basa en una disponibilidad de objetos limitada, por lo que algunos procedimientos, aparentemente muy sencillos, requieren *script* bastante complejos.[18]

2.9.2. Lenguajes de programación del lado del cliente. Ajax

Ajax no es una tecnología. Es realmente muchas tecnologías, cada una floreciendo por su propio mérito, uniéndose en poderosas nuevas formas. AJAX incorpora:

- Presentación basada en estándares usando XHTML⁹ y CSS¹⁰;
- Exhibición e interacción dinámicas usando el Document Object Model;
- Intercambio y manipulación de datos usando XML¹¹ and XSLT¹²;

⁷ Es un lenguaje didáctico en español para facilitar el aprendizaje y la enseñanza de la programación orientada a objetos.

⁸ Navegador Web.

⁹ Lenguaje extensible de marcado de hipertexto, es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas Web.

¹⁰ Son las hojas de estilo en cascada, un lenguaje formal usado para definir la presentación de un documento estructurado.

- Recuperación de datos asincrónica usando XMLHttpRequest; y
- JavaScript escribiendo todo junto.[19]

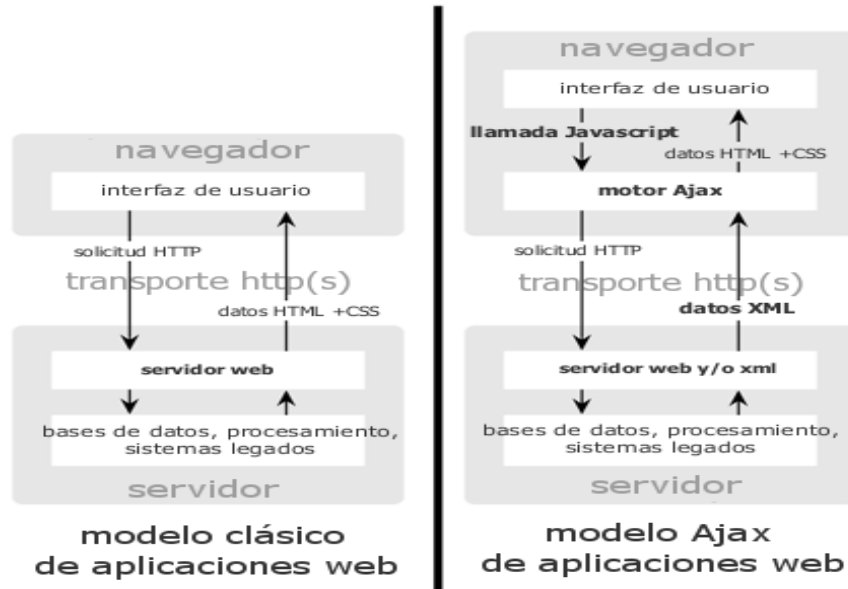


Figura 3 Comparación Modelo Clásico de una aplicación Web y Modelo Ajax de una aplicación Web.

Ventajas del Uso de Ajax:

- **Basado en los estándares abiertos:** Está formado por las tecnologías Javascript, HTML, XML y CSS, soportados por los navegadores más utilizados de Internet como son los basados en Mozilla, Internet Explorer, Safari y Opera.
- **Usabilidad:** Permite a las páginas hacer una pequeña petición de datos al servidor y recibirla sin necesidad de cargar la página entera. El incremento de este tipo de actualizaciones elimina el tener que refrescar el navegador, algo bastante apreciado a la hora de operar en una aplicación Web.
- **Válido en cualquier plataforma y navegador:** Internet Explorer, los basados en Mozilla y Firefox son los que se llevan las palmas en el mercado de Internet y además son los navegadores en los que es más fácil programar aplicaciones Web Ajax, pero ahora es posible construir aplicaciones Web basadas en Ajax para que funcionen en los navegadores más modernos.

¹¹ Es un Lenguaje de Etiquetado Extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos.

¹² XSLT o Transformaciones XSL es un estándar de la organización W3C que presenta una forma de transformar documentos XML en otros.

- Es independiente del tipo de tecnología de servidor que se utilice: Así como Ajax funciona en cualquier navegador, es perfectamente compatible con cualquier tipo de servidor estándar y lenguaje de programación Web. PHP. El ser completamente compatible el desarrollo en éstas tecnologías ha ayudado a Ajax a que vaya cada vez más en incremento.
- Mejora la estética de la Web: Con Ajax se puede interactuar la imaginación del desarrollador con la usabilidad de una aplicación Web de forma que se pueda realizar una aplicación que si no estuviera dentro de un navegador, podría pasar por una aplicación normal de escritorio. [20]

De forma general se puede decir que la utilización de Ajax en el desarrollo de aplicaciones Web es de gran beneficio pues esta es una agrupación de varias tecnologías lo cual propicia el desarrollo más eficiente y fácil de aplicaciones Web con una interfaz de usuario más amigable y funcional.

2.9.3. Lenguajes de programación del lado del servidor. ASP.NET

Desde hace algún tiempo, Microsoft está llevando adelante una estrategia para construir una nueva tecnología tendiente a crear aplicaciones Web distribuidas y que aprovechen al máximo las posibilidades que ofrece Internet. Esta tecnología, que lleva el nombre de .NET, y que incluye un nuevo lenguaje denominado C#, una nueva versión de Visual Basic, con el nombre de Visual Basic.Net y otra serie de tecnologías, entre las que se encuentra: ASP.NET, que viene a reemplazar a las Active Server Pages (ASP), logrando el desarrollo de aplicaciones Web más dinámicas, con un código más claro y limpio, por ende reusable, multiplataforma y definitivamente más simple, ya que el entorno ASP.NET permite la creación automática de alguna de las tarea más comunes para un creador Web, cómo los formularios o la validación de los datos.

Entre las características más importante se encuentran:

- **Eficiencia.**

Desde el principio, uno de los objetivos más importantes del diseño de .NET ha sido su gran rendimiento y nivelación. Para que .NET tenga éxito, las empresas deben estar capacitadas para migrar sus aplicaciones y no sufrir de un rendimiento deficiente debido a la forma en que CLR¹³ ejecuta el código. Para asegurarse un óptimo rendimiento, el CLR compila, en algún punto, todos los códigos de aplicaciones en códigos naturales de máquina.

¹³ Lenguaje común en tiempo de ejecución, es el componente de máquina virtual del .NET Framework de Microsoft.

Esta conversión puede hacerse, o bien en el momento en que se ejecuta la aplicación (método por método), o cuando se instala la aplicación por primera vez. El proceso de compilación hará uso automáticamente de todas las características del microprocesador, disponibles en diferentes plataformas, algo que las aplicaciones tradicionales de Windows nunca podrían hacer, a menos que se cargara distintos binarios para distintas plataformas.

- **Soporte de Lenguajes.**

Esta es una de las novedades más importantes que vienen de la mano de ASP.NET. La posibilidad de escribir código en diferentes lenguajes es un alivio para los desarrolladores que en numerosas ocasiones, veían acotadas sus aplicaciones Web, al estar obligados a trabajar con VBScript o JScript. ASP.NET soporta la programación en lenguajes potentes como, VisualBasic.Net (VB) y C#, el nuevo lenguaje creado por Microsoft con la intención de aprovechar la potencia del C++ y combinarlo con las facilidades que brinda a la programación en Internet un lenguaje como Java.

- **Contenido y Código, por separado.**

Muchos desarrolladores de sitios Web han tenido que lidiar con el inconveniente de tener que crear la interfaz de usuario y el código ASP todo junto. Esta mezcla de imágenes, botones y tablas en código HTML con pedazos de código en VBScript o JScript llegaba a ser algo muy molesto para el desarrollador. ASP.NET viene a solucionar este problema, utilizando un criterio similar al que utiliza Visual Basic, es decir, separar la interfaz de usuario con el código.

- **Código Compilado**

ASP.NET ya no interpreta el código como la hace la versión anterior de ASP. Dentro del entorno NGWS (New Generation Windows Services) el código es compilado just-in-time, logrando un enorme aumento en el rendimiento, a través de soporte nativo y servicios de caché. Uno de los aspectos más importantes dentro del .NET Framework es su librería de clases.[21]

2.9.4. Lenguajes de programación del lado del servidor. PHP

¿Qué es PHP? PHP es el acrónimo de Hypertext Preprocessor (Preprocesador de Hipertexto). Se trata de un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Está muy orientado al desarrollo de aplicaciones Web y permite insertar contenidos dinámicos en las páginas.[22]

Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas Web dinámicas:

- Soporte para una gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server.
- Integración con varias bibliotecas externas, que permite generar documentos en PDF (documentos de Acrobat Reader) hasta analizar código XML.
- Ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación.
- Perceptiblemente más fácil de mantener y poner al día que el código desarrollado en otros lenguajes.
- 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 reparen rápidamente.
- El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.
- Con PHP se puede hacer cualquier cosa con un script CGI¹⁴, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. [23]
- PHP es un lenguaje script procesado en el lado del servidor. El PHP se muestra como código embebido dentro de una página HTML.

Las principales características de PHP son: su rapidez; su facilidad de aprendizaje; su soporte multiplataforma tanto de diversos Sistemas Operativos, como servidores HTTP y de bases de datos; y el hecho de que se distribuye de forma gratuita bajo una licencia abierta, características que lo hacen sobresalir en la elección.

2.10. IDE de desarrollo

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por si solas o pueden ser parte de aplicaciones existentes. Establecen marcos de trabajo amigable, lo que hace posible que un mismo IDE pueda funcionar con

¹⁴ Interfaz de entrada común es una importante tecnología de la World Wide Web (WWW) que permite a un cliente solicitar datos de un programa ejecutado en un servidor web.

varios lenguajes de programación. Este es el caso de Eclipse, que mediante pluggins se le puede añadir soporte de lenguajes adicionales.

2.10.1. Eclipse

Eclipse es una plataforma universal para integrar herramientas de desarrollo, con una arquitectura abierta y basada en pluggins¹⁵. Eclipse da soporte a todo tipo de proyectos que abarcan desde el ciclo de vida del desarrollo de aplicaciones, incluyendo soporte para modelado.

El entorno integrado de desarrollo (IDE) de Eclipse emplea módulos (en inglés plug-in) para proporcionar toda su funcionalidad al frente de la plataforma de cliente, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no. Este mecanismo de módulos es una plataforma ligera para componentes de software, adicionalmente a permitirle a Eclipse extenderse usando otros lenguajes de programación como son C/C++ y Phyton, permite a Eclipse trabajar con lenguajes para procesado de texto como LaTeX, aplicaciones en red como Telnet y Sistema de gestión de base de datos. La arquitectura pluggins permite escribir cualquier extensión deseada en el ambiente, como sería Gestión de la configuración.

Presenta características principales como:

- Editor visual con sintaxis coloreada
- Compilación incremental de código
- Modifica e inspecciona valores de variables
- Avisa de los errores cometidos mediante una ventana secundaria
- Depura código que resida en una máquina remota.[24]

En cuanto a las aplicaciones clientes, Eclipse provee al programador con frameworks muy ricos para el desarrollo de aplicaciones gráficas, definición y manipulación de modelos de software, aplicaciones Web, etc. Por ejemplo, GEF (Graphic Editing Framework - Framework para la edición gráfica).

¹⁵ Un plugin o *componente enchufable* (o plug-in -en inglés "enchufar"-, también conocido como addin, add-in, addon o add-on) es una aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica.

2.10.2. Zend Studio

Son muchos los desarrolladores que trabajan con Zend Studio, es posiblemente uno de los mejores IDE del momento. Se trata de un programa de la casa Zend, uno de los mayores impulsores de PHP, orientada a desarrollar aplicaciones Web, en PHP.

Zend Studio es un editor de texto para páginas PHP que proporciona un buen número de ayudas desde la creación y gestión de proyectos hasta la depuración del código. [25]

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts. [26]

Presenta características como:

Entorno de Desarrollo Poderoso para PHP

- Integración del uso y completado de código personalizado de Zend Framework y vista de la lista de las funciones del framework desde la Visualización de Funciones PHP.
- Visualización de eventos de Zend Platform en una ventana de lista de eventos personalizada y dedicada.
- Aumenta la productividad con: Soporte PHP 5 completo, Analizador de Código, carpeta de Código, completado de Código, coloreado de Sintaxis, Administrador de Proyecto, Editor de Código, Depurador de gráficos y asistentes.
- Documentación del código de forma más sencilla, aplicaciones, y proyectos con PHPDocumentor, la herramienta de documentación standard para PHP.
- Simplifica el despliegue con la integración FTP¹⁶ y SFTP¹⁷ de forma tal que permita a los programadores en forma segura subir y descargar archivos de proyectos de modo transparente hacia y desde servidores remotos.

¹⁶ FTP (File Transfer Protocol) es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.

Desarrollo de Aplicaciones de Negocios Superiores

- Permite conectarse directamente con bases de datos profesionales tales como MySQL, Oracle, Microsoft SQL Server, PostgreSQL.
- Permite escribir y realizar consultas a servidores conectados usando el editor de consultas SQL de Zend con SQL92 y soporte de coloreado de Sintaxis.
- Visualiza las estructuras de la base de datos y administra el contenido con el explorador SQL de Zend.

Entorno de Depuración Completo para PHP

- Características de depuración avanzadas, incluyendo: condiciones límites, visualización de errores, vistas avanzadas, variables y buffer de salida.
- Depuración en forma local y remota en un medio conocido utilizando el depurador PHP más poderoso.
- Asegurar la protección máxima de ubicaciones de proyectos o en Internet con depuradores remotos seguros.
- Depurar y analizar su código directamente desde el nuevo Navegador IE con “un click en el depurador de navegador”.^[27]

2.10.3. Selección del IDE de Desarrollo a utilizar.

- Luego de analizar estos dos potentes IDEs se llega a la conclusión de utilizar el Eclipse, el cual presenta características muy favorables para el desarrollo de aplicaciones.
 - La arquitectura de pluggins permite integrar diversos lenguajes sobre un mismo IDE e introducir otras aplicaciones accesorias.
 - Conservan el registro de las versiones, generan y mantienen la documentación de cada etapa del proyecto.

¹⁷ Es una forma segura de transferir ficheros a un servidor ya que los datos circulan encriptados por la red.

Además de destacar que es un IDE de desarrollo de libre acceso, multiplataforma y multiproyecto.

2.11. Framework.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

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.

Existen varios tipos de frameworks Web: orientados a la interfaz de usuario, como Java Server Faces, orientados a aplicaciones de publicación de documentos, como Cocoon¹⁸, orientados a la parte de control de eventos, como Struts¹⁹ y algunos que incluyen varios elementos como Tapestry.²⁰

La mayoría de frameworks Web se encargan de ofrecer una capa de controladores de acuerdo con el patrón MVC, ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación.[20]

2.11.1. Symfony

Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones Web. Symfony está basado en un patrón clásico del diseño Web conocido como arquitectura Modelo Vista Controlador (MVC), que está formado por tres niveles; 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. 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.

¹⁸ Es un framework para el desarrollo de aplicaciones Web modulares que aísla los distintos aspectos involucrados en la construcción.

¹⁹ Framework de desarrollo.

²⁰ Tapestry es un framework de código abierto para la creación de aplicaciones web de forma dinámica, robusta y altamente escalable en Java.

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales arrojando resultados significativos. 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 *nix (Unix, Linux, etc.) como en plataformas Windows.

A continuación se muestran algunas de sus características.

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y *nix estándares)
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de los 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 *mejores prácticas* y patrones de diseño para la Web.
- Preparado para aplicaciones empresariales y es adaptable 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.[28]

2.12. Conclusiones

En este capítulo se definieron las principales tecnologías que se utilizarán en el desarrollo de la aplicación, se justificó la elección de la utilización de aplicaciones Web, de la herramienta CASE Visual Paradigm acompañada del Lenguaje de Modelado UML, del lenguaje PHP con el framework: Symfony, del Gestor de Bases de Datos Postgre, y por último de la tecnología AJAX. Un factor clave en la selección de cada una de las tecnologías fue la correspondencia con los requisitos establecidos por el cliente y con la característica de Cuba, desarrollar software con tecnología libre.

