

**Universidad de las Ciencias Informáticas**  
**Facultad 6**



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

**Título: Aplicación Web para la gestión de pozos de perforación.**

**Autora: Arianna Llorente Vargas**

**Tutor: Ing. Manuel Alejandro Pérez Rosabal**

La Habana, Junio/2013.

Año 55 de la Revolución.



*"Nuestra joven Revolución como una esponja, ávida de conocimientos (...) se impregnará en los anchos cauces de la ciencia socialista y de la solidaridad socialista para repartirla por el resto del continente americano..."*

*Ernesto Guevara de la Serna*

## DECLARACIÓN DE AUTORÍA

---

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Arianna Llorente Vargas

\_\_\_\_\_

Firma del Autor

Manuel Alejandro Pérez Rosabal

\_\_\_\_\_

Firma del Tutor

## DATOS DE CONTACTO

---

**Tutor:** Ing. Manuel Alejandro Pérez Rosabal.

**Formación Académica:** Ingeniero en Ciencias Informáticas (Julio/2012).

**Centro Laboral:** Universidad de las Ciencias Informáticas (UCI).

**Correo Electrónico:** [marosabal@uci.cu](mailto:marosabal@uci.cu)

## AGRADECIMIENTOS

---

*Quiero agradecer especialmente a:*

*Lo más grande de mi vida " Mi mamita: "Por su infinito amor, por su apoyo incondicional, por creer siempre en mí y darme ánimos de continuar cuando yo pensaba que ya no podía más, por ser mi luz, la que me sirvió y servirá de guía durante toda mi vida, por los sacrificios que ha hecho para que yo pudiera salir adelante y presenciar este momento de mi vida, gracias por existir y por ser parte de mi vida*

*Te Amo*

*A mi padrastro por ser un padre para mí, por apoyarme y creer también en mí, por sus consejos y por estar siempre ahí cuando lo necesito.*

*A mi hermanita Dianita y a mi primo Luismi por ser la motivación principal de llegar hasta aquí, al intentar darles un buen ejemplo a seguir y por ser de las personitas más importantes de mi vida.*

*A los mejores tíos del mundo, Chago, Susana, Mi Juguetico y César, por ser para ustedes una hija más, por su confianza, por darme la seguridad de que lograría graduarme y porque de una forma u otra me ayudaron hacer realidad mi sueño.*

*A mis abuelas Eraidita y Alicia por quererme tanto, por celebrar y sufrir cada paso que di, por ser tan buenas conmigo, por ayudarme y apoyarme siempre, las quiero mucho.*

## AGRADECIMIENTOS

---

*A mis primas Katy, Mercy y Cory, por estar siempre pendientes de mí y de mi carrera universitaria. Gracias por todo.*

*A Grecher y Gressy, por ser mi segunda familia en La Habana, por estar siempre presentes cuando las necesité. Muchas Gracias por todo.*

*A todas las amistades ganadas en estos años por haber sido como una familia grande aquí en la Universidad, por estar en los buenos y malos momentos y porque a pesar de la distancia los llevaré siempre en mi corazón: Katherine, Lisandra Mercedes (bruji), Yeineri, Jessica, Alexaidis, Grethell, Melixa, Aliuska, Alejandro, Yuliet, Thaimí, Rebeca, Wilber y Ángel.*

*Los Quiero.*

*A los profesores del proyecto Sistema Minero Cubano por sus críticas y sugerencias para poder lograr mi meta: Eddy, Dago, Maikel, Manuel, Milenis y Armando.*

*A todos los que de una forma u otra han contribuido a mi formación como profesional y humano en general.*

*Dedico de manera muy especial este logro a mi mamá Oralia, por ser mi guía y apoyo, por ser quien me impulsa día a día. Te Amo.*

*A mi padrastro Alexis por ser un padre para mí.*

*A mis abuelas Eraidá y Alicia.*

*A mis tíos queridos Susana, Chago, Pirolo y César.*

*A mis niñitos lindos Dianita y LuisMi.*

*A mis primas Katy, Mecho y Cory.*

*A la memoria de mi Ángel de la guarda, mi abuelito Rafael (papi), por todo el amor que me brindó durante su vida y porque sé que si estuviera aquí estaría muy orgulloso de mí.*

### **Resumen**

Las diversas empresas mineras existentes en el país han desarrollado la explotación de los recursos minerales a través de años de estudios e investigaciones. Los procesos y tareas que se realizan en dichas empresas para gestionar la información de los pozos de perforación son complejos, pues arrojan un elevado volumen de información, lo que conlleva a problemas de planificación y explotación si los datos son incorrectos, trayendo como consecuencia pérdidas monetarias en este sector. El trabajo que a continuación se expone, propone como solución el desarrollo de una Aplicación Web para la gestión de pozos de perforación asociado a un yacimiento mineral. Para la construcción de la aplicación propuesta se hace necesario realizar un exhaustivo estudio de los distintos procesos que rodean a las entidades mineras cubanas, además de escogerse un conjunto de herramientas y tecnologías orientadas al *software* libre, lo cual ratifica la independencia tecnológica que lleva a cabo el país. Dicha herramienta informática tiene como objetivo ofrecer seguridad, disponibilidad y centralización de los datos, además de ahorro considerable de tiempo y esfuerzo para estas entidades mineras, beneficiándolas de esta forma en el desempeño de sus actividades.

### **Palabras Claves:**

Aplicación Web, explotación, gestión, planificación, pozos de perforación, *software* libre.



## Índice

Resumen.....	VIII
Índice de Figuras .....	XII
Índice de Tablas.....	XIII
Introducción.....	1
1. Capítulo 1: Fundamentos teóricos de la Investigación.....	6
1.1 Conceptos Asociados al dominio del problema.....	6
1.2 Objeto de estudio. ....	8
1.2.1 Descripción general. ....	8
1.2.2 Descripción del dominio del problema. ....	10
1.2.3 Situación problemática. ....	10
1.3 Análisis de soluciones existentes. ....	11
1.4 Conclusiones Parciales.....	12
2. Capítulo 2: Tecnologías, metodologías y herramientas.....	14
2.1 Arquitectura de Software.....	14
2.2 Estilo Arquitectónico. Modelo Vista Controlador. ....	15
2.3 Metodología de desarrollo.....	16
2.4 Lenguaje de Modelado. UML 2.1.1 .....	17
2.5 Herramienta para el modelado: Visual Paradigm 8.0.....	18
2.6 Lenguaje de programación utilizados .....	19
2.7 Sistema Gestor de Bases de Datos.....	20
2.8 Servidor Web. Servidor Apache 2.0.....	21
2.9 Framework.....	22
2.10 NetBeans 7.2.1 como entorno de desarrollo integrado (IDE). ....	23

2.11	Conclusiones Parciales .....	24
3.	Capítulo 3: Características del Sistema.....	25
3.1	Propuesta del Sistema .....	25
3.2	Modelo de Dominio.....	25
3.2.1	Glosario de términos del dominio.....	26
3.3	Modelo del Sistema.....	27
3.3.1	Requisitos funcionales.....	27
3.3.2	Requisitos no funcionales .....	28
3.3.3	Diagrama de casos de uso del sistema.....	30
3.3	Conclusiones Parciales.....	34
4.	Capítulo 4. Diseño, Implementación y Pruebas de la solución propuesta.....	35
4.1	Análisis y Diseño .....	35
4.2	Modelo de Diseño.....	35
4.3	Patrones de diseño utilizados .....	36
4.4	Diagramas de Interacción .....	<b>¡Error! Marcador no definido.</b>
4.5	Diagramas de clases del diseño.....	36
4.6	Modelo de despliegue .....	38
4.7	Diagrama de Componentes .....	39
4.8	Modelo de Prueba .....	40
4.9	Conclusiones Parciales.....	44
	Conclusiones Generales.....	45
	Recomendaciones .....	46
	Bibliografía Referenciada .....	47
	Bibliografía .....	50

Anexos ..... 53

### Índice de Figuras

Figura 1. Zona Mineral con varios pozos de perforación.....	3
Figura 2. Etapas de la Minería .....	9
Figura 3. Modelo Cliente/Servidor ( <a href="http://t1.gstatic.com/">http://t1.gstatic.com/</a> ).....	15
Figura 4. Fases e Iteraciones de la Metodología RUP( <a href="http://jummp.wordpress.com">http://jummp.wordpress.com</a> ).....	17
Figura 5. Diagrama del Modelo de Dominio.....	26
Figura 6. Diagrama de caso de uso del sistema.....	30
Figura 7. Diagrama de clases del diseño para el CU Establecer Conexión.....	37
Figura 8. Diagrama de despliegue. ....	38
Figura 9. Diagrama de componentes de la aplicación web.....	40
Figura 10. Diagrama de clases del diseño para el CU Gestionar datos Tablas. ....	62
Figura 11. Diagrama de clases del diseño para el CU Calcular Coordenadas.....	62
Figura 12. Diagrama de clases del diseño para el CU Refrescar Información .....	63

### Índice de Tablas

Tabla 1. Descripción de los actores del sistema.....	31
Tabla 2. Descripción del caso de uso “Establecer Conexión” .....	31
Tabla 3. Escenarios del caso de uso Establecer conexión .....	41
Tabla 4. Variables del caso de uso Establecer conexión .....	42
Tabla 5. Matriz del caso de uso Establecer conexión.....	43
Tabla 6. Descripción del caso de uso Gestionar datos tabla .....	53
Tabla 7. Descripción del caso de uso Calcular Coordenadas .....	54
Tabla 8. Descripción del caso de uso Validar Datos .....	55
Tabla 9. Descripción del caso de uso Filtrar datos .....	56
Tabla 10. Descripción del caso de uso Identificar Tablas .....	59
Tabla 11. Descripción del caso de uso Refrescar Información .....	60

## Introducción

En la historia de la humanidad, el hombre ha procurado garantizar y mejorar su nivel de vida mediante un mejor conocimiento del mundo que le rodea y un dominio más eficaz del mismo, es decir, mediante el desarrollo constante de las ciencias y las técnicas. La tecnología existente hoy en día, es un medio indispensable para el desarrollo de la sociedad, aplicada en diferentes sectores en todo el mundo. Esta revolución tecnológica que dio paso a la era de la computación, trajo consigo la rápida evolución de la informática, logrando reducir el tiempo para procesar, archivar y recuperar grandes volúmenes de datos.