CAPÍTULO 3

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.1. Introducción

En este capítulo se describe la propuesta de solución planteada, se realiza un análisis de los procesos que se llevan a cabo en la ONRM para el Balance Nacional de Recursos Disponibles y en Explotación, lo que facilita la modelación del negocio presentado en este capítulo, luego se analizan las reglas que este propone, se identifican los actores, trabajadores, casos de uso y la relación entre ellos así como la descripción de cada caso de uso. También se analiza el funcionamiento del sistema enumerando los requisitos funcionales y no funcionales con los que debe cumplir el sistema teniendo en cuenta las necesidades de la ONRM. Por último se identifican los casos de uso del sistema y se relacionan con los actores involucrados.

3.2. Modelo de Negocio

Para comprender mejor todo el proceso de negocio relacionado con el Balance Nacional de Recursos Disponibles y en Explotación se hará uso de las técnicas de modelado que presenta UML, utilizando como herramienta CASE Visual Paradigm, siendo la obtención de estos procesos de negocio un paso fundamental para el modelado del mismo. Se propone desarrollar Modelo de Negocio debido a que se tienen bien claras las fronteras del este, así como las personas involucradas que tienen una responsabilidad bien definida en cada actividad dentro de los procesos del negocio.

3.2.1. Reglas del negocio

- Al surgimiento de un nuevo yacimiento se debe solicitar la concesión y surge un nuevo Concesionario, este debe entregar a la Dirección Técnica los datos primarios.
- Al surgimiento de un nuevo yacimiento que será explotado por un Concesionario existente, este debe de entregar los datos primarios correspondientes a la Dirección Técnica.
- No se considera un yacimiento en explotación sin antes haber entregado a la Dirección Técnica los datos primarios y haber obtenido los Certificados de Aprobación correspondientes.
- Sólo se puede entregar el Proyecto de Explotación a la Dirección Técnica una vez certificado el

Informe de la Investigación Geológica e Hidrogeológica por la misma.

- En caso de la explotación de Aguas Minerales Naturales se tiene que entregar el Proyecto de Explotación y Procesamiento a la Dirección Técnica.
- El Balance o Estado Anual de los Recursos Disponibles y de Explotación de las Aguas Minerales de la Nación debe entregarse anualmente a la Dirección Técnica.
- Todos los documentos entregados por el Concesionario a la Dirección Técnica deben ser revisados y aprobados con una Certificación de Aprobación y en caso de no ser aprobado la Dirección Técnica debe emitir un Acta de Corrección.
- Los usuarios externos sólo tendrán acceso a informaciones generales sobre los Recursos y Reservas de Aguas Minerales.

3.2.2. Actores del negocio

Los actores del negocio están concebidos como el individuo, grupo, organización, máquina o sistema que interactúa y se beneficia con la existencia del negocio, los cuales deben identificarse en la primera etapa que describe RUP, para de esta forma lograr un mayor entendimiento del proceso. A continuación se definen los actores que se identificaron:

Actor	Descripción
Concesionarios.	Son todas las entidades, empresas relacionadas con la ONRM que comercializan o utilizan las Aguas Minerales.
Usuarios	Son todas aquellas entidades o personas que en algún momento pueden solicitar información sobre el Balance.

Tabla 1 Actores del Negocio.

3.2.3. Trabajadores del negocio

Los trabajadores del negocio son aquellas personas o sistemas que están involucrados en uno o más procesos del negocio, que participan en ellos, pero no obtienen ningún resultado de valor.

Trabajador	Descripción
Especialista de la DT.	Es el responsable de revisar y emitir el acta de corrección en caso de errores y el certificado de aprobación en el caso de que los documentos recibidos de los diferentes Concesionarios estén correctos.
Documentador	Es el intermediario entre el Concesionario y la Dirección Técnica, se encarga de recoger y entregar los documentos.

Tabla 2 Trabajadores del Negocio.

3.2.4 Diagrama de Casos de Uso del Negocio

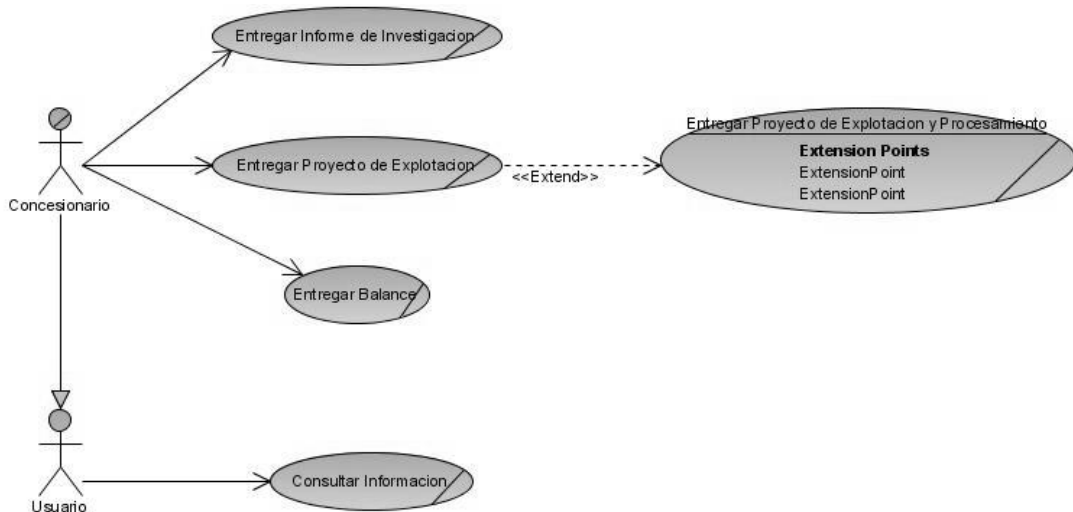


Figura 4 Diagrama de Casos de Uso del Negocio.

3.2.5. Descripción de los Casos de Uso del Negocio

3.2.5.1. Descripción CU_Entregar Informe de Investigación.

Caso de Uso:	Entregar Informe de Investigación
Actores:	Concesionarios.
Resumen:	El Concesionario entrega un informe de la investigación geológica e hidrogeológica a la Dirección Técnica cada vez que surja un nuevo yacimiento.

Precondiciones:	Ya debe existir un concesionario.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Entrega el informe de investigación a la Dirección Técnica. 4. Recibe Certificación de Aprobación.	2. Revisa el informe de investigación. 3. Si esta correcto se envía Certificación de Aprobación.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
5. Recibe Acta de Corrección. 6. Arregla los errores marcados. 7. Envía el informe de investigación a la Dirección Técnica.	3.1 Si no esta correcto se envía Acta de Corrección.
Poscondiciones	Al continuar incorrecto se retorna al evento 1 hasta ser correcto.

Tabla 3 Descripción CU_Entregar Informe de Investigación.

3.2.5.2 Descripción CU_Entregar Proyecto de Explotación.

Caso de Uso:	Entregar Proyecto de Explotación
Actores:	Concesionarios.
Resumen:	Luego de ser aprobado el informe de investigación el Concesionario entrega el Proyecto de Explotación a la Dirección Técnica.
Precondiciones:	Debe de estar aprobado el informe de la investigación.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Entrega el Proyecto de Explotación a la Dirección Técnica. 4. Recibe Certificación de Aprobación.	2. Revisa el Proyecto de Explotación.

	3. Si esta correcto se envía Certificación de Aprobación.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
1.1. Si es un Concesionario que utiliza las Aguas Minerales Naturales entrega el Proyecto de Explotación y Procesamiento.	
5. Recibe Acta de Corrección.	3.1 Si no está correcto se envía Acta de Corrección.
6. Arregla los errores marcados.	
7. Envía el Proyecto de Explotación a la Dirección Técnica.	
Poscondiciones	Al continuar incorrecto se retorna al evento 1 hasta ser correcto.

Tabla 4 Descripción CU_Entregar Proyecto de Explotación.

3.2.5.3. Descripción CU_Entregar Balance.

Caso de Uso:	Entregar Balance
Actores:	Concesionarios.
Resumen:	Anualmente los Concesionarios tienen que entregar a la Dirección Técnica el informe del Balance o Estado Anual de los Recursos Disponibles y de Explotación de las Aguas Minerales Naturales.
Precondiciones:	Deben de estar creados los siguientes documentos, el formulario Resumen General de los Yacimientos de Aguas Minerales, el formulario del Movimiento o Comportamiento de los Recursos Disponibles y de Explotación de Aguas Minerales, el documento Proyecto de Resolución de Balance de Agua y la gráfica de Comportamiento de los diferentes

	parámetros hidrogeológicos y variables o elementos del período analizado.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Entrega el Balance a la Dirección Técnica.	2. Revisa el Balance.
4. Recibe Certificación de Aprobación.	3. Si está correcto se envía Certificación de Aprobación.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
5. Recibe Acta de Corrección.	3.1 Si no esta correcto se envía Acta de Corrección.
6. Arregla los errores marcados.	
7. Envía el Proyecto de Explotación a la Dirección Técnica.	
Poscondiciones	Al continuar incorrecto se retorna al evento 1 hasta ser correcto.

Tabla 5 Descripción CU_Entregar Balance.

3.2.5.4. Descripción CU_Consultar Información.

Caso de Uso:	Consultar Información
Actores:	Usuarios.
Resumen:	Este caso de uso comienza con la entrada de algún usuario a la aplicación que desea consultar una información determinada.
Precondiciones:	El usuario para poder ver la información que desea consultar debe tener los permisos previos.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. Solicita consultar una información	

determinada.	2. Verifica usuario y accesibilidad a la información.
	3. Si todo está correcto muestra información solicitada.
4. Consulta información.	
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
	3.1 Si no está correcto se envía un mensaje al usuario informando que no tiene permiso para ver esa información.
Poscondiciones	

Tabla 6 Descripción CU_Consultar Información.

3.2.6. Diagrama de Objetos

A través del modelado de los objetos del negocio, se identifican las entidades del negocio y los trabajadores que participan en las actividades que se realizan en el mismo. Describe cómo cada caso de uso es llevado a cabo por un conjunto de trabajadores que utilizan un conjunto de entidades y unidades de trabajo del negocio. A continuación se muestra el Modelo de Objetos propuesto:

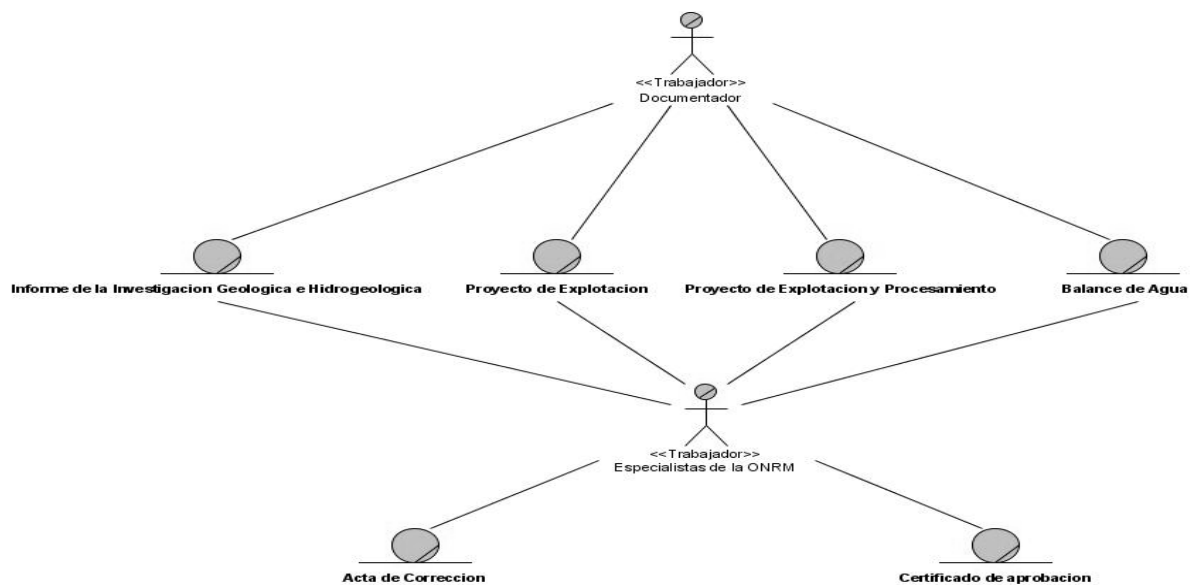


Figura 5 Diagrama de Objetos del Negocio.

3.3 Requerimientos del sistema

En este epígrafe se listan los requisitos funcionales y no funcionales que debe cumplir la aplicación, siendo esto una de las premisas fundamentales en el desarrollo del software debido a que en ellos se reflejan las necesidades del cliente. Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Las propiedades o cualidades que el producto debe tener son llamados requerimientos no funcionales.

Requerimientos funcionales

- R.1 Gestionar datos de Yacimientos
 - R.1.1 Adicionar datos de Yacimientos
 - R.1.2 Modificar datos de Yacimientos
 - R.1.3 Buscar datos de Yacimientos
- R.2 Gestionar Recursos Disponibles
 - R.2.1 Adicionar Recursos Disponibles
 - R.2.2 Modificar Recursos Disponibles
 - R.2.3 Buscar Recursos Disponibles
- R.3 Gestionar Recursos de Explotación
 - R.3.1 Adicionar Recursos de Explotación
 - R.3.2 Modificar Recursos de Explotación
 - R.3.3 Buscar Recursos de Explotación
- R.4 Generar Balance General
- R.5 Mostrar Información de Yacimientos.
- R.6 Mostrar Información de Materia Prima.
- R.7 Mostrar Búsquedas Especializadas.

Requerimientos No Funcionales

R.8 Requisitos de Software

Cliente

- Navegador Internet Explorer 4.0 o superior, o compatible con Mozilla.

Servidor

- Apache Web Server para Linux.

- Postgre

R.9 Requisitos de Hardware

Cliente

- Procesador Pentium 2.0 GHz o superior.
- 128 MB de memoria RAM mínima, recomendable (256 RAM).
- Mínimo 20 GB de disco duro.

Servidor

- Procesador Pentium 2.8 GHz o superior
- 512 MB de memoria RAM mínima, recomendable (1GB o mayor)
- Mínimo 80 GB de disco duro, recomendable (120 GB o mayor)

R.10 Requerimientos de usabilidad

- El Sistema podrá ser usado por los especialistas de la ONRM con un conocimiento básico sobre el uso de las computadoras.

R.11 Restricciones en el diseño y la implementación.

- La aplicación será desarrollada utilizando lenguaje de programación PHP, como Framework se utilizará Symfony, el IDE de Desarrollo será Eclipse y como Gestor de Base de Datos se utilizará Postgre.
- El tipo de arquitectura que se utilizará será la arquitectura en Capas específicamente la de tres capas.
- Como herramienta de modelado CASE se utilizará Visual Paradigm.

R.12 Requerimientos de apariencia o interfaz externa

- Diseño sencillo y fácil de usar, permitiendo que no sea necesario mucho entrenamiento para utilizar el sistema por los especialistas de la oficina.

R.13 Requerimientos de confiabilidad y seguridad

- Identificar al usuario antes de que pueda realizar cualquier acción sobre la configuración del sistema.

- Garantizar que la información sea vista únicamente por quién tiene derecho a verla.
- Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- Garantía de que el sistema funcione correctamente aun cuando no haya conectividad.
- El sistema de gestión de bases de datos debe tener soporte para recuperación ante fallos y errores.

R.14 Requerimientos de portabilidad y operatividad.

- La aplicación será multiplataforma para que sea operable desde cualquier sistema operativo.

3.4. Descripción del Sistema Propuesto

3.4.1. Descripción de los actores

Actor	Descripción
Especialista de la DT	La Dirección Técnica es un departamento dentro de la Oficina Nacional de Recursos Mineros (ONRM). Los especialistas que trabajan en dicho departamento se encargan de llevar el control sobre el estado de los recursos disponibles y en explotación que estén siendo utilizados por los concesionarios, así como realizar un balance anual sobre estos.
Usuario	Este es cualquier persona, sistema o centro que este interesado en consultar información relacionada con el tema que se gestiona. (Centros de Investigación, Universidades, el CITMA, Recursos Hidráulicos, posibles inversionistas, entidades gubernamentales, la Oficina Nacional de Estadística)

Tabla 7 Descripción de los Actores.

3.4.2. Descripción CU_Gestionar datos de Yacimientos

Caso de Uso:	Gestionar datos de Yacimientos.
Actores:	Especialista de la DT
Resumen:	<p>Este CU consiste en que el Especialista de la DT selecciona la operación que desea realizar sobre el yacimiento, operaciones tales como:</p> <ul style="list-style-type: none"> * Adicionar datos Yacimientos * Buscar datos Yacimientos * Modificar datos de Yacimientos
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF1
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. El Especialista de la DT selecciona el vínculo Yacimientos para realizar las operaciones con los datos del Yacimiento.</p>	<p>2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema.</p> <p>3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra un menú con las siguientes opciones:</p> <ul style="list-style-type: none"> * Adicionar datos de Yacimientos

<p>4. El Especialista de la DT selecciona la opción que desea.</p>	<p>* Buscar datos de Yacimientos</p> <p>* Modificar datos de Yacimientos</p> <p>5. El sistema ejecuta la opción que selecciono el Especialista de la DT.</p> <p>* Si este selecciona Adicionar datos de Yacimientos, ir a la sección "Adicionar datos de Yacimientos".</p> <p>* Si este selecciona Buscar datos de Yacimientos, ir a la sección "Buscar datos de Yacimientos".</p> <p>* Si este selecciona Modificar datos de Yacimientos, ir a la sección "Modificar datos de Yacimientos".</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre esto."No tiene permisos para esta sección" Finalizando así el CU.</p>
<p>Poscondiciones</p>	<p>-</p>
<p>Sección "Adicionar datos de Yacimientos"</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Usuario del sistema selecciona Adicionar datos de Yacimientos manualmente.</p> <p>3. El Usuario llena los campos del formulario, dándole al botón Siguiente.</p>	<p>2. El sistema muestra un formulario con los campos a llenar.</p> <p>4. El sistema verifica que todos los campos hayan sido llenados correctamente, y muestra al usuario una</p>

<p>5. El usuario comprueba que esos son los datos que desea insertar y presiona al botón Guardar.</p>	<p>confirmación en forma de reporte.</p> <p>6. El sistema introduce los datos y muestra un mensaje al usuario “Datos insertados correctamente”, de lo contrario muestra un mensaje “Error en la inserción de datos”.</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1.1 El Usuario del sistema selecciona Adicionar datos de Yacimientos importando un fichero.</p> <p>1.2 Selecciona el fichero a importar cargando los datos que este contiene.</p>	<p>1.3 El sistema verifica si los datos fueron importados correctamente de ser así los envía a la BD mostrándole al usuario un reporte de confirmación con todos los datos insertados, de lo contrario le muestra un mensaje de error notificando que los datos no fueron insertados.</p>
<p>Sección “Buscar datos de Yacimientos”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Usuario del sistema selecciona Buscar datos de Yacimientos.</p> <p>2. Introduce el criterio de búsqueda para mostrar los datos que desee del yacimiento.</p>	<p>3. El sistema según el criterio de búsqueda que introdujo el usuario busca los datos y los muestra en una tabla.</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>3.1 Si no se pudo realizar la búsqueda se le muestra un mensaje al usuario. “No se pudo efectuar la búsqueda”</p>

Sección “ Modificar datos de Yacimientos”	
Acción del Actor	Respuesta del Sistema
<p>1. El Usuario del sistema selecciona Modificar datos de Yacimientos.</p> <p>2. Busca el o los yacimientos que desea modificar por criterios de búsqueda.</p> <p>4. El usuario selección sobre cual desea modificar los datos.</p> <p>6. Realiza las modificaciones.</p>	<p>3. El sistema muestra los yacimientos en una tabla.</p> <p>5. Muestra los datos para efectuar la modificación.</p> <p>7. Registra la modificación y muestra un reporte de confirmación con los datos ya modificados.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>7.1 Si la modificación no se realizó satisfactoriamente se le envía un mensaje al usuario para informarle que existe errores y se marca de color rojo los campos donde están los errores.</p>

Tabla 8 Descripción CU_Gestionar Datos Yacimiento.

3.4.3. Descripción CU_Gestionar Recursos Disponibles

Caso de Uso:	Gestionar Recursos Disponibles
Actores:	Especialista de la DT
Resumen:	<p>Este CU consiste en que el Especialista de la DT selecciona la operación que desea realizar sobre recursos disponibles, operaciones tales como:</p> <ul style="list-style-type: none"> * Adicionar Recursos Disponibles * Modificar Recursos Disponibles

	* Buscar Recursos Disponibles
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF2
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista de la DT selecciona el vínculo Recursos Disponibles para realizar las operaciones.	2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema.
	3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra un menú con las siguientes opciones:
	* Adicionar Recursos Disponibles
	* Modificar Recursos Disponibles
	* Buscar Recursos Disponibles
4. El Especialista de la DT selecciona la opción que desea.	5. El sistema ejecuta la opción que seleccionó el Especialista de la DT.
	* Si este selecciona Adicionar Recursos Disponibles, ir a la sección “Adicionar Recursos Disponibles”.
	* Si este selecciona Modificar Recursos Disponibles, ir a la

	<p>sección “Modificar Recursos Disponibles”.</p> <p>* Si este selecciona Buscar Recursos Disponibles, ir a la sección “Buscar Recursos Disponibles”.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre esto. Finalizando así el CU.
Poscondiciones	-
Sección “ Adicionar datos de Recursos Disponibles”	
Acción del Actor	Respuesta del Sistema
<p>1. El Usuario del sistema selecciona Adicionar Recursos Disponibles manualmente.</p> <p>3. El usuario llena los campos del formulario.</p> <p>5. El usuario comprueba que esos son los datos que desea insertar y presiona al botón Guardar.</p>	<p>2. El sistema muestra un formulario con los campos a llenar.</p> <p>4. El sistema verifica que todos los campos hayan sido llenados correctamente, y muestra al usuario una confirmación en forma de reporte.</p> <p>6. El sistema introduce los datos y muestra un mensaje al usuario “Datos insertados correctamente”, de lo contrario muestra un mensaje “Error en la inserción de datos”.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
<p>1.1 El Usuario del sistema selecciona Adicionar Recursos Disponibles importando un fichero.</p> <p>1.2 Selecciona el fichero a importar</p>	

<p>cargando los datos que este contiene.</p>	<p>1.3 El sistema verifica si los datos fueron importados correctamente de ser así los envía a la BD mostrándole al usuario un reporte de confirmación con todos los datos insertados, de lo contrario le muestra un mensaje de error notificando que los datos no fueron insertados.</p>
----------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Sección “ Modificar datos de Recursos Disponibles ”

Acción del Actor	Respuesta del Sistema
<p>1. El Usuario del sistema selecciona Modificar Recursos Disponibles.</p> <p>2. Busca el yacimiento por criterios de búsqueda. que desea modificar sus recursos disponibles</p> <p>4. El usuario selección sobre cual desea modificar los datos.</p> <p>6. Realiza las modificaciones.</p>	<p>3. Busca el yacimientos y lo muestra en una tabla</p> <p>5. Muestra los datos para efectuar la modificación.</p> <p>7. Registra la modificación y muestra un reporte de confirmación con los datos ya modificados.</p>

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>7.1 Si la modificación no se realizó satisfactoriamente se le envía un mensaje al usuario para informarle que existe errores y se marca de color rojo los campos donde están los errores.</p>

Sección “Buscar datos de Recursos Disponibles”

Acción del Actor	Respuesta del Sistema

1. El Usuario del sistema selecciona Buscar Recursos Disponibles.	
2. Introduce el criterio de búsqueda para mostrar los datos que desee de los Recursos Disponibles.	
	3. El sistema según el criterio que introdujo el usuario busca los datos y los muestra al usuario.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Si no se pudo realizar la búsqueda se le muestra un mensaje al usuario. "No se pudo efectuar la búsqueda"

Tabla 9 Descripción CU_Gestionar Recursos Disponibles.

3.4.4 Descripción CU _ Gestionar Recursos de Explotación

Caso de Uso:	Gestionar Recursos de Explotación
Actores:	Especialista de la DT
Resumen:	Este CU consiste en que el Especialista de la DT selecciona la operación que desea realizar sobre recursos disponibles, operaciones tales como: * Adicionar Recursos de Explotación * Modificar Recursos de Explotación * Buscar Recursos de Explotación
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF3
Prioridad	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. El Especialista de la DT selecciona el vínculo Recursos de Explotación para realizar las operaciones.</p>	<p>2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema.</p> <p>3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra un menú con las siguientes opciones:</p> <ul style="list-style-type: none"> * Adicionar Recursos Disponibles * Modificar Recursos Disponibles * Buscar Recursos Disponibles
<p>4. El Especialista de la DT selecciona la opción que desea.</p>	<p>5. El sistema ejecuta la opción que selecciono el Especialista de la DT.</p> <ul style="list-style-type: none"> * Si este selecciona Adicionar Recursos de Explotación, ir a la sección “Adicionar Recursos de Explotación”. * Si este selecciona Modificar Recursos de Explotación, ir a la sección “Modificar Recursos de Explotación”. * Si este selecciona Buscar Recursos de Explotación, ir a la sección “Buscar Recursos de Explotación”.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre</p>

	esto. Finalizando así el CU.
Poscondiciones	-
Sección “ Adicionar datos de Recursos de Explotación ”	
Acción del Actor	Respuesta del Sistema
1. El Usuario del sistema selecciona Adicionar Recursos de Explotación manualmente.	2. El sistema muestra un formulario con los campos a llenar.
3. El usuario llena los campos del formulario.	4. El sistema verifica que todos los campos hayan sido llenados correctamente, y muestra al usuario una confirmación en forma de reporte.
5. El usuario comprueba que esos son los datos que desea insertar y presiona al botón Guardar.	6. El sistema introduce los datos y muestra un mensaje al usuario “ Datos insertados correctamente ”, de lo contrario muestra un mensaje “ Error en la inserción de datos ”.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
1.1 El Usuario del sistema selecciona Adicionar Recursos de Explotación importando un fichero.	
1.2 Selecciona el fichero a importar cargando los datos que este contiene.	
	1.3 El sistema verifica si los datos fueron importados correctamente de ser así los envía a la BD mostrándole al usuario un reporte de confirmación con todos los datos insertados, de lo contrario le muestra un mensaje de error notificando que los datos no fueron insertados.
Sección “ Modificar datos de Recursos de Explotación ”	
Acción del Actor	Respuesta del Sistema
1. El Usuario del sistema selecciona	

<p>Modificar Recursos de Explotación.</p> <p>2. Busca el yacimiento por criterios de búsqueda. que desea modificar sus recursos de explotación</p> <p>4. El usuario selección sobre cual desea modificar los datos.</p> <p>6. Realiza las modificaciones.</p>	<p>3. Busca el yacimientos y lo muestra en una tabla</p> <p>5. Muestra los datos para efectuar la modificación.</p> <p>7. Registra la modificación y muestra un reporte de confirmación con los datos ya modificados.</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>7.1 Si la modificación no se realizó satisfactoriamente se le envía un mensaje al usuario para informarle que existe errores y se marca de color rojo los campos donde están los errores.</p>
<p>Sección “ Buscar datos de Recursos de Explotación”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Usuario del sistema selecciona Actualizar Recursos en Explotación.</p> <p>2. Introduce el criterio de búsqueda para mostrar los datos que desee de los Recursos en Explotación.</p>	<p>3. El sistema según el criterio que introdujo el usuario busca los datos y los muestra al usuario.</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>3.1 Si no se pudo realizar la búsqueda se le muestra un mensaje al usuario. “No se pudo efectuar la búsqueda”</p>

Tabla 10 Descripción CU_Gestionar Recursos de Explotación.

3.4.5. Descripción CU_Generar Balance General

Caso de Uso:	Generar Balance General
Actores:	Especialista de la DT
Resumen:	Este CU consiste en que el Especialista de la DT selecciona la operación generar balance general para elaborar un documento que será enviado a varias entidades.
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF4
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Especialista de la DT selecciona el vínculo Generar balance General.	2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema.
	3. Si los permisos son correctos y tiene acceso a este nivel. El sistema le permite elaborar el balance en dependencia hacia quien este dirigido (ACE, Provincia o Municipio etc.)
4. El Especialista de la DT elabora el balance general escogiendo la información que desea según el destinatario.	5. El sistema registra el balance y le da la opción de enviar y/o de imprimir.
Flujos Alternos	

Acción del Actor	Respuesta del Sistema
	3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre esto. Finalizando así el CU.
Poscondiciones	-

Tabla 11 Descripción CU_Generar Balance General.

3.4.6. Descripción CU_Mostrar información Yacimiento

Caso de Uso:	Mostrar información Yacimiento
Actores:	Usuario
Resumen:	<p>Este CU consiste en que el Usuario selecciona la operación donde se le muestran todas opciones posibles para realizar búsquedas de los datos de los Yacimientos y mostrarlas , las cuales pueden ser sobre dos tipos de Yacimientos:</p> <p>*Yacimiento en Explotación</p> <p>*Yacimiento sin Explotar</p>
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF5
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Usuario selecciona el vínculo Mostrar información Yacimiento.	2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este

<p>4. El Usuario selecciona la opción que desea para ver la información.</p>	<p>sistema.</p> <p>3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra un menú:</p> <ul style="list-style-type: none"> * Yacimiento en Explotación. * Yacimiento sin Explotar. <p>5. El sistema ejecuta la opción que selecciona el Usuario.</p> <p>* Si este selecciona Yacimiento en Explotación, ir a esta sección</p> <p>* Si este selecciona Yacimiento sin Explotar, ir a esta sección.</p>
<p>Flujos Alternos</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre esto. Finalizando así el CU.</p>
<p>Poscondiciones</p>	<p>-</p>
<p>Sección “Yacimiento en Explotación”</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Usuario del sistema selecciona Yacimiento en Explotación.</p> <p>3. El Usuario selecciona el Yacimiento del cual desea ver su información.</p>	<p>2. El sistema muestra un listado de todos los nombres de los Yacimientos en Explotación.</p> <p>4. El sistema muestra la información primaria del Yacimiento seleccionado.</p> <ul style="list-style-type: none"> • Concesionario

	<ul style="list-style-type: none"> • Tipo de materia prima • Provincia • Municipio
Flujos Alternos	
Sección “Yacimiento sin Explotar”	
Acción del Actor	Respuesta del Sistema
<p>1. El Usuario del sistema selecciona Yacimiento en Explotación.</p> <p>3. El Usuario selecciona el Yacimiento del cual desea ver su información.</p>	<p>2. El sistema muestra un listado de todos los nombres de los Yacimientos sin Explotar.</p> <p>4. El sistema muestra la información primaria del Yacimiento seleccionado.</p> <p>Tipo de materia prima</p> <p>Provincia</p> <p>Municipio</p> <p>Uso Potencial</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Tabla 12 Descripción CU_Mostrar Información Yacimiento.