En la actualidad, el sector de la industria minera es una de las ramas fundamentales de la economía a nivel mundial. En esta esfera se incorporan nuevos modelos basados en las Tecnologías de la Información y las Comunicaciones (TIC) para su informatización, dando paso a la utilización de un conjunto de sistemas especialmente diseñados para la planificación de la explotación minera.

Cuba ha tratado de mantenerse en contacto directo con las nuevas tecnologías nacientes relacionadas con la minería y con todos los procesos por los que ha ido transitando la informática a través de los años en esta rama, para llevar el control y manejo de enormes volúmenes de información, a pesar de todas las barreras que le impone el bloqueo y su estatus de país subdesarrollado. Dichas tecnologías se han ido incluyendo para suplantar procedimientos manuales, aumentar la calidad y disminuir los costos en diferentes áreas del sector, tratando muchas veces con productos de *software* propietarios, tales como DATAMINE, GEMCOM, VULCAN, SURPAC y GEOSTAT, sobresaliendo así la dependencia tecnológica de las empresas hacia las grandes compañías privadas desarrolladoras y comercializadoras de los sistemas de *software* mineros. (1)

Los precios de estos productos son muy costosos lo que imposibilita a las empresas de menor desarrollo el acceso a sus licencias o actualizaciones. Esto le trajo al país la necesidad de desarrollar herramientas como: TIERRA, implementada a partir de la colaboración conjunta del Instituto Superior Minero Metalúrgico de Moa (ISMMM), CUBANÍQUEL y el Sistema Integral Minero (SIM), realizada en la empresa mixta Pedro Soto Alba<sup>1</sup>, y otros como CORTE y MICRONIQ. Como consecuencia de la utilización de este último quedó una completa Base de Datos (BD) de la información geológica, siguiendo el desarrollo

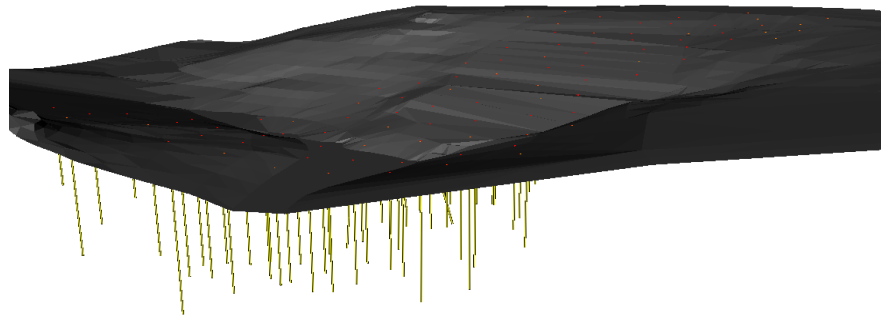
---

<sup>1</sup> Pedro Soto Alba: Es la tercera planta procesadora y productora de mineral creada en el Municipio de Moa, Provincia de Holguín.

sistemático alcanzado en esta etapa. En esta BD se expresa la esencia de los objetivos a lograr en la automatización del tratamiento de la información en la minería.

Estas herramientas de carácter nacional presentan diversas dificultades entre las que se encuentran: no han podido ser completadas pues no cuentan con una etapa inicial que realice el estudio de los procesos involucrados en la captura y procesamiento de la información obtenida durante la actividad minera al aplicar los métodos de prospección y exploración, pozos de sondeos. No posibilitan la determinación del tamaño y forma del yacimiento, así como del contenido y calidad del mineral existente, ni facilitan la búsqueda de indicios de nuevas áreas mineralizadas, no cuentan con una documentación para el uso de los mismos, lo que hace difícil la realización de un buen trabajo en esta actividad, y están desarrollados principalmente para un área determinada y no para la actividad minera en general. (2)

En la industria nacional, el trabajo con grandes volúmenes de datos obtenidos de los pozos de perforación en las etapas de exploración y prospección, tales como las coordenadas donde se encuentran, su profundidad, la concentración de mineral y las muestras contenidas, están expuestos a numerosos tipos de errores entre los que se encuentran: la existencia de intervalos repetidos, solapados (no consecutivos) en las muestras y litologías de los pozos o que los intervalos de las muestras no estén dentro de la profundidad del pozo al que pertenecen. Estos problemas pueden traer asociados pérdidas considerables a la planificación económica de los proyectos mineros ya que conllevarían a la toma de decisiones erróneas. Como ejemplo de estas decisiones erróneas tenemos, la realización de excavaciones en un terreno donde se piensa que existe un gran volumen de un mineral determinado y en la práctica se evidencia luego que no existe o que es menor que el valor esperado, y por tanto la extracción del mineral reporta pérdidas monetarias (Ver Figura 1). Por esta razón es muy importante la gestión, análisis y validación de los datos, procesos que se realizan de forma manual, resultando complejo y monótono como consecuencia del alto volumen de información que se maneja y de la diversidad de errores que pueden presentar; lo cual puede retardar la ejecución de los proyectos. También es importante destacar que los proyectos mineros suelen ser complejos, por lo que en un proyecto trabajan varios especialistas con la misma información; sin embargo las actualizaciones o correcciones que estos realizan en sus puestos de trabajo no son visibles para los demás.



**Figura 1. Zona Mineral con varios pozos de perforación**

Teniendo en cuenta la situación problemática expuesta anteriormente se identifica el siguiente **problema a resolver**: ¿Cómo facilitar la gestión de los datos geológicos mineros de los pozos de perforación en empresas mineras cubanas?

Determinándose como **objeto de estudio**: El proceso de gestión de los datos geológicos mineros en yacimientos minerales.

Derivándose como **campo de acción**: El proceso de gestión de los datos geológicos mineros obtenidos en las etapas de prospección y exploración de yacimientos minerales.

Se precisa la siguiente **idea a defender** para la investigación científica: Si se desarrolla un sistema informático que gestione los datos geológicos mineros de los pozos de perforación facilitará el manejo de la información en las empresas mineras cubanas.

El **objetivo general** de la investigación es: Desarrollar un sistema informático capaz de gestionar los datos geológicos mineros de un yacimiento mineral en empresas mineras cubanas.

Para dar cumplimiento al objetivo general propuesto y darle solución a la situación problemática planteada se trazan un grupo de **tareas investigativas**:

1. Caracterización de las tecnologías que son aplicadas para resolver los problemas de gestión y manipulación de los pozos de perforación.
2. Caracterización de la metodología y las tecnologías a utilizar en el desarrollo de la solución.



3. Modelación de los procesos del negocio, análisis y diseño.
4. Implementación de la solución propuesta.
5. Validación de la solución propuesta mediante pruebas de *software*.

A lo largo del desarrollo de la investigación se utilizan un conjunto de **métodos científicos (teóricos y empíricos)** entre los que se encuentran:

### **Métodos teóricos:**

- **Histórico-Lógico:** Permite consultar la bibliografía necesaria para dar cumplimiento a las tareas de la investigación y resumir los aspectos fundamentales de cada una de ellas.
- **Analítico-Sintético:** Se utilizó para el estudio de la bibliografía, conformada por documentos, sitios Web y tesis, para realizar la fundamentación teórica de la investigación y llegar a conclusiones sobre los conceptos fundamentales que sean necesarios para la solución del problema.
- **Modelación:** Se emplea para representar con artefactos ingenieriles los procesos de gestión de la información de los pozos de perforación facilitando un mejor entendimiento de la solución a implementar.

### **Métodos empíricos:**

- **Observación:** Se utiliza para entender el trabajo que se realiza en el proyecto y cómo debe funcionar el sistema. Como técnica se utiliza la observación no participante o externa, donde se propone observar desde afuera los procesos que tienen lugar en el proyecto, sin intervenir en su curso.

### **Resultados esperados:**

1. Una aplicación Web para la gestión de pozos de perforación.
2. La documentación técnica correspondiente a la aplicación.