3.4.7. Descripción CU_Mostrar información Materia Prima.

Caso de Uso:	Mostrar información Materia Prima
Actores:	Usuario
Resumen:	Este CU consiste en que el Usuario selecciona la opción Materia Prima, para ver la información de esta en un Yacimiento determinado.

Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF6
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Usuario escoge la opción Materia Prima.	2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema
	3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra un Listado con todos los Yacimientos.
4. El Usuario selecciona el Yacimiento del cual quiere ver la información de la Materia Prima.	5. El sistema según el nombre del Yacimiento busca la información y se la muestra al Usuario:
	<ul style="list-style-type: none"> • Elementos de Calidad. • Provincia. • Uso Actual.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

Tabla 13 Descripción CU_Mostrar Información Materia Prima.

3.4.8. Descripción CU_Mostrar Búsquedas Especializadas.

Caso de Uso:	Mostrar búsquedas especializadas
Actores:	Usuario
Resumen:	Este CU consiste en que el usuario selecciona la opción Búsquedas Especializadas y se le muestra un formulario con los campos requeridos para filtrar la búsqueda que desea.
Precondiciones:	El usuario debe identificarse en el sistema desde el Portal para poder verificar la autenticidad del usuario y los permisos que tendrá disponibles en el sistema de Balance según su nivel de acceso.
Referencias	RF7
Prioridad	Critico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Usuario selecciona el vínculo Búsquedas Especializadas.	<p>2. El sistema verifica desde que accede al Balance de las Aguas Minerales la existencia del usuario autenticado en el Portal y los permisos correspondientes para este sistema.</p> <p>3. Si los permisos son correctos y tiene acceso a este nivel. El sistema muestra una interfaz con un formulario con los campos indicados para filtrar la información que el Usuario desea.</p> <ul style="list-style-type: none"> ● Estado del Yacimiento. ● Provincia. ● Municipio ● Tipo de Materia prima. ● Uso Potencial.

4. El Usuario selecciona los campos por los cuales quiere realizar la búsqueda.	<ul style="list-style-type: none"> • Concesionario
	5. El sistema realiza la búsqueda solicitada por el Usuario y muestra los resultados de la misma.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Si el usuario autenticado no tiene permiso para este nivel el sistema muestra un mensaje informándole sobre esto. Finalizando así el CU.
Poscondiciones	-

Tabla 14 Descripción CU_Mostar Búsquedas Especializadas.

3.4.9 Casos de Uso del Sistema

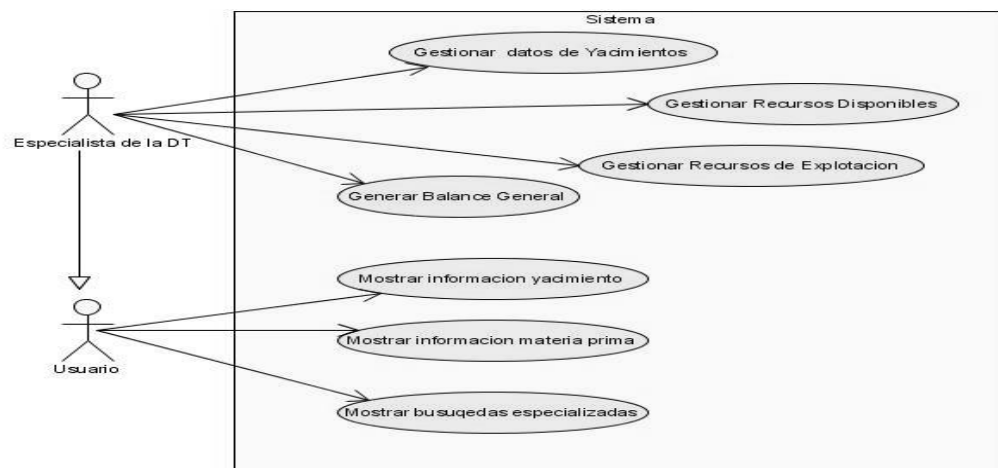


Figura 6 Diagrama de Casos de Uso del Sistema.

3.5 Conclusiones

Durante el desarrollo de todo el capítulo se llegan a resultados sustanciales para la puesta en marcha de la futura aplicación tales como: Descripción detallada de la propuesta de solución a través de la modelación del negocio propuesto; identificando los actores y trabajadores que intervienen, así como los Casos de Uso correspondientes. Se identifican los requisitos funcionales y no funcionales con los cuales debe cumplir la aplicación, llegando de esta forma a una Modelación detallada del sistema a desarrollar.

CAPÍTULO 4

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

4.1 Introducción

En este capítulo se realiza el análisis, diseño e implementación de la propuesta de solución para el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales. A través de los flujos de Análisis y Diseño, se presenta el diagrama de clases del análisis y del diseño de los diferentes casos de usos definidos en el capítulo anterior, a demás de los diagramas de interacción (colaboración) correspondientes a cada uno de ellos, se muestra el modelo lógico y físico de datos, y se especifican los principios de diseño. Por último se realiza el modelo de implementación donde se modelan el diagrama de despliegue que 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

El modelo de análisis es una parte fundamental del flujo de trabajo de Análisis y Diseño debido a que ayuda a refinar y estructurar los requisitos funcionales, es uno de los artefactos que se genera en este flujo de trabajo, proporcionando un mayor entendimiento para los desarrolladores pues se logra obtener una visión más clara de que debe hacer el sistema. Se desarrolla utilizando el lenguaje de los desarrolladores y es la primera aproximación al modelo de diseño es por ello su vital importancia , siendo un paso opcional a desarrollar. En éste modelo los casos de uso del sistema se describen a través de las clases del análisis y sus objetos.

4.2.1 Diagramas de clases del análisis

El diagrama de clases del análisis es uno de los artefactos que se generan en el modelo del análisis. Se utilizan para el modelado de las realizaciones de los casos de uso, mostrando las clases y sus relaciones.

Las clases que se encuentran en estos diagramas se clasifican en Interfaz, Control y Entidad. Las clases interfaz se encargan de modelar la interacción del actor con el sistema. Representan abstracciones de ventanas, formularios, así como la comunicación con otros sistemas o dispositivos.

Las clases controladoras modelan los aspectos dinámicos del sistema, de forma tal que puedan coordinar las acciones y los flujos de control. Las clases entidad modelan información que posee una larga vida y que a menudo es persistente. Suelen mostrar una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema. (Ver Anexos #2)

4.2.2 División por paquetes

El sistema estará dividido por paquetes, lo cual facilita el desarrollo del mismo de forma tal que se puedan organizar los artefactos que se generan en el análisis en partes más manejables y factibles para el desarrollador. Dichos paquetes en el análisis del sistema se nombran Paquete Gestión, Paquete Recuperación, Paquete Balance, la agrupación de estos paquetes se realizó por funcionalidad de los casos de uso por lo que queda agrupado de la siguiente forma:

Paquete Gestión agrupa los casos de uso:

- Gestionar Datos de Yacimiento.
- Gestionar Recursos Disponibles.
- Gestionar Recursos de Explotación.

Paquete Balance:

- Generar Balance General.

Paquete Recuperación:

- Mostar Información Yacimiento.
- Mostar Información Materia Prima.
- Mostar Búsquedas Especializadas.

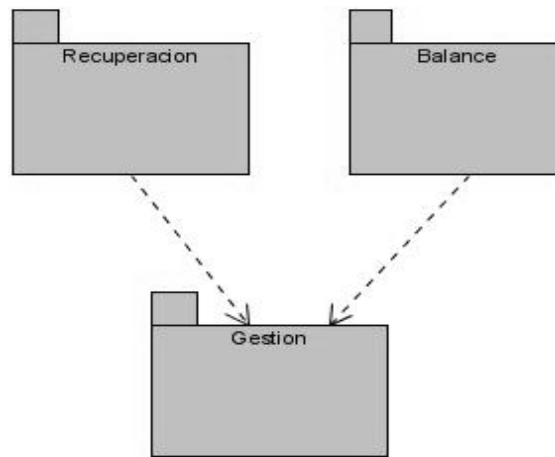


Figura 7 Diagrama de Paquetes del Análisis.

4.3 Diagramas de interacción.

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos del sistema, son la realización de los casos de uso. Modelan instancias concretas o prototípicas de clases interfaces, componentes y nodos, así como los mensajes enviados entre ellos. Los diagramas de interacción se denotan en dos tipos diferentes, los diagramas de secuencia y diagramas de colaboración, que en su generalidad los dos responden a realizaciones de los casos de usos correspondientes, en el desarrollo de este Trabajo de Diploma se realizó específicamente los diagramas de colaboración.

4.3.1 Diagrama de Colaboración.

El diagrama de colaboración muestra las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar. Este presenta elementos como: objetos y enlaces. Los objetos representan las clases con sus nombres y los enlaces que constituyen una instancia de una asociación en un diagrama de clases. Los enlaces están acompañados por un número que indica el orden dentro de la interacción, simbolizando el orden en que se van a efectuar las llamadas y respuestas del sistema. (Ver el Anexo #3.)

4.4 Modelo de Diseño

El modelo de diseño 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. Además, el

modelo de diseño sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como entrada fundamental de las actividades de implementación

Durante el modelado del diseño se tiene en cuenta el framework a utilizar para el desarrollo de la aplicación, en el caso de este trabajo se utilizará Symfony el cual es muy potente en el desarrollo de aplicaciones Web debido a sus funcionalidades.

4.4.1 Diagrama de Clases del Diseño

Los diagramas o modelos de clases del diseño sirven para la toma de decisiones, en el diseño y la implementación. Los modelos brindan una vista a menor y menos costosa escala, de la solución final y resulta mucho más fácil efectuar cambios a los mismos, una vez detectados los problemas, que tener que efectuar cambios a la solución ya implementada. En el caso de este trabajo dado la novedad y complejidad de la tecnología seleccionada se hace necesario dividir el diseño en tres paquetes lógicos que representan la filosofía del framework dejando claramente evidenciada la funcionalidad de cada uno de los elementos que se encuentran agrupados en dichos paquetes.

- **Modelo**

En la aplicación con el uso de Symfony, el acceso y la modificación de los datos almacenados en la base de datos se realiza mediante objetos; de esta forma nunca se accede de forma explícita a la base de datos. Este comportamiento permite un alto nivel de abstracción y permite una fácil portabilidad.

La forma en que se accede a la base de datos es a través de una interfaz que traduce la lógica de los objetos a la lógica relacional, esta interfaz se llama ORM (object-relational mapping) o “- mapeo de objetos a bases de datos”, y está formada por objetos que permiten acceder a los datos y que contienen en sí mismos el código necesario para hacerlo, específicamente se utiliza Propel como ORM. Debido a esta forma de mapeo se generan cuatro clases por cada tabla del esquema, quedando organizadas dentro de este paquete todas las relaciones existentes entre ellas.

BaseClasses	Son las que se generan directamente a partir del esquema. Nunca se deben modificar esas clases, porque cada vez que se genera el modelo, se borran todas las clases.
ClasesPeer	Las clases “peer” contienen métodos estáticos para obtener registros de la base de datos.

BasePeer	Son las clases que tienen métodos estáticos para trabajar con las tablas de la base de datos. Proporcionan los medios necesarios para obtener los registros de las tablas.
Clases	Son las clases en las que se añaden los métodos propios y heredan de las clases bases además de ser las clases objeto que representan un registro de la base de datos.

- **Vista**

La vista se encarga de producir las páginas que se muestran como resultado de las acciones. La vista en Symfony está compuesta por diversas partes, los success, las paginas clientes y los formularios, cada success responde a un action siendo esto una facilidad pues de esta forma estas partes pueden ser modificables por la persona que normalmente trabaja con cada aspecto del diseño de las aplicaciones.

- **Controlador**

Este paquete está dividido en dos partes: el controlador frontal, que es el único punto de entrada a la aplicación para un entorno dado además de manejar todas las peticiones Web, y las acciones, que contienen la lógica de las páginas, por cada acción es una clase que se crea y todas a su vez heredan de sfaction la cual contiene métodos comunes para las demás clases facilitando la usabilidad y en caso de cambio no se afectaría nada más que aquellas clases implicadas en él.

Para el modelado del diseño se siguieron varios patrones de diseño, los cuales se mencionan a continuación:

Básicos: GRASP (General Responsibility Assignment Software Patterns)

Creador:

El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento.

Beneficios:

- Se brinda soporte a un bajo acoplamiento, lo cual supone menos dependencias respecto al mantenimiento y mejores oportunidades de reutilización. Es probable que el acoplamiento no aumente, pues la clase creada tiende a ser visible a la clase creador, debido a las asociaciones actuales que llevaron a elegirla como el parámetro adecuado.

Controlador:

Este patrón ofrece una guía para tomar decisiones apropiadas que generalmente se aceptan. Un defecto frecuente al diseñar controladores consiste en asignarles demasiada responsabilidad. Normalmente un controlador debería delegar a otros objetos el trabajo que ha de realizarse mientras coordina la actividad.

Beneficios:

- Mayor potencial de los componentes reutilizables. Garantiza que la empresa o los procesos de dominio sean manejados por la capa de los objetos del dominio y no por la de la interfaz. Desde el punto de vista técnico, las responsabilidades del controlador podrían cumplirse en un objeto de interfaz, pero esto supone que el código del programa y la lógica relacionada con la realización de los procesos del dominio puro quedarían incrustados en los objetos interfaz o ventana.
- Reflexionar sobre el estado del caso de uso. A veces es necesario asegurarse de que las operaciones del sistema sigan una secuencia legal o poder razonar sobre el estado actual de la actividad y las operaciones en el caso de uso subyacente.

Experto:

Experto es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen. Nótese que el cumplimiento de una responsabilidad requiere a menudo información distribuida en varias clases de objetos. Ello significa que hay muchos expertos "parciales" que colaboraron en la tarea.

El patrón Experto ofrece una analogía con el mundo real.

Beneficios:

- Se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece al hecho de tener sistemas más robustos y de fácil mantenimiento.
- El comportamiento se distribuye entre las clases que cuentan con la información requerida, alentando con ello definiciones de clases "sencillas" y más cohesivas que son más fáciles de comprender y de mantener.

Bajo Acoplamiento:

El Bajo Acoplamiento es un principio que se debe recordar durante las decisiones de diseño, es la meta principal que es preciso tener presente siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño. El Bajo Acoplamiento estimula asignar una responsabilidad de modo que su colocación no incremente el acoplamiento tanto que produzca los resultados negativos propios de un alto acoplamiento.

El Bajo Acoplamiento soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que acrecientan la oportunidad de una mayor productividad. No puede considerarse en forma independiente de otros patrones como Experto o Alta Cohesión, sino que más bien ha de incluirse como uno de los principios del diseño que influyen en la decisión de asignar responsabilidades. (21)

Beneficios:

- No se afectan por cambios de otros componentes
- Fáciles de entender por separado
- Fáciles de reutilizar

Alta Cohesión:

Como el patrón Bajo Acoplamiento, también Alta Cohesión es un principio que se debe tener presente en todas las decisiones de diseño, es la meta principal que ha de buscarse en todo momento. Grady Booch señala que se da una alta cohesión funcional cuando los elementos de un componente "colaboran para producir algún comportamiento bien definido".

El patrón Alta Cohesión presenta semejanzas con el mundo real, ya que si alguien asume demasiadas responsabilidades -sobre todo las que debería delegar-, no será eficiente.

Beneficios:

- Mejoran la claridad y la facilidad con que se entiende el diseño.
- Se simplifican el mantenimiento y las mejoras en funcionalidad.

- A menudo se genera un bajo acoplamiento.
- La ventaja de una gran funcionalidad soporta una mayor capacidad de reutilización, porque una clase muy cohesiva puede destinarse a un propósito muy específico.

Avanzados: GOF (Gang of Four)

Fábrica Abstracta (Abstract Factory):

Es un patrón creacional en la clasificación de los patrones GOF. Proporciona una interfaz para crear familias de objetos sin especificar su clase de forma concreta.

Consecuencias:

- Se potencia el encapsulamiento, puesto que se aísla a los clientes de las implementaciones.
- Se incrementa la flexibilidad del diseño, resultando fácil cambiar de familia de productos.
- Se refuerza la consistencia (alta cohesión y bajo acoplamiento), puesto que se restringe el uso a productos de una sola familia cada vez.

Instancia única o Solitario (Singleton):

El patrón Singleton garantiza que una clase sólo tenga una instancia y proporciona un punto de acceso global a ésta instancia.

Consecuencias:

- Acceso controlado a la única instancia. Puede tener un control estricto sobre cómo y cuándo acceden los clientes a la instancia.
- Espacio de nombres reducido. El patrón Singleton es una mejora sobre las variables globales.
- Permite el refinamiento de operaciones y la representación. Se puede crear una subclase de Singleton.
- Permite un número variable de instancias. El patrón hace que sea fácil cambiar de opinión y permitir más de una instancia de la clase Singleton.

Modelo-Vista-Controlador (MVC):

El patrón Modelo-Vista-Controlador separa el modelamiento del dominio, la presentación, y las acciones basadas en las entradas hechas por el usuario en tres clases fundamentales:

Modelo: Administra y maneja el comportamiento y los datos del dominio de aplicación, da respuestas a peticiones de información sobre el estado de la aplicación (normalmente desde la Vista), y responde con instrucciones de cambio de estado (usualmente desde el controlador) a la vista.

Vista: Gestiona lo relacionado con mostrar la información al usuario.

Controlador: El controlador interpreta los eventos que son lanzados por la entrada estándar del usuario (normalmente mouse y teclado), informando de los mismos al modelo y/o la vista para que se ejecuten los cambios apropiadamente.

(Ver Anexo #4)

4.5 Principios de Diseño

Varios factores son los que entran en juego a la hora de diseñar una Web: el monitor del ordenador, con sus limitaciones y las diferencias entre marcas. Las múltiples plataformas. La pobre estandarización que hay entre navegadores. Las diferentes versiones de navegadores que aun están vivos en el mercado. La necesidad de incluir en toda página un sistema de navegación coherente, claro y eficiente. Las diferentes tecnologías que almacenan los servidores, siempre limitadas, que el profesional tiene que tener en cuenta al crear su diseño, para un conocimiento de lo que puede hacer y lo que no. Además debe basarse en los usuarios que la vayan a utilizar. En el caso que se analiza, son trabajadores que tienen poca experiencia en el mundo de la informática su alcance sólo llega a realizar operaciones básicas con el ordenador. Por lo que se hace necesario el guiarse por los principios de diseño Web que se siguen hoy en el mundo para lograr un diseño exitoso sin olvidarnos de las características que tienen nuestros futuros usuarios.

1. Anticipación

Con la aplicación de este principio se logra anticiparse a las necesidades y deseos del usuario, sin esperar que el usuario busque o recuerde algo, pues la información y herramientas que él necesite en cada etapa siempre estarán visibles.

2. Autonomía

Le permite al usuario sentirse libre dentro del sitio, que sienta que tiene el control sobre las operaciones que realiza, o sea, que trabaje sin dudas ni temor.

3. Consistencia

El sitio Web debe ser consistente, es decir, si un elemento se ve y se comporta de una cierta manera, se debe mantener así dentro de todo el sitio Web, o sea, hacer uso de plantillas. Aclarar que usar una

misma plantilla no quiere decir que la página sea poco interactiva para el usuario, al contrario utilizar la misma plantilla mantiene una página consistente, limpia y ordenada.

4. Eficacia del usuario

Al aplicar este principio se logra la productividad del usuario no del Sitio Web. Cada vez que el usuario tiene que esperar la respuesta del sistema, significa tiempo y/o dinero perdido, demora en una decisión importante, aprobación de un convenio que depende de los datos que devuelve el sistema, lentitud e ineficiencia en el proceso que se ha informatizado.

5. Interfaces explorables

La aplicación de este principio consta de darles a los usuarios sistemas de navegación bien señalizados y luego dejarlos libres. Darle nociones estables como saber cómo llegar al inicio.

El hacer uso de elementos visuales estables ayuda a navegar más rápido y actúan como una referencia necesaria para sentirse seguro. Como la ubicación del logo de la ONRM en la parte superior izquierda en todas las páginas del Sitio Web, el mapa de navegación así como el mantener el nombre de la sección en que se encuentra el usuario.

6. Legibilidad

Aplicar este principio es muy importante para el usuario ya que garantiza la claridad y entendimiento de la información que se brinda, a través de la utilización de textos de alto contraste y letras de un tamaño que puedan ser leídas cómodamente en los monitores más comunes se puede lograr esto. Hay que tener en cuenta que se trabaja para usuarios con un promedio de entre los 30 y 50 años, donde muchos pueden presentar problemas en la vista, por los que debemos de mantener un balance en legibilidad, tipografía, simbolismos, visualización y color en la interfaz del usuario para poder comunicarse efectivamente. En el prototipo presentado se usa como máximo 3 tipos de letras y 3 tamaños de fuente diferentes, con un máximo de 18 palabras o 50-80 caracteres por línea en la parte de texto.

7. Navegación visible

La mayoría de los usuarios no pueden mantener mapas mentales complejos. Si tienen que hacerlo, se cansarán o se perderán. Por lo que se debe reducir la navegación al máximo y ofrecer lo imprescindible de forma clara y natural.

8. Dimensiones

El diseño de Web implica una serie de limitaciones. Dada la naturaleza del medio, nuestras páginas deben de funcionar en diferentes plataformas y con distintas dimensiones de pantalla.

Actualmente la mayor parte de usuarios usa una resolución de pantalla de 1024 x 768 y en segundo lugar sigue la resolución de 800x600. Antes de empezar el diseño y la producción de un sitio de Web, se definió que formato seguiremos. Para resolución 1024x768 se trabaja con dimensiones reales de 955x600, este es el espacio visible. La diferencia en los números se debe al espacio utilizado por botones de navegación que requiere el navegador y nuestro propio sistema operativo.

9. No le tengas miedo al espacio en blanco

Es difícil valorar la importancia de los espacios en blanco, pero estos son importantes porque no sólo ayudan a reducir la carga cognoscitiva, sino que también mejoran la percepción de la información. Las estructuras complejas son más difíciles de leer, examinar, analizar y de trabajar con ellas.

Siempre que se tenga en cuenta la aplicación de los principios de diseño para la creación de un sitio Web se lograra un diseño más apropiado al objetivo que se persigue que es la satisfacción del cliente.

4.6 Diseño de la base de datos

El diseño de la BD esta conformado por el Diagrama de clases persistentes y por el Diagrama Entidad-Relación. Las clases que se encuentran en el Diagrama Entidad-Relación en colores diferentes son aquellas que pertenecen a otros esquemas de la BD general del Proyecto como Nomencladores, Común y Cartografía tributando en gran medida al Balance Nacional de Recursos Disponible y de Explotación.

4.6.1 Diagrama de clases persistentes

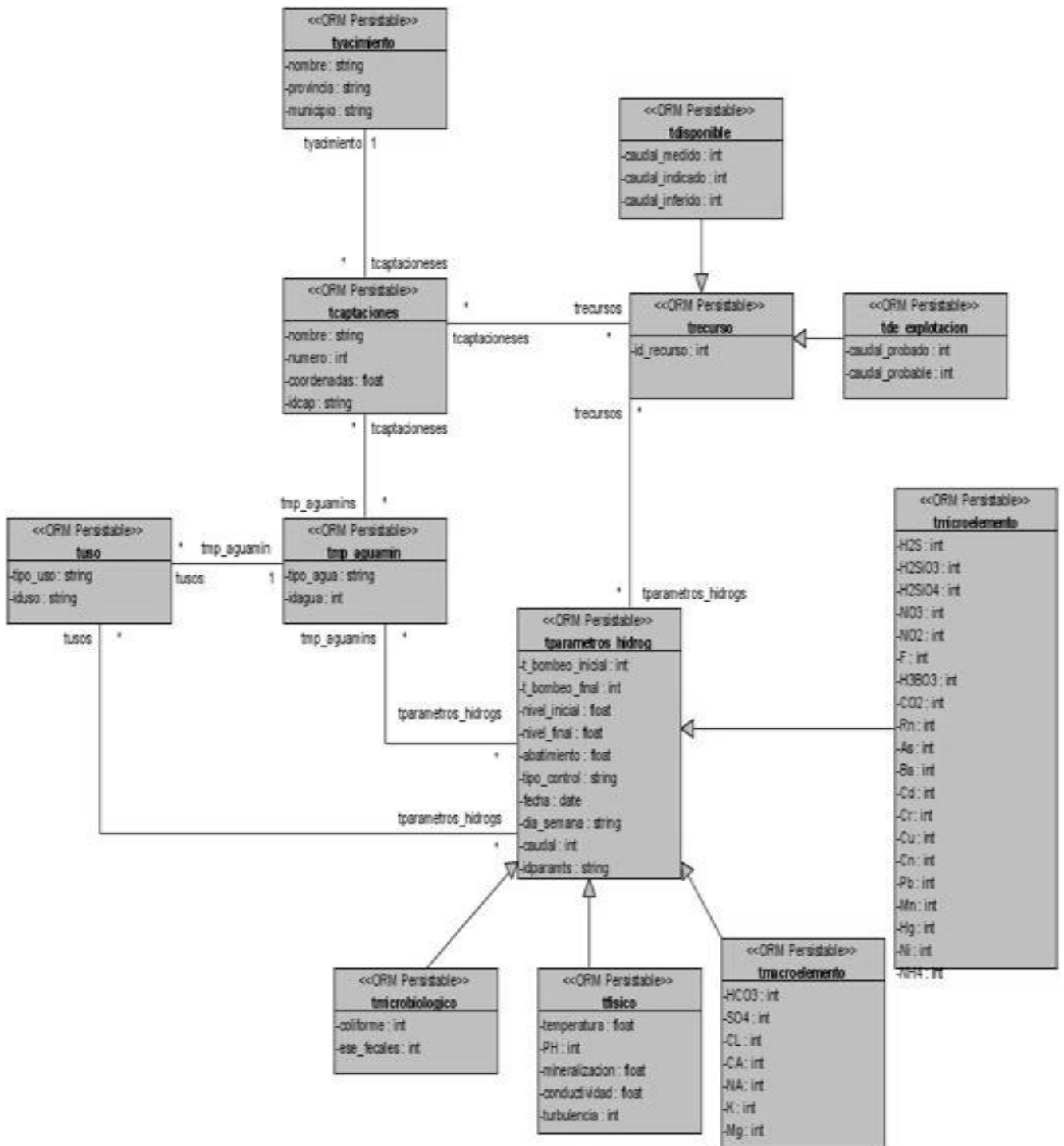


Figura 8 Diagrama de Clases Persistentes.

4.6.2 Modelo de datos

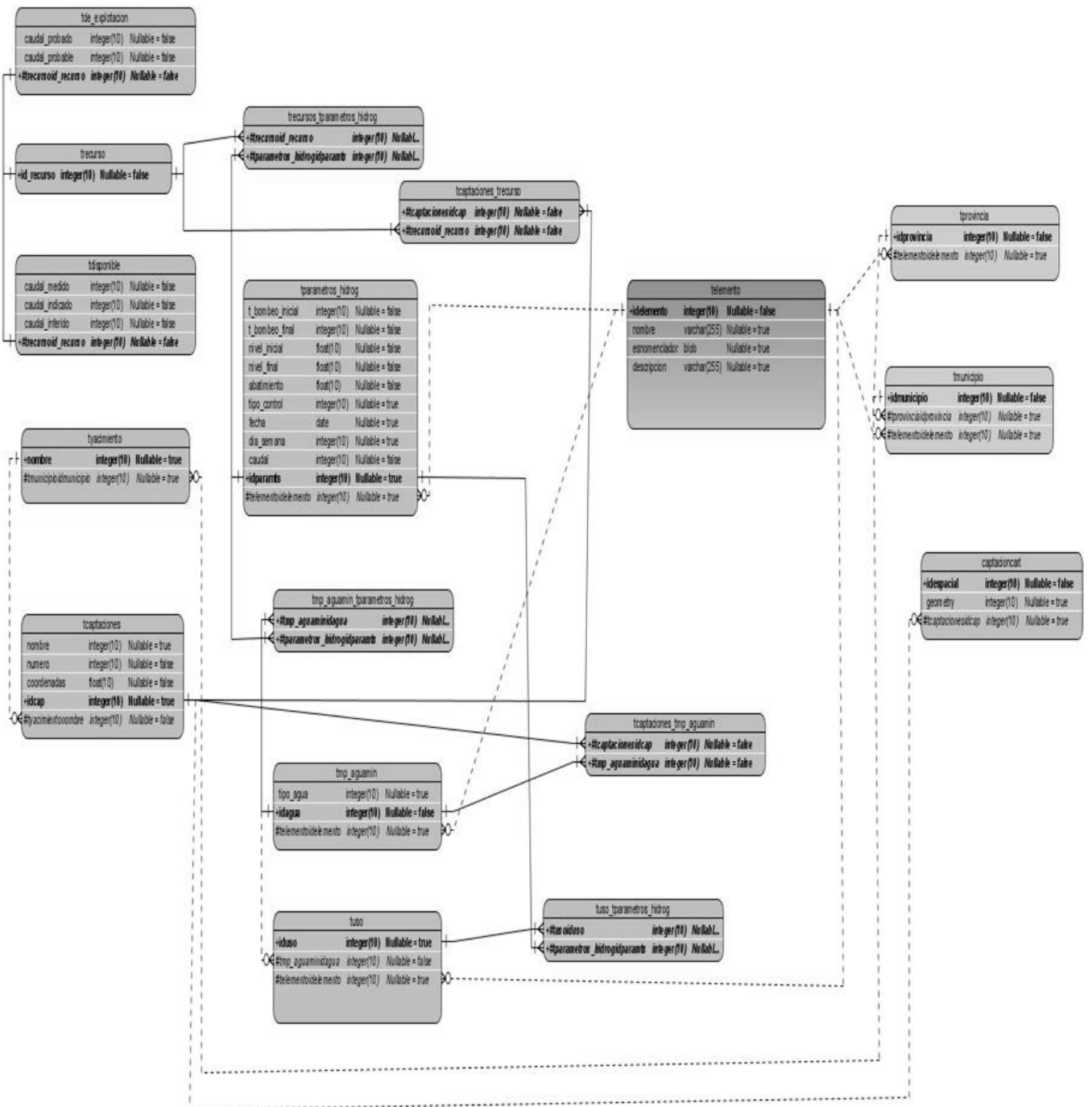


Figura 9 Modelo de Datos.

4.6 Diagrama de Despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes *hardware* y *software* en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes *software* (procesos y objetos que se ejecutan en ellos). Formados por instancias de los componentes *software* que representan manifestaciones del código en tiempo de ejecución (los componentes que sólo sean utilizados en tiempo de compilación deben mostrarse en el diagrama de componentes). A continuación se muestra el diagrama de despliegue:

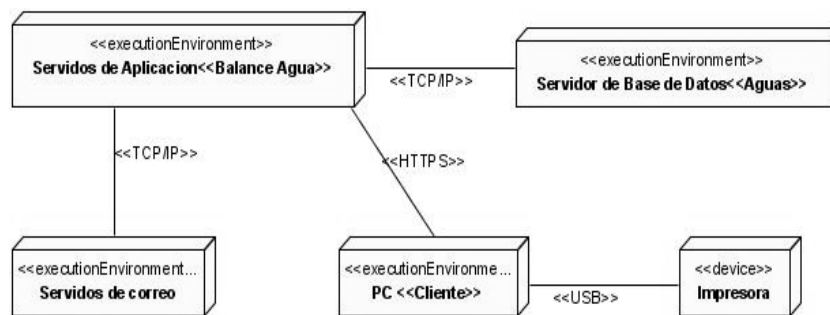


Figura 10 Diagrama de Despliegue.