El trabajo consta de introducción, cuatro capítulos, conclusiones, recomendaciones, bibliografía y anexos.

### **Capítulo 1. Fundamentos Teóricos de la Investigación.**

Se expone la descripción del entorno donde se encuentra el negocio y su organización, se describe detalladamente la situación problemática y el análisis de otras soluciones que puedan brindar respuesta al problema de la investigación planteado en el presente trabajo.

## **Capítulo 2. Tecnologías, metodología y herramientas.**

Se define la arquitectura propuesta para el sistema a desarrollar, principales tecnologías existentes que se emplean para llegar a la solución, así como la descripción de las características del marco de trabajo, las principales herramientas y la metodología de desarrollo de *software* que se utiliza.

## **Capítulo 3. Características del Sistema.**

Se realiza la descripción de los principales artefactos relacionados con la modelación del dominio, especificando los actores, casos de uso, descripciones y diagramas correspondientes, así como los requisitos propios del sistema, tanto funcionales como no funcionales, según indica la metodología seleccionada para guiar el proceso de desarrollo.

## **Capítulo 4. Diseño, Implementación y Pruebas de la solución propuesta.**

Se desarrollan los artefactos del Diseño e Implementación, dígase modelo de despliegue, diagrama de clases del diseño, diagrama de componente y diagrama de secuencia. Para finalizar en el capítulo se realizan las pruebas pertinentes al *software*.

## **1. Capítulo 1: Fundamentos teóricos de la Investigación.**

En el presente capítulo se abordan los temas relacionados con la fundamentación teórica de la investigación, se describen los conceptos asociados al dominio del problema, así como la descripción del objeto de estudio, referente a la gestión de la información de los pozos de perforación en las empresas mineras cubanas. Además, se realiza un análisis de las soluciones existentes tanto a nivel nacional como internacional y se identifican los principales problemas que motivaron el desarrollo de la investigación.

### **1.1 Conceptos Asociados al dominio del problema.**

Existen un conjunto de conceptos que se relacionan con la investigación y el problema planteado, estrechamente relacionados con la minería, los cuales se describirán a continuación para una mejor comprensión del trabajo en general.

#### **Geología**

La Geología es una ciencia multifacética, la cual tiene por objeto investigar la composición, estructura, origen y evolución de la tierra. Su estudio abarca el proceso de la formación de las montañas, las rocas, los minerales, los restos orgánicos fosilizados. (3)

#### **Minería**

La Minería es una actividad económica del sector primario representada por la explotación o extracción de los minerales que se han acumulado en el suelo y subsuelo en forma de yacimiento. (4)

#### **Yacimiento Mineral**

Un Yacimiento Mineral es cualquier acumulación natural de sustancias minerales en el suelo o en el subsuelo, que pueda ser utilizado y explotado como fuente de materia prima y como fuente de energía, y las concentraciones de piedras preciosas y semipreciosas y de cualquier otra sustancia mineral cuya explotación tenga importancia económica. (5)

#### **Pozos de perforación**

Se denomina pozos de perforación al conjunto de excavaciones de sección redonda en las rocas. (6)

#### **Datos Geológicos**

Los datos geológicos son aquella fuente de toda la información básica geológica y minera disponible en la actualidad. Estos surgen como resultado de profundos estudios realizados por especialistas en la materia (geólogos), que rigurosamente buscan y evalúan recursos minerales, hídricos o de combustibles fósiles, como el carbón y el petróleo. Los datos geológicos, en su mayoría, constituyen datos geoespaciales. (7)

### **Gestión de datos de pozos de perforación**

Este proceso ayuda a los exploradores a trabajar con rapidez e inteligencia al organizar y compartir datos e información entre equipos de manera que puedan darle sentido a grandes cantidades de datos digitales para la toma de decisiones en exploración. (8)

- Organiza y cataloga conjuntos de datos.
- Crea y administra datos.
- Proporciona vistas previas de los datos.
- Administra información y envío de datos de actividades de exploración. (8)

### **Aplicación Web**

Una Aplicación Web es una aplicación informática que los usuarios pueden utilizar accediendo a un servidor Web a través de Internet o de una intranet mediante un navegador. Son soluciones informáticas que nos permiten interactuar con la información, actualizarla o modificarla, según los privilegios de seguridad que posea el usuario, dentro de la aplicación.

Las Aplicaciones Web poseen diferentes ventajas entre las cuales se pueden encontrar:

- Compatibilidad Multiplataforma: Las diferentes tecnologías utilizadas para el desarrollo de aplicaciones Web como PHP, Java, AJAX, JavaScript, MySQL, PostgreSQL, SQLServer y Oracle permiten un desarrollo efectivo y robusto de las mismas, por lo cual pueden ser ejecutadas en los principales sistemas operativos sin ninguna restricción, a diferencia del *software* de escritorio en donde debe existir una versión instalable diferente para cada uno de ellos. (9)
- Menos requerimientos de hardware: Las aplicaciones Web consumen menos recursos *hardware* de un computador que los programas instalados localmente, en cuanto a memoria RAM<sup>2</sup> y espacio en disco duro. Esto se debe a que las aplicaciones se alojan y ejecutan en los servidores del proveedor o de la

---

<sup>2</sup> RAM: Memoria de Acceso Aleatorio

empresa, dejando más espacio para correr múltiples aplicaciones al mismo tiempo sin deteriorar el rendimiento de los equipos del usuario final. (9)

- Acceso inmediato: Dado que las aplicaciones Web no necesitan ser descargadas, instaladas y configuradas en cada uno de los equipos, un usuario puede acceder de forma remota mediante un navegador Web y comenzar a trabajar sin importar la configuración y el *hardware* de su equipo. (9)
- Múltiples usuarios concurrentes: Las aplicaciones Web pueden ser utilizadas por múltiples usuarios al mismo tiempo, con lo cual diferentes usuarios pueden ver e incluso editar toda la información de manera conjunta. (9)
- Información segura: Dado que la información se almacena de forma centralizada y a las copias de seguridad de los servidores, los usuarios van a tener mucho menos riesgo de perder sus datos ocasionado por un daño en sus discos duros o un virus en su computador. (9)
- Menos errores y fallas: Las aplicaciones Web son menos propensas a crear problemas técnicos debido a conflictos con el *software*, *hardware* o con otras aplicaciones existentes. Con las aplicaciones Web todos los usuarios utilizan la misma versión y por ende las posibles fallas pueden ser corregidas tan pronto como sean descubiertas, y ningún usuario habrá actualizado individualmente la aplicación. (9)
- Facilidad de actualización: Cuando una aplicación Web debe ser actualizada este procedimiento se realiza una vez para todos los usuarios, por lo cual no es necesario que cada usuario tome acciones pro-activas que interfieran con sus hábitos de trabajo al iniciar nuevas descargas y procedimientos de instalación individuales. (9)

### 1.2 Objeto de estudio.

#### 1.2.1 Descripción general.

Se acostumbra decir que la Minería es la madre de todas las industrias, por ser ella quien aporta las materias primas que las demás actividades procesan para cubrir las necesidades planteadas por el desarrollo de la Humanidad. De esta industria madre, su primer paso lo constituye la Exploración, esa parte del proceso cuyo objetivo es encontrar y evaluar los yacimientos de minerales útiles para satisfacer la demanda de la sociedad.

Según la Ley de Minas publicada en la Gaceta Oficial de la República de Cuba el 23 de enero de 1995, se divide la actividad minera en las siguientes fases:

- a) Reconocimiento.
- b) Investigación Geológica: que se divide en las subfases Prospección y Exploración.
- c) Explotación.
- d) Procesamiento.
- e) Comercialización.



**Figura 2. Etapas de la Minería**

El proceso inicial en la búsqueda de un yacimiento mineral, consiste en realizar la prospección, que no es más que el conjunto de trabajos con empleo de técnicas y métodos que tienen como objetivo la búsqueda de indicios y concentraciones minerales que pudieran constituir un yacimiento. Luego de cumplir la etapa de prospección inicia la etapa de exploración. Este proceso está enfocado en realizar un reconocimiento detallado del depósito mineral descubierto en la fase de prospección. Se delimitan las dimensiones exactas y el enriquecimiento del depósito mineral en un área reducida y claramente definida. Se realizan sondeos, mediciones geológicas y geofísicas en los pozos generados. Se toman muestras representativas a través de una red de muestreo con espaciamiento mucho más reducida. Los métodos de exploración se aplican en la superficie y en el subterráneo. (10)

Luego un geólogo o un minero procesa dicha información y determina si es factible, de serlo se procede a realizar sondeos y mediciones en los pozos generados guardándose la información obtenida en una Base de Datos geológica (BD) en cuatro ficheros de texto: (11)

- Fichero de collares (Coordenadas X, Y, Z de los collares de los Sondajes), donde la X es la coordenada Este del Collar del Sondaje, Y es la coordenada Norte del Collar del Sondaje y Z es su elevación.