4.7 Modelo de Implementación.

El modelo de implementación describe los elementos del modelo del diseño, como las clases se implementan en términos de componentes. Describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización, disponibles en el entorno de implementación y en el lenguaje utilizado, y cómo dependen los componentes unos de otros.

4.7.1 Diagrama de componentes.

El diagrama de componentes describe cómo interactúan los componentes físicos, o sea, muestran las dependencias lógicas que existe entre componentes software, ya sean ejecutables, de código fuente o binario, se utiliza para modelar la vista de implementación estática del sistema.

Éste diagrama es una representación más detallada de la descripción de requisitos no funcionales, tales como, las restricciones impuestas por el lenguaje de programación o las herramientas que se vayan a utilizar, en este caso la aplicación Web a desarrollar se implementará sobre Symfony, marco de trabajo (framework) basado en php5. Se modela un diagrama genérico dividido por paquetes,

donde en cada paquete se especifica las componentes que llevan dentro, en el caso del paquete Componentes Symfony sólo se representan los más significativos para el desarrollo de la aplicación sin tener en cuenta la totalidad con la que cuenta el framework. A continuación se muestra el Diagrama de Componentes genérico y seguidamente cada paquete con sus componentes dentro.

4.7.1.1 Diagrama de componente

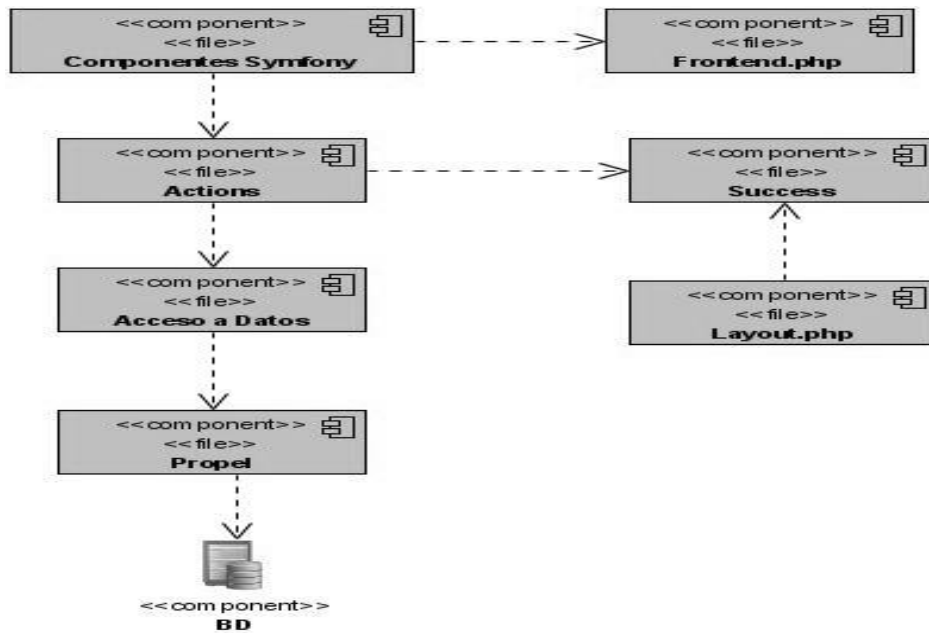


Figura 11 Diagrama de Componente.

4.7.1.2 Paquete Componente Symfony.

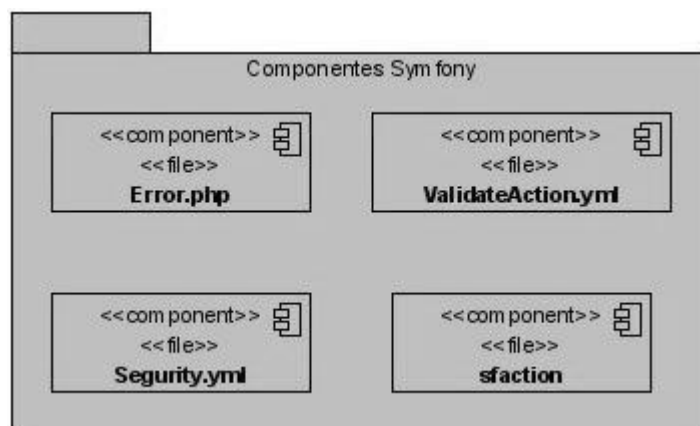


Figura 12 Paquete Componente Symfony.

4.7.1.3 Paquete Actions.

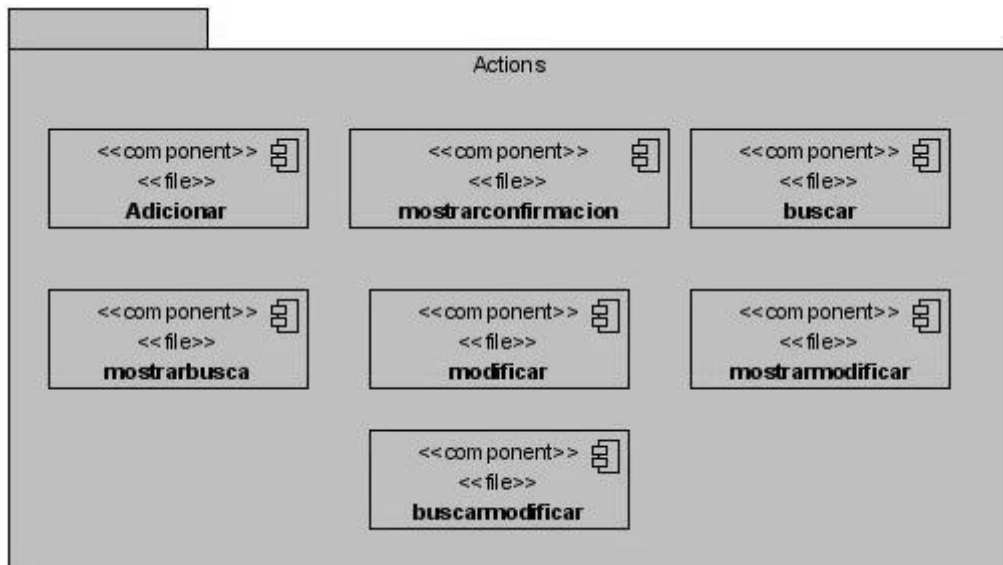


Figura 13 Paquete Actions.

4.7.1.4 Paquete Success.

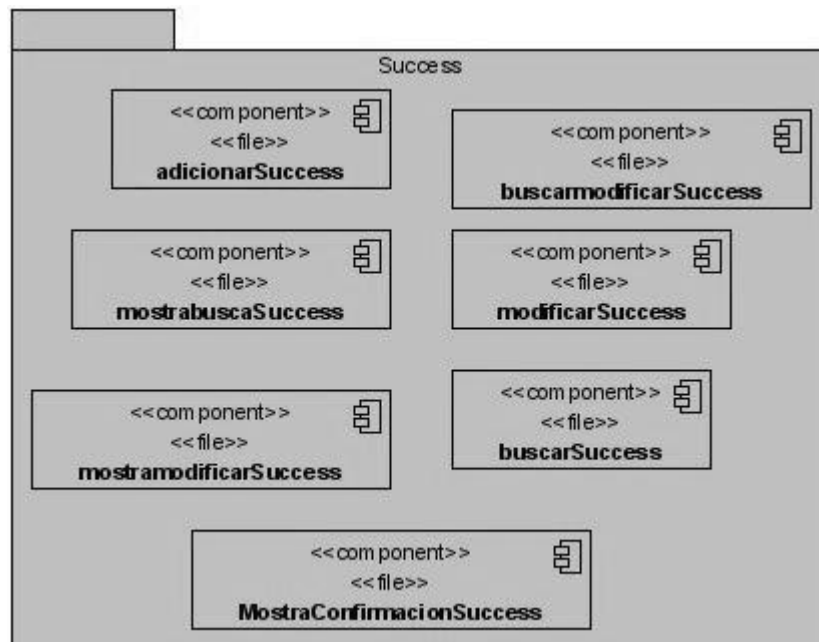


Figura 14 Paquete Success.

4.7.1.5 Paquete Acceso a Datos

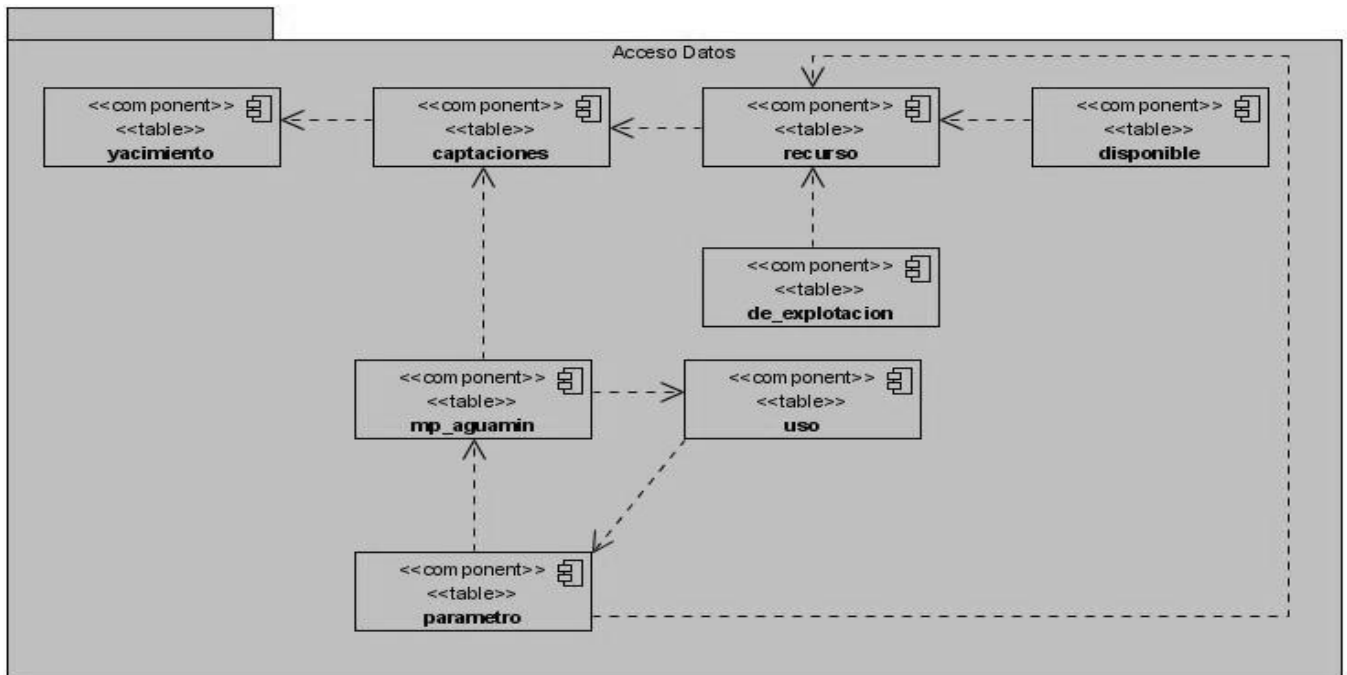


Figura 15 Paquete Acceso a Datos.

4.8 Conclusiones parciales.

En éste capítulo se han tratado aspectos esenciales que posibilitan el diseño del software con la calidad requerida. Se han presentado los elementos que permiten la construcción del sistema propuesto, ejemplo: los diagramas de clases del análisis, los diagramas de interacción, los diagramas de clases del diseño, los modelos de despliegue e implementación, así como la descripción de algunos aspectos de los estándares de la interfaz. De forma general, con el desarrollo de éste capítulo, se ha cumplido el objetivo propuesto: Diseñar una aplicación informática que permita la captura de información necesaria para el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales en la ONRM.

- Se logra eficiencia en la captura de información, eliminando errores humanos a la hora de realizar el balance de recursos disponibles y de explotación.
- Se satisfacen los deseos del usuario de mejorar el proceso para la captura de información.
- Se sientan las bases para una futura implementación del sistema.

Conclusiones Generales

Con la culminación del análisis y diseño para una aplicación que automatice el proceso de gestión para llevar a cabo el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales para la ONRM se llegan a las siguientes conclusiones:

- Se logró el objetivo general con la realización del análisis y el diseño para una aplicación Web que facilite la administración y gestión eficiente de los datos del Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales en la ONRM.
- 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 gestión de datos para el Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales.
- Con el empleo de los métodos teóricos y empíricos se logró conocer el estado del objeto de estudio.
- Se logró un diseño con la utilización de Symfony para una aplicación que gestione con más eficiencia el Balance Nacional de Recursos Disponibles y de Explotación.
- Se logró la modelación de la base de datos que será utilizada por la aplicación, a la cual tributan esquemas dentro de la Base de Datos General del Proyecto al cual esta asociado este trabajo de Diploma.
- La propuesta mejora en organización y rendimiento al actual proceso del Balance Nacional de Recursos y Reservas de la República de Cuba.

Recomendaciones

A partir de los resultados o beneficios que proporciona esta investigación, se recomienda:

- En Primer Lugar: Que este documento sea consultado por los futuros desarrolladores de la aplicación a la hora de implementarla.
- En Segundo Lugar: La implementación de la aplicación propuesta en este documento.
- En Tercer Lugar: Después de implementada la aplicación integrar funcionalidades con la aplicación de captura de la información de campo.

Referencias Bibliográficas

1. Sologren, A.L.P.e.I.B. *Utilización Terapéutica de las aguas y fangos mineromedicinales*. . 2007 [cited; Available from: http://www.bvs.sld.cu/revistas/far/vol36_1_02/far09102.htm. .
2. Romeo., M.D.C.D.L.R.J.y.M.Á.M. *Diversidad microbiana de las aguas minerales termales*. . 2007 [cited; Available from: <http://aguas.igme.es>.
3. Castillo., M.S.M.y.J.R.F. *Papel del agua mineral en el organismo y características terapéuticas de las aguas*. 2007 [cited; Available from: <http://www.sld.cu>.
4. MINBAS. 2008 [cited; Available from: http://www.cubagob.cu/des_eco/minbas.htm.
5. "Ley de Minas". . 1977., 23 de Enero de 1995 se publicó en la Gaceta Oficial.
6. *Soporte Técnico y Extensión Institucional del Sistema de Informatización de Aguas Subterráneas SIAGAS – Cuba.*, in *Granma*. 2008: Habana. p. Cuba.
7. LARMAN, C. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. 2007 [cited; Available from: <http://bibliodoc.uci.cu>.
8. RUMBAUHG, J.I.J., *El Proceso Unificado de Desarrollo de Software*. 2000.
9. MOLPECERES, A. (2003) *Procesos de desarrollo: RUP, XP y FDD, javaHispano*. **Volume**,
10. HERNÁNDEZ ORALLO, E. *El Lenguaje Unificado de Modelado (UML)*. 2007 [cited; Available from: <http://www.disca.upv.es>.
11. RUMBAUHG, J.I.J. *El Lenguaje Unificado de Modelado*. 2000 [cited.
12. *Soluciones racionales*. 2008 [cited; Available from: http://www.solucionesracionales.com/data_mod.html.
13. John Worsley, J.D., *PostgreSQL Práctico (Practical PostgreSQL)*. 2001.
14. Castilla, I.C.U.d., *Prácticas de la ingeniería de software, Una Herramienta CASE para ADOO*. 2007.
15. *Conferencia 1 Introducción a las Bases de Datos. Modelo Entidad-Relación (ER)*. 2008 [cited.
16. *postgresql Documentación*. 2008 [cited; Available from: http://www.netpecos.org/docs/mysql_postgres/x15.html.
17. *Lenguaje de programación*. 2007 [cited; Available from: http://enciclopedia.us.es/index.php/Lenguaje_de_programaci%C3%B3n.
18. *Aspectos y características generales* 2006.
19. Marsal, A.B., "Sistema de reservaciones de extensión universitaria." 2006, Universidad de las Ciencias Informática. p. 121 p.

20. Heredia, M.G. *10 Razones para Usar AJAX*. 2007 [cited; Available from: www.tufuncion.com/ventajas-ajax].
21. *Introduccion a Asp.Net*. 2008 [cited; Available from: http://www.gamarod.com.ar/articulos/introduccion_a_aspnet.asp].
22. *¿Que es PHP?* 2008 [cited; Available from: <http://www.adrformacion.com/cursos/php/leccion1/tutorial1.html>].
23. Hinostroza, R.R., *Características de PHP*. 2005.
24. Laura Bermejo Sanz, E.G.M. *Características principales, funcionalidad, utilización y caso práctico*. 2007 [cited].
25. *Zend_Studio*. 2008 [cited; Available from: <http://www.tufuncion.com/zend-studio>].
26. *Desarrollo Web*. 2008.
27. (2007) *Zend Studio 5 I LAS SOLUCIONES MÁS COMPLETAS PARA EL DESARROLLO DE PHP. Volume*,
28. Fabien Potencier, F.Z., *Symfony la Guia definitiva*. 2007.

Bibliografía Consultada

En español

1. AEURUS Diseño Web. Principios del diseño de interfaces de sitios Web. 2000-2008
[<http://www.aeurus.cl/articulos/2006/12/principios-del-diseno-de-interfaces-de-sitios-web.php>].
2. Principios Básicos De Diseño Para Web. 2008.
[<http://www.un.org/spanish/Depts/dpi/seminario/pdf/principiosfireworks.pdf>].
3. Roger S. Pressman. Ingeniería del Software. Un enfoque practico.
[<http://bibliodoc.uci.cu/pdf/reg02689.pdf>]
4. Ivar Jacobson, Grady Booch, James Rumbaugh. El Proceso Unificado de Desarrollo de Software.
[<http://bibliodoc.uci.cu/pdf/reg00060.pdf>]
5. Fabien Potencier, Francois Zaninotto. Symfony la guía definitiva. [www.librosweb.es/symfony]
6. Guía para el desarrollo de aplicaciones Web seguras. DesarrolloWeb.com, 2002. [2008].
[<http://www.desarrolloweb.com/articulos/996.php>].
7. ALVAREZ, S. Sistema gestor de base de datos, 2006. [2007].
[<http://www.desarrolloweb.com/faq/504.php>].
8. DORANTES GONZÁLEZ, V. H.; F. MAGARIÑOS LAMAS, et al. Curso de Bases de Datos y PostgreSQL, 2001. [2007]. [<http://es.tldp.org/Tutoriales/NOTAS-CURSOBDD/notas-curso-BD>]

En ingles.

1. Bruce Toqnazzini. First Principles of Interaction Design. 2006.
[<http://www.asktog.com/basics/firstPrinciples.html>]
2. WPDFD. Browser Grid. 2007 [<http://www.wpdfd.com/browsergrid.htm>]

3. Smashing Magazine. 10 Principles Of Effective Web Design. 2008.

[<http://www.smashingmagazine.com/2008/01/31/10-principles-of-effective-web-design/>]

4. EXTREME PROGRAMMING. Extreme Programming: A gentle introduction, 2006. [2007].

[<http://www.extremeprogramming.org>]

Glosario de Términos.

Término	Descripción
Balance	Es un resumen de todo lo que tiene o posee una la empresa, entidad u organismo para tener constancia de ello.
BNRDEAM	Balance Nacional de Recursos Disponibles y de Explotación de las Aguas Minerales.
Carratraca	Es un municipio de la provincia de Málaga, en España, situado entre las estribaciones más nororientales de la Serranía de Ronda (sierras de Alcaparaín, Baños y Aguas).
CASE	Ingeniería del software asistida por computadora.
Concesión	Es el otorgamiento del derecho de explotación por un lapso de tiempo.
Concesionarios:	Empresa o entidad que pide una concesión para determinados bienes y servicios.
CU	Caso de uso
Debian	Asociación o comunidad conformada por desarrolladores y usuarios que pretenden crear y mantener un sistema operativo GNU basado en software libre precompilado y empaquetado en un formato sencillo en múltiples arquitecturas y varios núcleos.
Esculapio	Hijo de Apolo y Coronis, que fue instruido por su padre en el arte de la medicina.
Etruscos	Los etruscos fueron un enigmático pueblo cuyo núcleo histórico fue la Toscana, a la cual dieron su nombre (eran llamados <i>Τυρσηνοί</i> (<i>tyrsenoi</i>) o <i>Τυρρηνοί</i> (<i>tyrrhenoi</i>) por los griegos y <i>tusci</i> o luego <i>etrusci</i> por los romanos; ellos se denominaban a sí mismos <i>rasena</i> o <i>rašna</i>).
Framework	Estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se pues considerar como una aplicación genérica incompleta y configurables a la que se puede añadir las últimas piezas para una aplicación concreta.
Gestión de los datos	Es la recolección, transferencia, seguridad y administración sistemática de los datos.
GNU	Conjunto de programas desarrollados por miembros de la fundación del software libre, de uso gratuito (FSF-Free Software Foundation).

GPL	Es una licencia creada por la Fundación de Software Libre y orientada principalmente a los términos de distribución, modificación y uso del software. Su propósito es declarar que el software cubierto por esta licencia es software libre (General Public License).
Linux	Sistema operativo multitarea, multiusuario, multiplataforma y multiprocesador, es una implementación de libre distribución UNIX para computadoras personales (PC), servidores y estaciones de trabajo.
Metodología	Del griego (metà "más allá" odòs "camino" logos "estudio"). Se refiere a los métodos de investigación que se siguen para alcanzar una gama de objetivos en una ciencia
Proceso	Un proceso (del latín processus) es un conjunto de actividades o eventos que se realizan o suceden (alternativa o simultáneamente) con un determinado fin.
Racional	Se llama acción racional a aquella en que el agente sopesa consistentemente los medios de los que dispone para alcanzar los fines que se ha propuesto, de modo que pueda lograrlos de la mejor manera posible
Recurso	En un sentido muy general, un recurso es un elemento que presenta una utilidad para las sociedades humanas. Se incluyen tradicionalmente en esta categoría la energía y las materias primas, pero también la información y las disponibilidades humanas.
Sistema	Un sistema (lat. <i>systema</i> , proveniente del griego <i>σύστημα</i>) es un conjunto ordenado de elementos cuyas propiedades se interrelacionan e interactúan de forma armónica entre sí. Estos conjuntos se denominan módulos. A su vez cada módulo puede ser un subsistema, dependiendo si sus propiedades son abiertas o cerradas.
Soporte	Nivel de soporte, dentro de la informática, representa la acción de solucionar problemas de una aplicación.
Vizcaya	Vizcaya (en euskera, Bizkaia) es una provincia de España y un territorio histórico de la comunidad autónoma del País Vasco

Anexos.

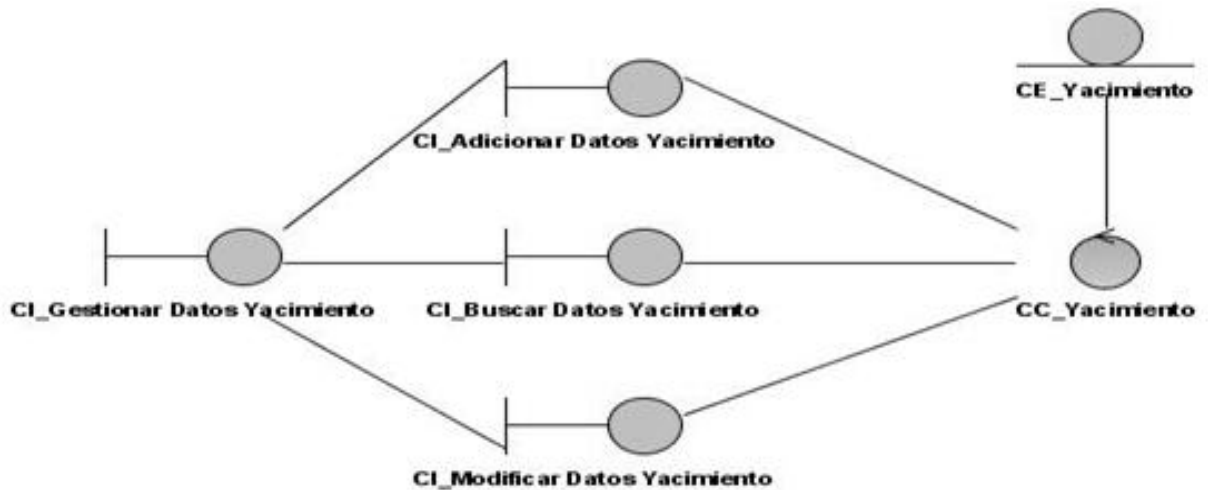
Anexo#1: Tabla que tributa al Balance Nacional de Recursos Disponibles y de Explotación.

a) Tabla Resumen General de los Yacimientos de Aguas Minerales

TABLA 1. TABLA RESUMEN GENERAL DE LOS YACIMIENTOS DE AGUAS MINERALES																										
Provincia.....		Municipio.....																								
Nombre del yacimiento.....										y de la Concesión.....							No de Expediente de la Concesión.....									
El Nombre del Representante legal.....												Firma														
Materia Prima.....																										
Número o Nombres de los Pozos o Manantiales en explotación.....																										
Recursos (Caudal) Disponibles. l/seg (m3/días).....																										
Recursos (Caudal) de Explotación l/seg (m ³ /días).....																										
Abatimiento Máximo Permitido (metro).....																										
SEC	POZO	l/seg - m3/días			MINE	TEMP.	MACROELEMENTOS								MICROELEMENTOS				MICROBIOLÓGICOS							
TOR		Medidos/	Indici/	Interidos	RALIZ.	PH	HCO ₃	SO ₄	CL	Ca	Na	K	Mg	H ₂ S	H ₂ SiO ₃	NO ₃	NO ₂	CT	CF	EF	SD					
		Probadas	Prob		(g/l)	°C	mg/l																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24			
I. RECURSOS DISPONIBLES IDENTIFICADOS (ODI)																										
II. REC. DE EXPLOTACION (ODE)																										
III. REC. EN EXPLOTACION (OEE)																										
IV. RESULTADOS DE LOS CONTROLES EXTERNOS																										
V. DESVIACIONES																										

Anexo#2: Diagramas de clases del análisis

a) Del CU_Gestionar Datos Yacimiento



Anexo#3: Diagramas de iteración del Análisis.

a) Del CU_Gestionar Datos Yacimiento, sección Adicionar Yacimiento.

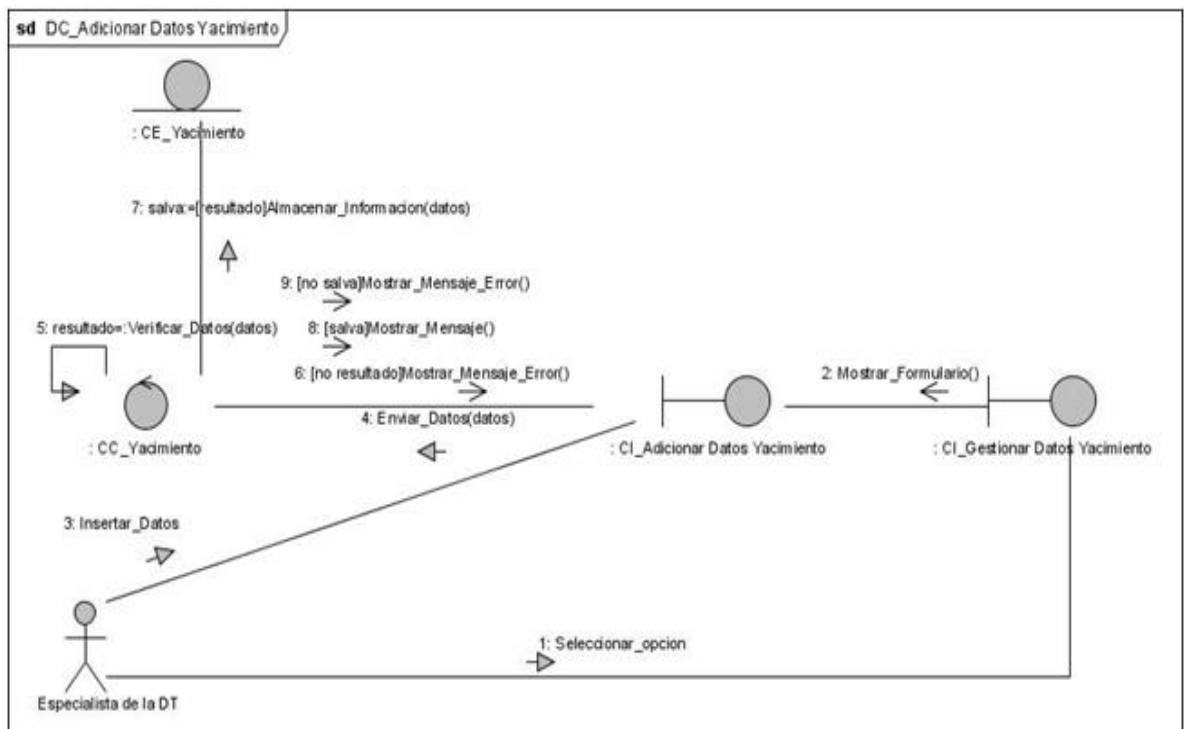


Figura 16 Diagrama de Interacción del Análisis, Sección Adicionar Yacimiento, CU_Gestionar Datos Yacimiento.

b) Del CU_Gestionar Datos Yacimiento, sección Modificar Yacimiento.

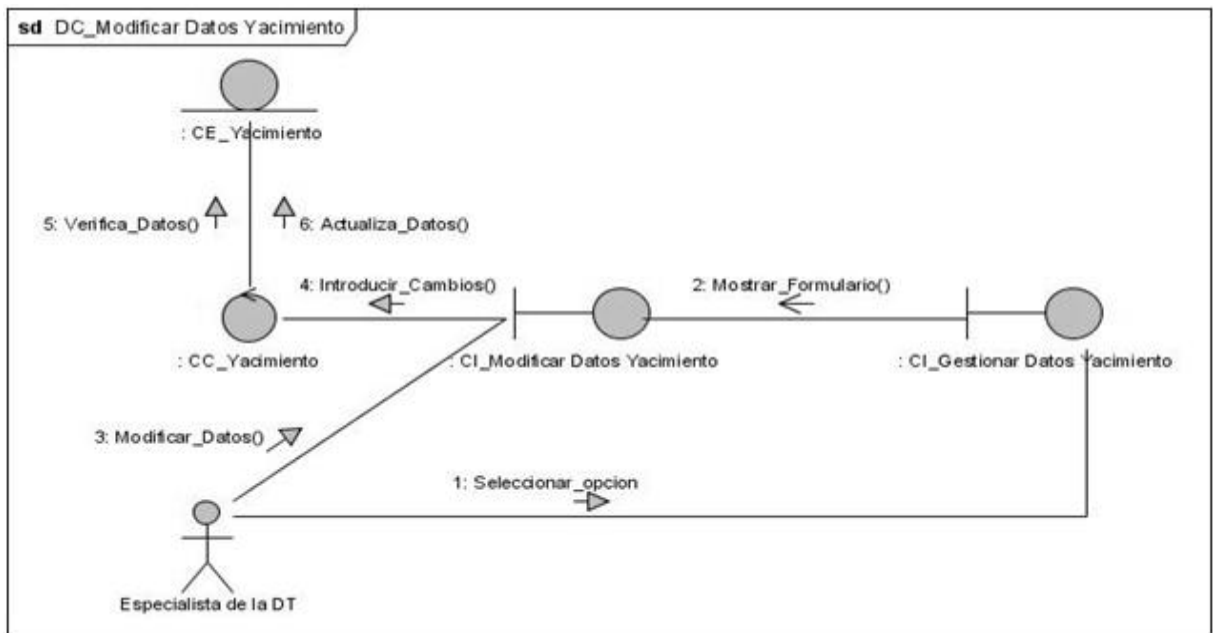


Figura 17 Diagrama de Interacción del Análisis, Sección Mostar Yacimiento, CU_Gestionar Datos Yacimiento.

c) Del CU_Gestionar Datos Yacimiento, sección Buscar Yacimiento.