- Ficheros de inclinometría (Azimut e Inclinación de cada muestra del Sondaje), este fichero contiene la profundidad a lo largo del pozo a la cual ocurre la desviación y el azimut e inclinación respectivamente de cada muestra independiente.
- Fichero de ensayo o muestra (Campos de interés de las muestras), en este fichero se encuentra la distancia a lo largo del pozo del inicio y fin de cada muestra y los campos de leyes definidos por los usuarios.
- Ficheros de litología (Clasificación Geológica (Litológica) de las muestras y luego se pasa a su explotación), este fichero contiene la distancia a lo largo del pozo del inicio y fin de cada muestra y la litología es decir el tipo de roca definida por el usuario. (11)

### **1.2.2 Descripción del dominio del problema.**

La Universidad de las Ciencias Informáticas, ha adquirido gran experiencia a través de los años en el desarrollo de proyectos productivos, contribuyendo de una forma u otra en la formación de profesionales con grandes conocimientos en áreas de la informática. Entre estos proyectos se encuentra el proyecto Sistema Minero Cubano, el cual, tiene como objetivo la creación de aplicaciones informáticas para la gestión de información de datos geológicos mineros de pozos de perforación, de manera que cumplan con las exigencias propuestas.

Con el desarrollo de estas aplicaciones informáticas se podrán obtener varios beneficios, entre ellos:

- La garantía de seguridad, confiabilidad, disponibilidad y protección de los datos en tiempo real para la toma de decisiones.
- La publicación de la información al servicio geológico y la economía en general con mayor velocidad de acceso y menos recursos.
- Un amplio uso al desarrollarse en un ambiente multiplataforma y con el empleo de *software* libre.
- La proporción de herramientas estandarizadas de captación y consultoría de datos al servicio geológico.
- El incremento del número de usuarios informados y la eliminación o pérdida de los expedientes de las concesiones mineras.

### **1.2.3 Situación problemática.**

En la actualidad, la mayoría de las herramientas informáticas utilizadas en la Industria Minera son de carácter propietarias. En Cuba, a causa del bloqueo impuesto a partir del año 1959 se le niegan sistemas de *software* mineros que posibilitan la planificación, organización y control integral de las operaciones de explotación minera porque la mayoría son producidos por compañías de EE.UU. (12)

El *software* privativo y la propia adquisición de este tipo de aplicaciones dependen de costosos recursos y algunas de sus características son:

- El usuario no puede realizarle ningún tipo de modificación al código fuente.
- No se puede distribuir sin el permiso del propietario.
- Restricciones en el uso (marcadas por la licencia).
- El coste de las aplicaciones es mayor.
- El soporte de la aplicación es exclusivo del propietario.
- El usuario que adquiere *software* propietario depende al 100% de la empresa propietaria. (13)

La industria minera cubana necesita de la realización de *software* cubanos para automatizar los procesos de prospección, planificación y control de la actividad minera, para no depender así de una gama de *software* privativos que son utilizados para llevar a cabo estos procesos en la actualidad, persiguiendo como objetivo ofrecer a la misma un herramienta informática que permita modelar, estimar, y reportar los recursos minerales, así como optimizar, diseñar, y planificar la actividad minera cubana, reduciendo los costos por conceptos de licencia, soporte y capacitación, adecuándose de igual forma a las exigencias internacionales en el área.

Siguiendo las necesidades que plantea la Industria Minera Cubana, se necesita realizar una aplicación ligera para el trabajo con los volúmenes de datos obtenidos de los pozos de perforación en las etapas de exploración y prospección y así llevar la gestión, análisis y validación de los datos de manera organizada, logrando la obtención de buenas prácticas y beneficios para estas empresas.

### **1.3 Análisis de soluciones existentes.**

Actualmente a nivel mundial existen varios *software* dedicados a la gestión de los procesos geológicos mineros, dentro de ellos se destacan GEMCOM, SURPAC y DATAMINE. Muchas de las empresas cubanas han adquirido licencias de estos sistemas informáticos para certificar la calidad de la estimación



de los recursos y reservas de yacimientos minerales cubanos, y poder utilizar capital extranjeros para financiar sus proyectos mineros. Aunque estas aplicaciones gocen de gran prestigio internacional, el alto precio de costo de sus licencias provoca que la mayoría de las empresas cubanas que realizan la pequeña minería no puedan adquirirlos.

En Cuba se han estado desarrollando algunas herramientas informáticas para la gestión de la información minera, los cuales no han podido ser completados y cuentan con diversos problemas para la realización de esta actividad.

Todas las aplicaciones informáticas mencionados anteriormente, realizan la gestión de información referente a los datos geológicos de yacimientos minerales y les permiten a los usuarios:

Controlar los recursos minerales de un yacimiento mineral.

La planificación de la producción minera a corto, mediano y largo plazo.

Reportar los pozos de perforación que aporten beneficios para la producción de la industria

Aunque todos estos sistemas realizan la gestión de la información de yacimientos minerales, llevando a cabo las fases de prospección, exploración y tienen mucho prestigio debido a su utilidad y seguridad de los datos, se emplean poco en el país pues su principal inconveniente es que son privativos, lo que conlleva a que tengan un elevado costo por concepto de licencia, no se le pueden hacer cambios en su código fuente. El tipo de SGDB utilizado por estos sistemas no tiene una arquitectura cliente-servidor lo cual provoca que los usuarios no puedan acceder concurrentemente a los datos.

### **1.4 Conclusiones Parciales**

El creciente auge que ha tenido la minería en las esferas socioeconómicas a nivel mundial, ha demostrado que contar con sistemas que faciliten su manipulación es de vital importancia. El análisis realizado en el presente capítulo ratifica que estas aplicaciones constituyen un campo de investigación muy activo y amplio. Para el desarrollo del mismo fue necesaria la indagación de gran cantidad de información acerca del proceso de exploración minera, detallándose así los problemas actuales y las condiciones que lo rodean y todos los elementos que afectan este proceso. Además de analizar los sistemas enfocados a la

gestión de datos de yacimientos minerales, lo que permitió valorar la definición de funcionalidades para la aplicación a desarrollar así como establecer pautas que permitan crear un sistema informático que gestione la información obtenida de los pozos de perforación.

## 2. Capítulo 2: Tecnologías, metodologías y herramientas.

En el presente capítulo se explicará mediante una descripción breve y detallada las tecnologías y herramientas utilizadas en la investigación, así como las características de cada una de ellas, para alcanzar un buen desarrollo de la aplicación.

### 2.1 Arquitectura de Software

Sobre la arquitectura de software, en (14) se afirma lo siguiente:

La Arquitectura de *Software* se refiere a “las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos.”

El Modelo Arquitectónico que se utilizará para el despliegue del sistema es **Cliente-Servidor**: Este modelo es utilizado para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes. La utilización de este modelo trae consigo menores costes de operación.

Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste le responde proporcionándolo. Normalmente, pero no necesariamente, el cliente y el servidor están ubicados en distintos procesadores. Los clientes se suelen situar en ordenadores personales y/o estaciones de trabajo y los servidores en procesadores departamentales o de grupo. Para la comunicación de los procesos con la red se emplea un tipo de equipo lógico denominado middleware que controla las conversaciones. Su función es independizar ambos procesos (cliente y servidor). La interfaz que presenta es la estándar de los servicios de red que hace que los procesos “piensen en todo momento que se están comunicando con una red.

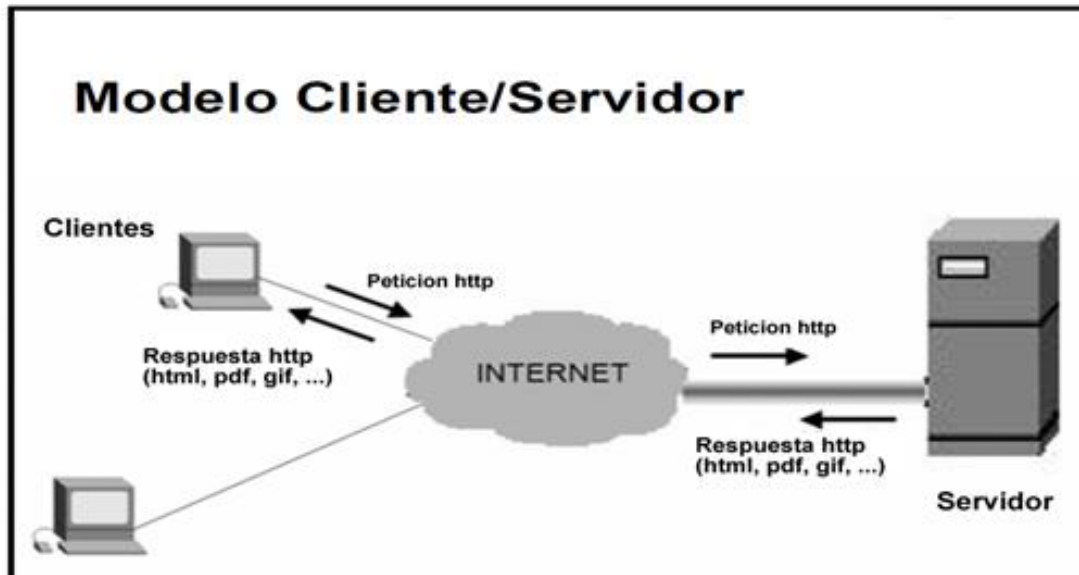


Figura 3. Modelo Cliente/Servidor (<http://t1.gstatic.com/>)

Entre las características que presenta se encuentran:

- El servidor presenta a todos sus clientes una interfaz única.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa y no depende de la ubicación física del servidor, ni de sus características específicas.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente. (15)

## 2.2 Estilo Arquitectónico. Modelo Vista Controlador.

El estilo Modelo Vista Controlador es utilizado para establecer la estructura lógica de la aplicación. Separa el modelo del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes:

- El Modelo: Es el encargado de administrar el comportamiento y los datos del dominio de aplicación.
- La Vista: Maneja la visualización de la información.

- El Controlador: Interpreta las acciones del ratón y el teclado, informando al modelo y/o a la vista para que cambien según resulte apropiado.

La utilización del MVC<sup>3</sup> brinda un conjunto de ventajas, tales como:

- El soporte de vistas múltiples dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista.
- La adaptación al cambio dado que el modelo no depende de las vistas, por lo que agregar nuevas opciones de presentación generalmente no afecta al modelo. (16)

### 2.3 Metodología de desarrollo

Las metodologías de desarrollo del *software* son un conjunto de procedimientos, técnicas y ayudas utilizadas para que el producto final tenga la calidad requerida, que se ajuste al tiempo planificado para su construcción y que utilice los menores recursos posibles.

#### **Metodología de desarrollo: RUP<sup>4</sup>.**

RUP es un producto de Rational *Software* que presenta un modelo de proceso de ingeniería de software completo. Se escoge como la más adecuada para el desarrollo, por las ventajas que ofrece su uso al garantizar un proceso de desarrollo organizado, robusto y correctamente documentado. Para la construcción del sistema se necesita una metodología conocida por el equipo de proyecto, que permita integrar las etapas de desarrollo fácilmente y que sea compatible o tenga relación con UML<sup>5</sup>, que dirija las tareas de cada miembro del equipo por separado pero a la vez como un todo, y que ofrezca criterios para el control y medición de la calidad del producto. Todas estas características están presentes en RUP.

Presenta además tres características fundamentales:

1. Dirigido por casos de uso: Para desarrollar el sistema propuesto con éxito es necesario conocer las necesidades de los usuarios que consultan información en el proyecto Sistema Minero Cubano, para de esta manera poder brindar un servicio de calidad. A partir de estas necesidades se pueden

---

<sup>3</sup> Modelo Vista Controlador (*Model View Controller*).

<sup>4</sup> Proceso Unificado de Rational (*Rational Unified Process*).

<sup>5</sup> Lenguaje de Modelado Unificado (*Unified Model Language*).

identificar casos de uso que van a guiar todo el proceso de desarrollo de aquí en adelante. Estos representan un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante. La unión de todos los casos de uso constituye el modelo de casos de uso el cual describe la funcionalidad total del sistema. Los casos de uso guían la arquitectura del sistema y la arquitectura del sistema influye en la selección de los casos de uso.

2. Centrado en la arquitectura: La arquitectura va a permitir la descripción del sistema a desarrollar mediante diferentes vistas que incluyen los aspectos estáticos y dinámicos más significativos del mismo, logrando así una mayor organización y entendimiento de este. Esta se refleja en los casos de uso pues cada producto tiene tanto una función como una forma, ninguna es suficiente por sí sola.
3. Iterativo e incremental: Para el desarrollo del sistema, RUP permite que pueda dividirse el trabajo en partes más pequeñas o mini proyectos los que no son más que iteraciones que resultan en un incremento. Una iteración es una secuencia de actividades con un plan establecido y criterios de evaluación, cuyo resultado es una versión del *software*. (17)

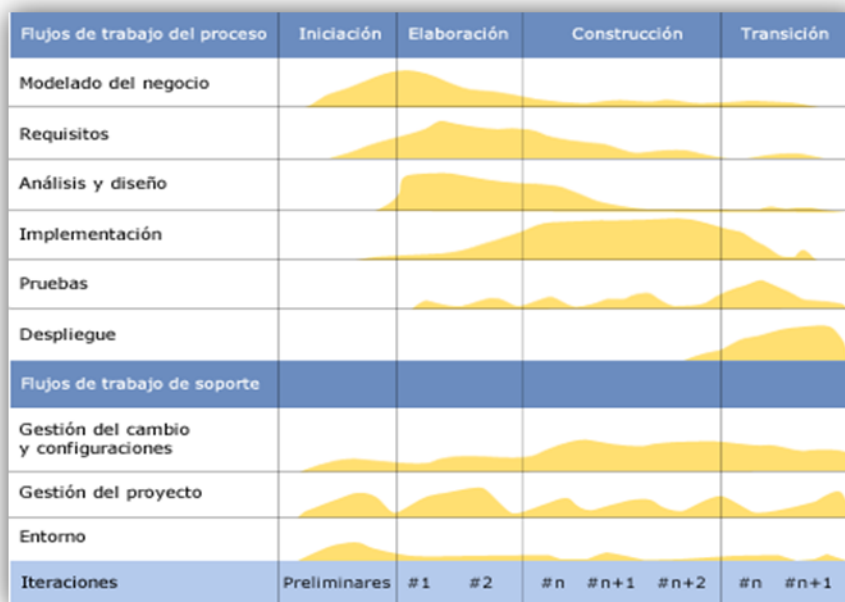


Figura 4. Fases e Iteraciones de la Metodología RUP(<http://jummp.wordpress.com>)

2.4 Lenguaje de Modelado. UML 2.1.1

Es importante definir además el lenguaje de modelado para permitir que cada una de las partes que comprende el desarrollo de todo *software* de diseño orientado a objetos, se especifique, documente y visualice con un lenguaje común.

UML 2.1.1 es un lenguaje que permite modelar construir y documentar los diferentes aspectos de un sistema de *software* en desarrollo así como también el modelado de negocios y almacenamiento de datos. Es un lenguaje visual de modelación centrado en objetos, independiente del lenguaje, procesos de implementación o desarrollo. En la actualidad es el lenguaje modelo en el análisis y diseño de sistemas de cómputo. Se puede decir que mientras más complejo es el sistema a desarrollar más beneficios puede brindar la utilización de UML gracias a sus ventajas, las cuales se mencionan a continuación:

- Diseño y documentación.
- Código reutilizable.
- Descubrimiento de fallas.
- Ahorro y tiempo en el desarrollo de *software*.
- Mayor facilidad a la hora de realizar modificaciones.
- Fácil comunicación entre programadores.

En cada fase del proceso de desarrollo del *software* se crean un conjunto de artefactos que constituyen una entrada fundamental a las fases que le suceden.

UML es utilizado en la realización de la aplicación pues la metodología RUP se basa en ese lenguaje de modelado para la creación de sus artefactos. Los modelos creados sirven como documentación técnica de la aplicación Web para obtener un entendimiento de la misma. (18)

### **2.5 Herramienta para el modelado: Visual Paradigm 8.0**

Visual Paradigm 8.0 es una herramienta CASE<sup>6</sup> que utiliza UML como lenguaje de modelado. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de

---

<sup>6</sup>Ingeniería de Software Asistida por Computador (*Computer Aided Software Engineering*).

desarrollo del *software* a través de la representación de todo tipo de diagramas. Constituye una herramienta de *software* libre de probada utilidad para el analista. Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de *software* de forma fiable a través de la utilización de un enfoque OO<sup>7</sup>. (19)

Se utiliza esta herramienta CASE pues está basada en *software* libre, es multiplataforma, fácil de instalar, actualizar, y además, propicia un conjunto de ayudas para la realización de los artefactos que se generan durante el desarrollo del *software*.

### 2.6 Lenguaje de programación utilizados

#### PHP<sup>8</sup> 5

El lenguaje de programación del lado del servidor escogido es PHP en su versión 5, pues por sus características se considera que posee la mejor mezcla entre flexibilidad y rendimiento para la realización de páginas Web dinámicas. Cuenta con abundante documentación lo que permite un mayor entendimiento del mismo. Este lenguaje está asociado con una extensa biblioteca que va creciendo conforme se realizan nuevas versiones, la cual permite la realización de disímiles tareas como la encriptación, el acceso a base de datos y el tratamiento de ficheros. Es un lenguaje abierto, multiplataforma, presenta un amplio soporte para la Programación Orientado a Objetos.

Soporta varios SGBD<sup>9</sup>, como MySQL, Oracle y PostgreSQL, está integrado con varias bibliotecas externas que permiten por ejemplo generar documentos PDF<sup>10</sup> y hacer análisis de código XML, y es atendido por una amplia comunidad de desarrolladores que lo actualizan frecuentemente con extensiones y mejoras y además encuentran y reparan errores de funcionamiento del lenguaje. (20)

#### JavaScript

JavaScript es un lenguaje de programación interpretado que se utiliza principalmente para crear páginas Web dinámicas. JavaScript, desarrollado por Netscape es un lenguaje de script multiplataforma basado en objetos. Es un lenguaje ligero, está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores Web. (21)

---

<sup>7</sup> Orientado a Objetos.