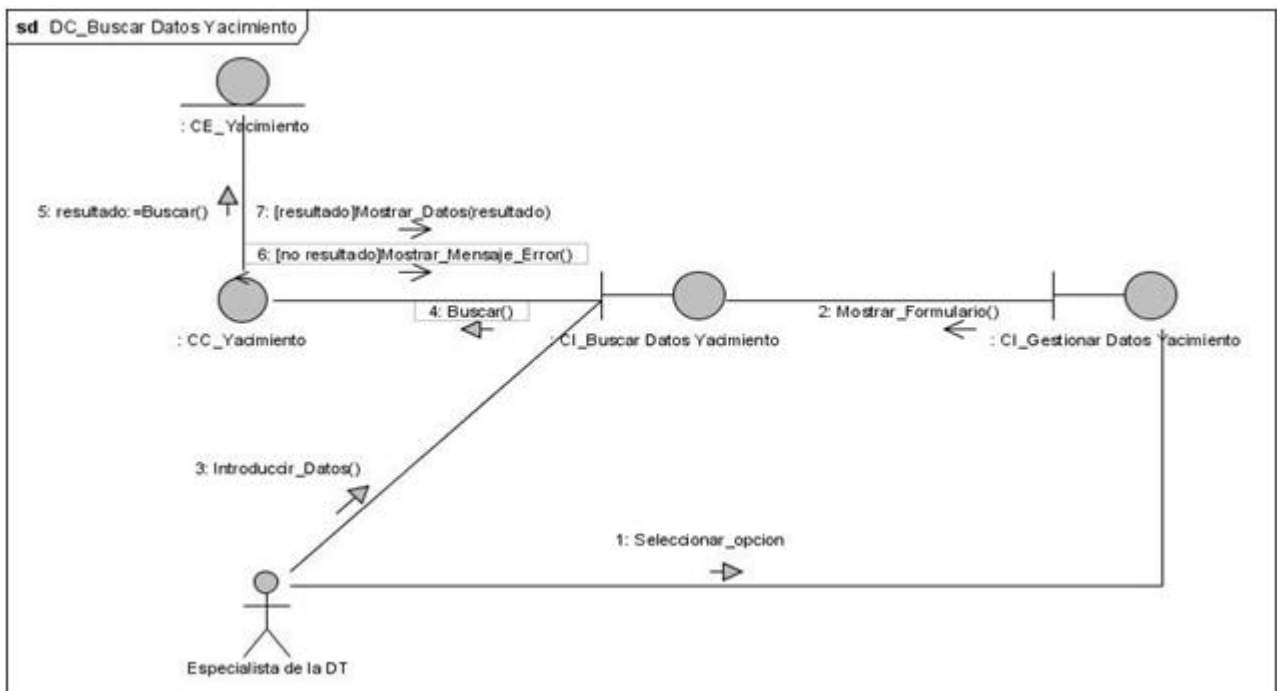


Figura 18 Diagrama de Interacción del Análisis, Sección Buscar Yacimiento, CU_Gestionar Datos Yacimiento.

Anexo#4: Diagramas del Diseño.

a) Del CU_Gestionar Datos Yacimiento. Sección Adicionar Datos Yacimiento.

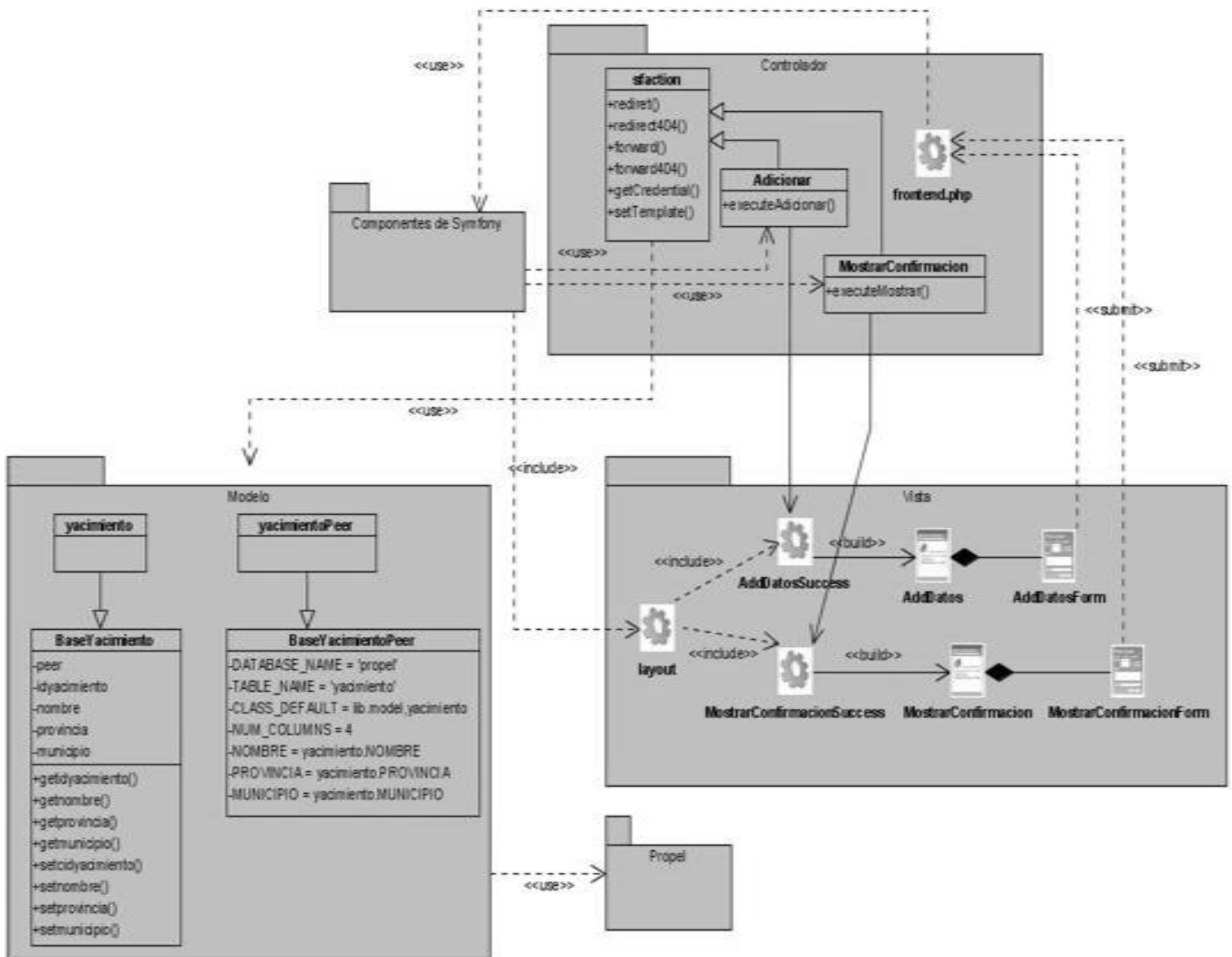


Figura 19 Diagrama del Diseño, Sección Adicionar Datos Yacimiento, CU_Gestionar Datos Yacimiento.

b) Del CU_Gestionar Datos Yacimiento. Sección Modificar Datos Yacimiento.

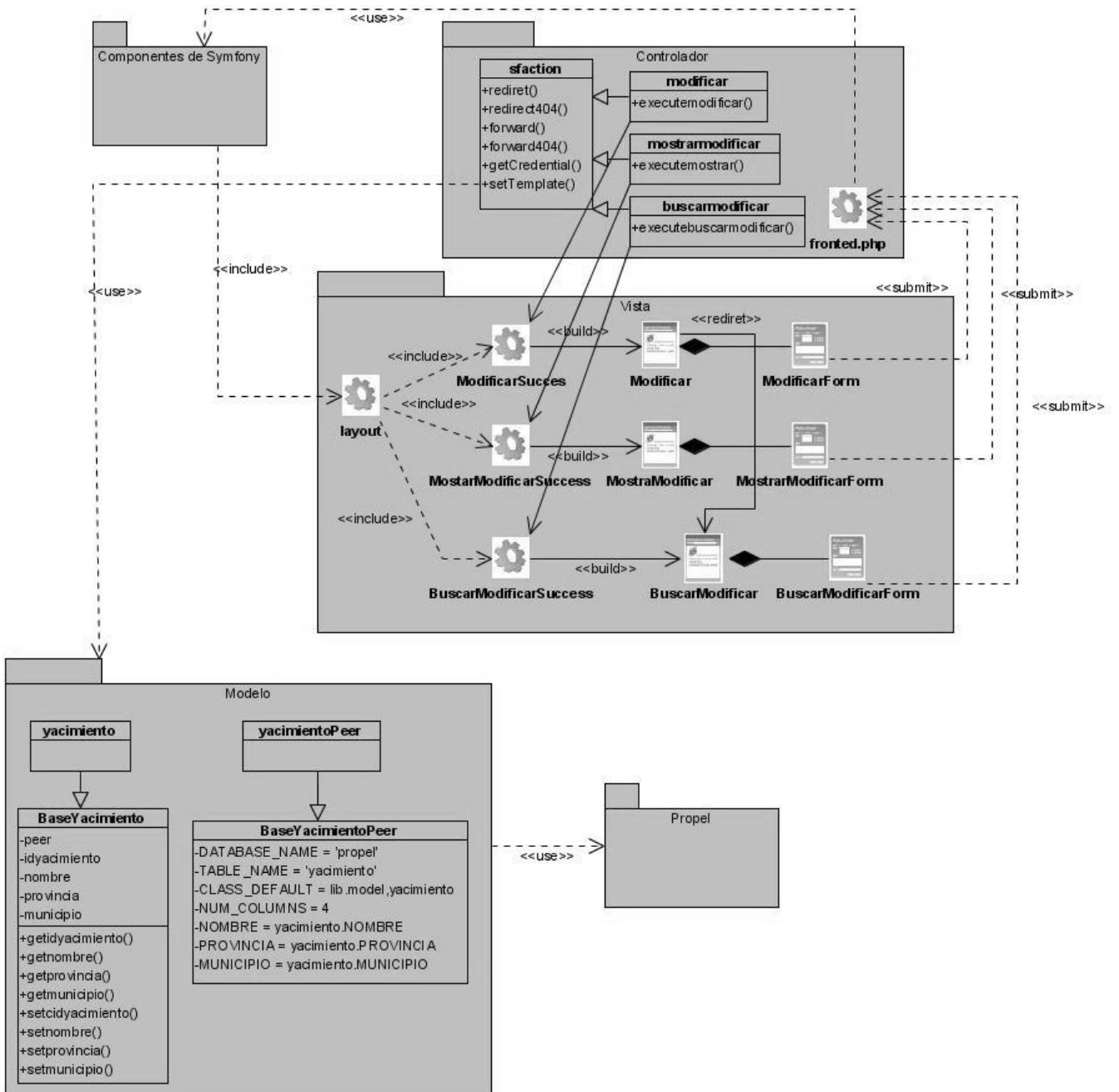


Figura 20 Diagrama del Diseño, Sección Modificar Datos Yacimiento, CU_Gestionar Datos Yacimiento.

c) Del CU_Gestionar Datos Yacimiento. Sección Buscar Datos Yacimiento.

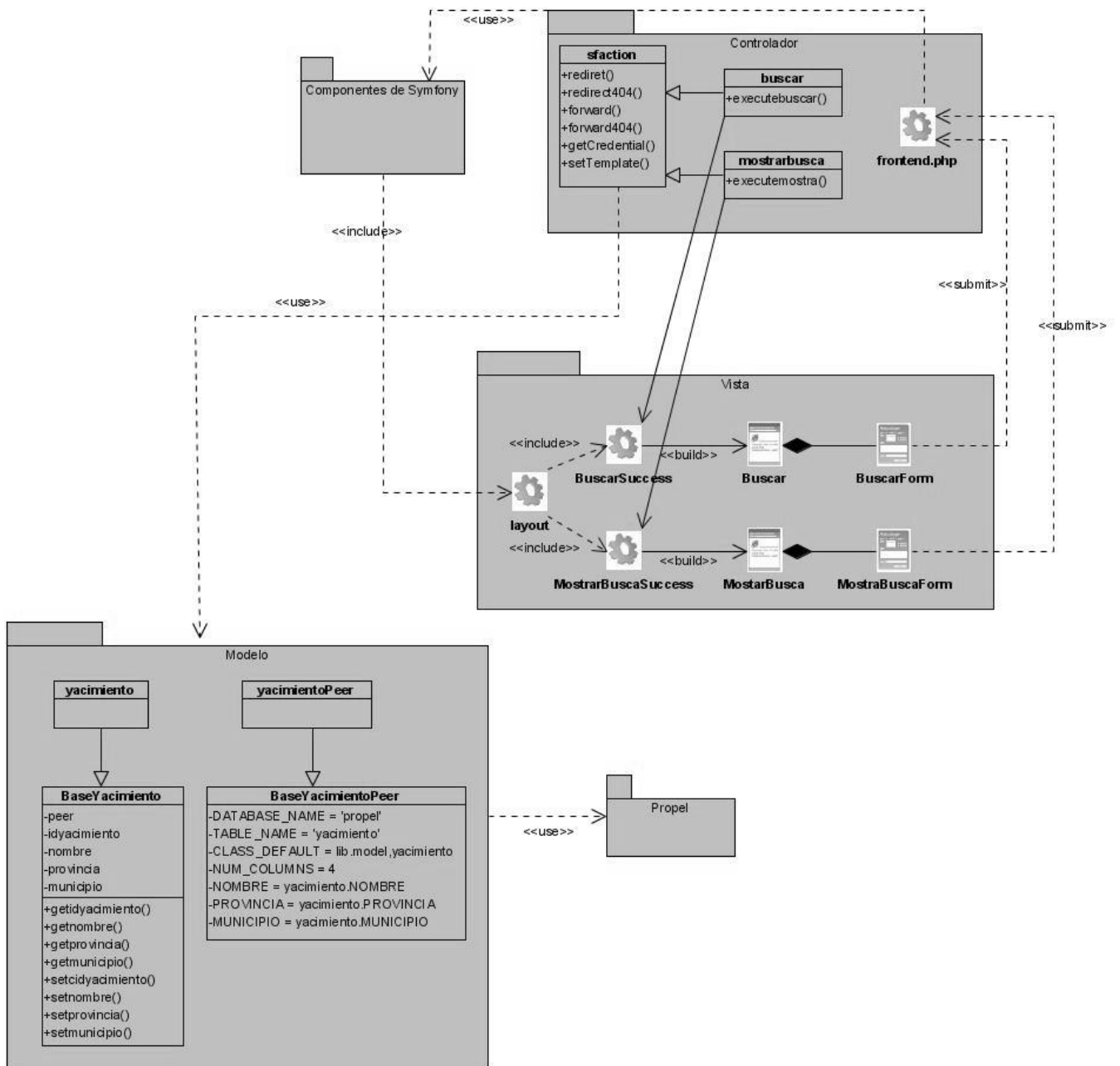


Figura 21 Diagrama del Diseño, Sección Buscar Datos Yacimiento, CU_Gestionar Datos Yacimiento.