<sup>8</sup> Preprocesador de Hipertexto (*Hyper Text Preprocessor*).

<sup>9</sup> Sistema Gestor de Base de Datos.

<sup>10</sup> Formato de Documento Portátil (del término en inglés PDF: Portable Document Format).



Se utilizó JavaScript por ser una tecnología usada principalmente para la validación del lado del cliente en cuanto a la entrada de datos en los formularios. Este permite enviar los datos al servidor de manera correcta, esto no indica que del lado del servidor la validación no se implemente, también se debe hacer dado que el uso de estos scripts se puede limitar en la mayoría de los navegadores.

### 2.7 Sistema Gestor de Bases de Datos.

Los SGBD son programas que permiten crear y mantener una base de datos asegurando su confidencialidad, integridad y seguridad. Entre las características deseables de un SGBD se encuentran el control de redundancia, la restricción de los accesos, y el cumplimiento de las restricciones de integridad. (22)

#### PostgreSQL 9.1.

PostgreSQL 9.1 es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD<sup>11</sup> y con su código fuente disponible libremente. Es uno de los sistemas de gestión de bases de datos de código abierto más utilizado en el mercado. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multi-hilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. Actualmente se considera el sistema libre más avanzado, ya que soporta la gran mayoría de las transacciones SQL y el control concurrente. (23)

Las principales ventajas que presenta son:

- Gran escalabilidad: posibilita atender un mayor número de peticiones concurrentes.
- Extensible: el código fuente está disponible para todos sin costo alguno.
- Diseñado para ambientes de alto volumen: usa una estrategia de almacenamiento de filas llamada MVCC<sup>12</sup> para conseguir una mejor respuesta en ambientes de grandes volúmenes.
- Soporte: existe una comunidad de profesionales que contribuyen con su desarrollo.
- Multi-plataforma: está disponible en casi cualquier UNIX y con versión nativa para Windows. (23)

---

<sup>11</sup> Licencia de software libre que permite el uso del código fuente en software no libre (*Berkeley Software Distribution*).

<sup>12</sup> Múltiples Versiones de Control de Concurrencia (*Multi-Version Concurrency Control*).

La utilización de PostgreSQL 9.1 permite, modificarlo y distribuirlo en productos comerciales o no comerciales, puede ser extendido por el usuario añadiendo elementos como tipo de datos, operadores, índices y funciones. Una de las ventajas que presenta este sistema, es que es una de las base de datos más potentes y robustas en el mercado. Brinda facilidad de administración e implementación de estándares y funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema.

### **PgAdmin III 1.14.0**

PgAdmin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en diferentes Sistemas Operativos. Es capaz de gestionar versiones a partir de PostgreSQL 7.3 ejecutándose en cualquier plataforma. PgAdmin III está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. (24)

Se utilizó pgAdmin III ya que es una aplicación diseñada para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. Además de que la interfaz gráfica soporta todas las características de PostgreSQL y facilita la administración.

### **2.8 Servidor Web. Servidor Apache 2.0**

Un servidor web es un ordenador que utiliza el protocolo HTTP<sup>13</sup> para enviar páginas Web al ordenador de un usuario cuando estos se conectan al mismo. Gestionan cualquier aplicación en el lado del servidor generando una respuesta en cualquier lenguaje o aplicación en el lado del cliente. Para el desarrollo de la solución propuesta se utiliza el servidor web Apache en su versión 2.0, debido a las características que posee y los beneficios que aporta su uso.

Apache es el servidor Web, que provee servicios http, hecho por excelencia. Al ser altamente configurable, robusto y estable hacen que cada vez millones de usuarios reiteren su confianza en este programa. (25)

---

<sup>13</sup> Protocolo de Transferencia de Hipertexto (*Hypertext Transfer Protocol*).

Es utilizado este servidor pues puede ejecutarse en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal, es una tecnología gratuita de código fuente abierto, es altamente configurable, trabaja con gran cantidad de lenguajes, y permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.

### **2.9 Framework**

Un Framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Facilita la programación de aplicaciones, pues encapsula operaciones complejas en instrucciones sencillas. (26)

#### **Symfony 2.0**

Symfony es un framework PHP basado en la arquitectura MVC que permite optimizar el desarrollo de aplicaciones Web gracias a sus características. Fue escrito desde un origen para ser utilizado sobre la versión 5 de PHP ya que hace ampliamente uso de la orientación a objetos que caracteriza a esta versión. (27) Es multiplataforma, fácil de instalar y configurar y sigue las mejores prácticas y patrones de diseño para la Web.

Se seleccionó esta versión de Symfony para la construcción del sistema ya que posibilita dividir el sistema en aplicaciones y módulos independientes reduciendo considerablemente el trabajo del programador, simplifica el desarrollo de la aplicación mediante la automatización de patrones, utilizados para resolver las tareas comunes, encapsula operaciones complejas en instrucciones sencillas y separa la lógica del negocio, lógica del servidor y presentación de la aplicación. Por otra parte presenta un código fácil de leer que incluye comentarios y que posibilita que el mantenimiento sea una tarea sencilla. Se enfoca en agilizar el proceso de construcción al mismo tiempo que reduce el trabajo del implementado.

#### **ExtJS 4.07**

ExtJS es una biblioteca de clases JavaScript. Permite construir aplicaciones complejas en Internet además de flexibilizar el manejo de componentes de la página como el DOM, peticiones AJAX<sup>14</sup>,

---

<sup>14</sup>JavaScript Asíncrono y XML (del término en inglés AJAX: Asynchronous JavaScript And XML).

DHTML<sup>15</sup>, tiene la gran funcionalidad de crear interfaces de usuario bastante funcionales. Esta biblioteca incluye:

- Componentes UI<sup>16</sup> de alto rendimiento y personalizables.
- Modelo de componentes extensibles.
- Una API<sup>17</sup> fácil de usar, licencias *Open Source* (GPL<sup>18</sup>) y comerciales.

### **Ventajas:**

- Permite crear aplicaciones complejas utilizando componentes predefinidos.
- Evita el problema de tener que validar el código para que funcione bien en cada uno de los navegadores.
- Relación entre Cliente-Servidor balanceado: Se distribuye la carga de procesamiento, permitiendo que el servidor pueda atender más clientes al mismo tiempo.
- Eficiencia de la red: Debido a que disminuye el tráfico en la red pues las aplicaciones cuentan con la posibilidad de elegir qué datos desea transmitir al servidor y viceversa.
- Comunicación asíncrona: Este tipo de aplicación no está sujeta a un clic o una acción del usuario para cargar información sin que el cliente se dé cuenta. (28)

### **2.10 NetBeans 7.2.1 como entorno de desarrollo integrado (IDE).**

NetBeans es un entorno de desarrollo integrado, hecho principalmente para el lenguaje de programación Java, pero también utiliza otros como PHP. Es un proyecto de código abierto y una plataforma de aplicaciones que permiten a los desarrolladores crear rápidamente *software*. NetBeans IDE es un producto libre y gratuito sin restricciones de uso. (29)

---

<sup>15</sup>HTML dinámico (*Dynamic HTML*).

<sup>16</sup> Interfaz gráfica de usuario (*User Interface*).

<sup>17</sup> Interfaz de Programación de Aplicaciones (*Application Programming Interface*).

<sup>18</sup> Licencia Pública General de GNU (*GNU General Public License*).

Se utilizó NetBeans por ser libre de costo y multiplataforma, permitir desarrollar aplicaciones utilizando el lenguaje PHP y presentar un aceptable completamiento de código, lo que permite agilizar el trabajo. Además tiene soporte para utilizar Symfony y ayuda en la navegación de las clases predefinidas en la plataforma.

### **2.11 Conclusiones Parciales**

Con el análisis realizado en el presente capítulo se especificaron las descripciones de las herramientas y tecnologías a utilizar que permitan la edición de la Aplicación Web. Su selección fue basada teniendo en cuenta los principios de soberanía tecnológica que defiende el país. Finalmente, ya definidas las herramientas y tecnologías a utilizar para el desarrollo de la solución, se puede afirmar que los aspectos abordados en el presente capítulo proveen la base de conocimientos necesaria para presentar una propuesta de solución.

### **3. Capítulo 3: Características del Sistema.**

En el presente capítulo se propone la solución diseñada para las empresas mineras cubanas. Se especifican los actores y trabajadores, las descripciones de los casos de uso, los requisitos funcionales y no funcionales que debe cumplir el sistema a desarrollar. Además, se muestran los diagramas de casos de uso, que permiten observar las relaciones de los actores que interactúan con el sistema, lo que conlleva a lograr que el trabajo sea guiado hacia una solución factible.

#### **3.1 Propuesta del Sistema**

La solución que se propone estará diseñada para las empresas mineras existentes en las diferentes provincias del país, desde la industria del Níquel en Holguín, hasta las industrias de materiales de construcción en Artemisa, Camagüey, Cienfuegos, Santiago de Cuba etc. El objetivo general se centra en implementar una herramienta que gestione los datos geológicos mineros de pozos de perforación utilizados en el mismo. Este sistema informático contará con una base de datos en la cual se guardará la información de los diferentes tipos de datos existentes de los pozos y a su vez se podrán obtener los resultados para un posterior análisis.

Una vez que el Minero tenga acceso al sistema podrá interactuar con la aplicación web y realizar diferentes actividades entre las que se encuentran adicionar, eliminar, modificar, validar, calcular, mostrar y refrescar los datos de los pozos de perforación que se almacenan en la base de datos.

#### **3.2 Modelo de Dominio**

Un Modelo de Dominio es un artefacto de la disciplina de análisis, representando las clases conceptuales del mundo real. Este tipo de modelo explica los conceptos significativos en un dominio del problema, identificando los atributos y las asociaciones. Lo cual tiene como objetivo ayudar a comprender los conceptos que utilizan los usuarios, los conceptos con los que trabajan y con los que se deberá trabajar.

#### **Diagrama del Modelo de Dominio**

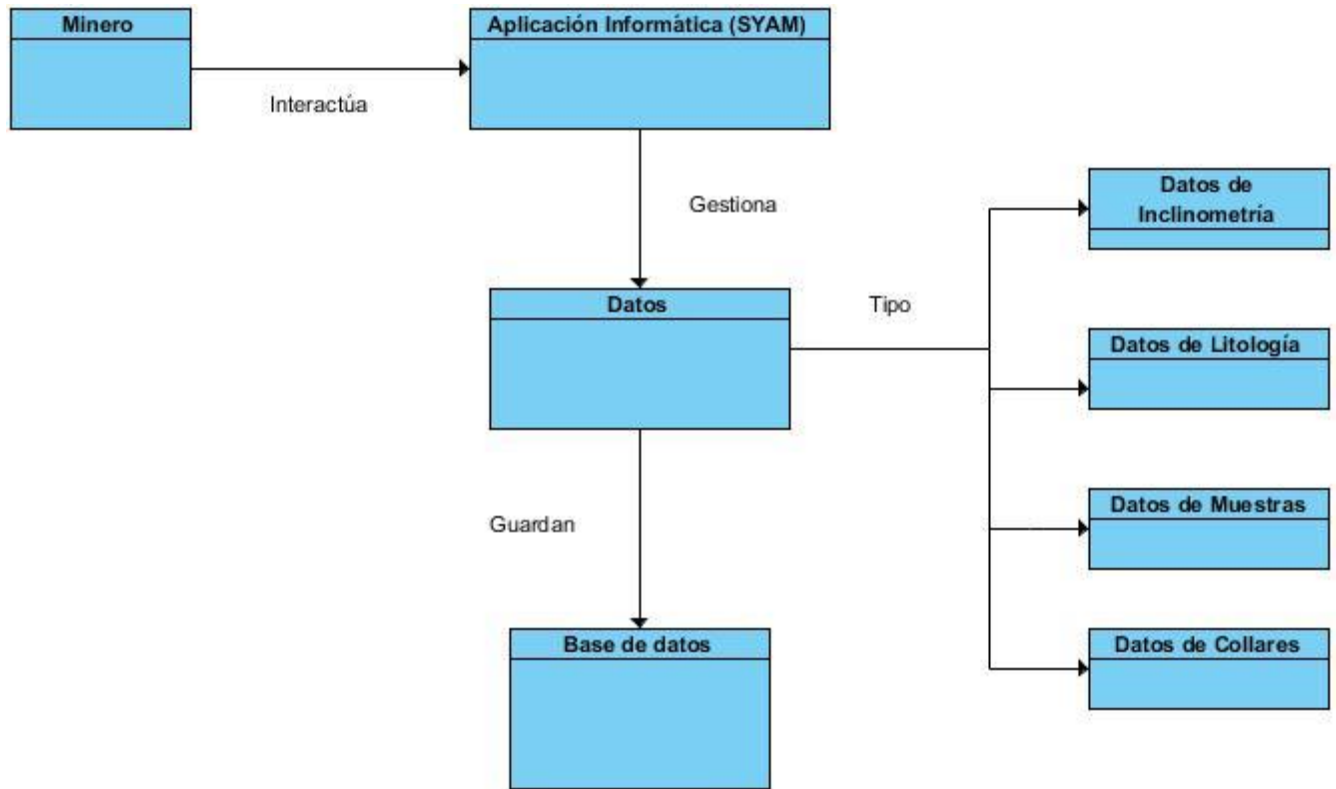


Figura 5. Diagrama del Modelo de Dominio.

### 3.2.1 Glosario de términos del dominio.

- Minero: Persona, organismo u organización que necesite acceder a los datos geológicos de los pozos de perforación a través de la Aplicación informática.
- Aplicación informática: Sistema de gestión de datos geológicos que permite mantener el control de los datos que poseen los pozos.
- Datos: Es la información de los pozos que se encuentran contenidos dentro de la base de datos existente.
- Datos de Inclínometría: Contienen la densidad de los minerales existentes en los diferentes pozos de sondeo almacenados en la Base de Datos.
- Datos de Litología: Maneja la información sobre la litología existente en los diferentes pozos de sondeos, es decir el tipo de roca existente en los pozos de sondeos.

- Datos de Muestra: Es la información detallada de un yacimiento mineral extraído con el objetivo de determinar la concentración de los minerales en dicho yacimiento. Las muestras se caracterizan por tener un identificador, un largo, un intervalo que comprende los valores en los que empieza y termina la muestra, así como, el azimut y la elevación. El azimut es el ángulo de una dirección contado en el sentido de las agujas del reloj a partir del norte geográfico. La elevación es el ángulo sobre el plano horizontal.
- Datos de Collares: Es la información del pozo referente a las dimensiones exactas.
- Base de datos: Fichero donde se encuentran registrados todos los resultados de los datos geológicos mineros de los pozos de perforación realizados.

### 3.3 Modelo del Sistema

Luego de haber establecido una relación entre las clases que conforman el dominio del sistema, así como la descripción de los principales conceptos a través del modelo de dominio, se presentan los principales artefactos del flujo de trabajo Requisitos que tiene como objetivo principal guiar el desarrollo hacia el sistema correcto.

#### 3.3.1 Requisitos funcionales

Los requisitos funcionales son condiciones o capacidades con los que debe cumplir el sistema, son acuerdos entre el cliente y los desarrolladores sobre lo que debe o no debe hacer el sistema. (30)

A continuación se expondrán los requisitos funcionales que debe cumplir el sistema propuesto:

RF<sup>19</sup> 1. Establecer conexión a la base de datos.

RF 2. Mostrar los datos almacenados en una tabla.

RF 3. Mostrar los datos resultantes de ejecutar una vista.

RF 4. Filtrar elementos de una tabla por rangos numéricos.

RF 5. Adicionar un registro a una tabla.

RF 6. Modificar un registro de una tabla.

RF 7. Eliminar un registro de una tabla.

---

<sup>19</sup> Requisito Funcional (RF)



- RF 8. Validar que no existan intervalos repetidos en las muestras, inclinometría y litología.
- RF 9. Validar que no existan intervalos solapados en las muestras, inclinometría y litología.
- RF 10. Validar que todos los pozos tengan inclinometría.
- RF 11. Validar que el intervalo de muestra sea mayor que 0.
- RF 12. Reportar los pozos que no tengan muestras.
- RF 13. Reportar los pozos que no tengan intervalos de litología.
- RF 14. Refrescar la información de una tabla que está siendo mostrada en el sistema.
- RF 15. Refrescar periódicamente la información de una tabla que está siendo mostrada en el sistema.
- RF 16. Calcular las coordenadas de los intervalos de la inclinometría.
- RF 17. Calcular las coordenadas de los intervalos de la litología.
- RF 18. Calcular las coordenadas de los intervalos de las muestras.
- RF 19. Configurar las validaciones a realizar sobre los datos.
- RF 20. Identificar las tablas que poseen columnas geométricas.
- RF 21. Calcular análisis estadístico básico.

### 3.3.2 Requisitos no funcionales

Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, tiempo de respuesta y la capacidad de almacenamiento. (31)

A continuación se expondrán los requisitos no funcionales que debe cumplir el sistema propuesto:

#### Requisitos de usabilidad

RnF<sup>20</sup> 1. La información deberá estar disponible en todo momento, limitada solamente por las restricciones de acuerdo a las políticas de seguridad definidas.

---

<sup>20</sup> Requisito no Funcional (*RnF*)

### Requisitos de fiabilidad

RnF 2. El sistema debe estar disponible todo el tiempo para sus usuarios, descontando el tiempo que se encuentre en mantenimiento y la ocurrencia de alguna falla externa.

### Requisitos de eficiencia

RnF 3. El tiempo de respuesta por transacción de las peticiones realizadas a la aplicación estará en el rango de 2 a 5 segundos, en dependencia de la cantidad de información a procesar.

### Restricciones de diseño

RnF 4. El producto de *software* final debe diseñarse sobre una arquitectura cliente-servidor.

### Requisitos de *hardware*

- PC Servidor

RnF 5. Memoria RAM 1024 MB.

RnF 6. Velocidad de procesamiento del microprocesador 1GHz o superior.

- PC Cliente

RnF 7. Memoria RAM 512 MB, recomendable 1024 MB.

RnF 8. Velocidad de procesamiento del microprocesador 1GHz o superior.

### Requisitos de *software*

- PC Servidor

RnF 9. Instalación del servidor web Apache 2.0.

RnF 10. Instalación del servidor de base de datos PostgreSQL 9.1

RnF 11. Instalación de PHP 5.3.3 o superior.

RnF 12. Sistemas operativos GNU/Linux o Windows Server 2000 o superior.

- PC Cliente

RnF 13. Un Navegador (Mozilla Firefox u otro navegador) que cumpla con los estándares W3C<sup>21</sup>.

RnF 14. Sistema operativo: GNU/Linux, Windows o Mac OS.

---

<sup>21</sup>World Wide Web Consortium

**3.3.3 Diagrama de casos de uso del sistema.**

Los casos de uso son un conjunto de escenarios que identifican una línea de utilización para el sistema que va a ser construido y que facilitan una descripción de cómo el sistema se usará. Por su parte, un DCUS<sup>22</sup> muestra la relación entre los casos de uso y los actores del sistema. (30) Para la aplicación Web se definen 9 casos de uso.

A continuación se muestra el diagrama:

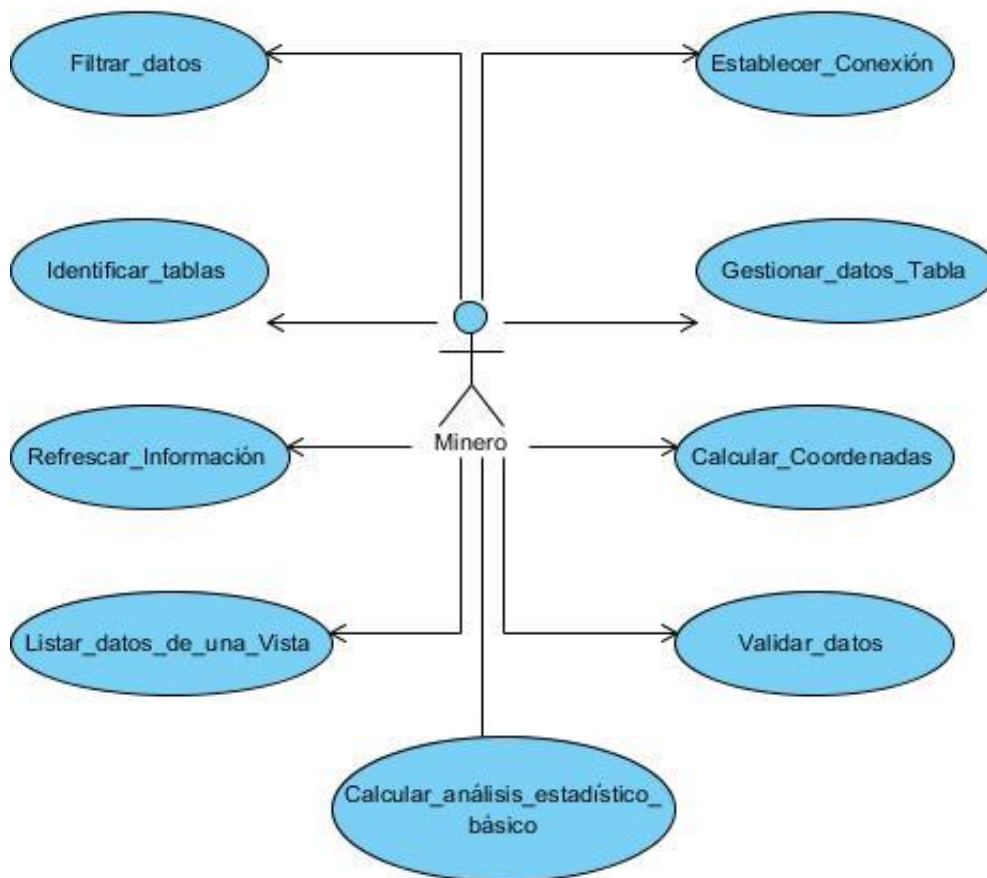


Figura 6. Diagrama de caso de uso del sistema.

**Descripción de los actores del sistema.**

<sup>22</sup> Diagrama de casos de uso del sistema (DCUS)

Los actores son personas, organizaciones u otros sistemas automatizados que interactúan directamente con la aplicación y están identificados con un rol en dependencia del papel que asuman dentro de la misma. Son los que interactúan con el sistema a través de los casos de uso que inicializan.

Para la aplicación se define 1 actor que a continuación se describe:

**Tabla 1. Descripción de los actores del sistema.**


Actor	Descripción
Minero	El Minero es el responsable de manejar los ficheros de los datos de los pozos de perforación. Puede adicionar, eliminar, modificar, validar, calcular, reportar y refrescar los datos de los pozos de perforación que se almacenan en la base de datos.

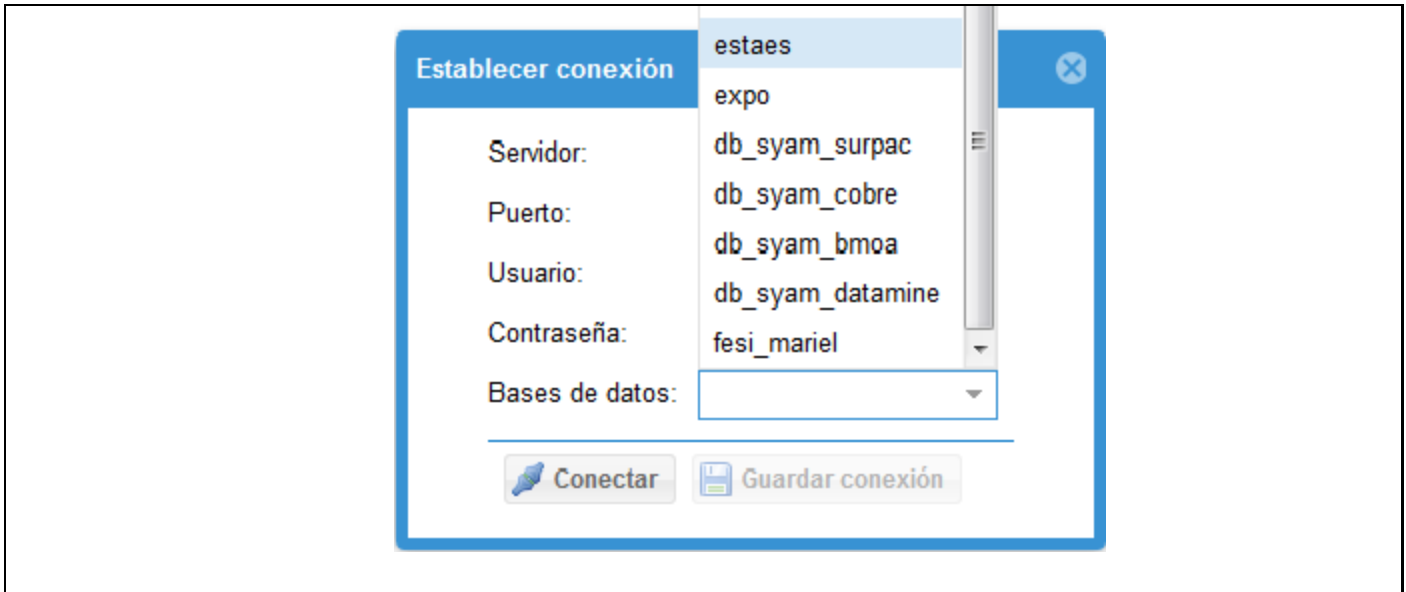
**Descripción de los casos de uso del sistema.**

Para lograr una mejor comprensión de la funcionalidad de cada caso de uso del sistema, se realiza la descripción textual de los mismos. A continuación se realiza la descripción del caso de uso Establecer conexión, las demás se pueden encontrar en **Anexo A**.

**Tabla 2. Descripción del caso de uso “Establecer Conexión”**

<b>Caso de Uso:</b>	Establecer conexión.
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permitir al minero conectarse a la base de datos.
<b>Resumen:</b>	El caso de uso inicia cuando el minero interactúa con la aplicación y ejecuta la pestaña “Configurar conexión” y termina cuando se cierra la conexión.
<b>Precondiciones:</b>	Se encuentre creada la base de datos.
<b>Postcondiciones:</b>	Se conecta a la aplicación.

<b>Referencias:</b>	RF1	
<b>Prioridad:</b>	Crítico	
<b>Flujo Normal de Eventos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. El caso de uso comienza cuando el Minero selecciona la opción "Configurar conexión".	2. El sistema muestra la interfaz con la configuración de las propiedades generales de la conexión (Servidor, Puerto, Usuario, Contraseña).	
<b>Prototipo de Interfaz</b>		
Interfaz 1		
		
3. El Minero pulsa el botón con la opción "Conectar".	4. El sistema habilita la opción "Base de Datos" para seleccionar la base de datos con la que se desea trabajar.	
<b>Prototipo de Interfaz</b>		
Interfaz 2		

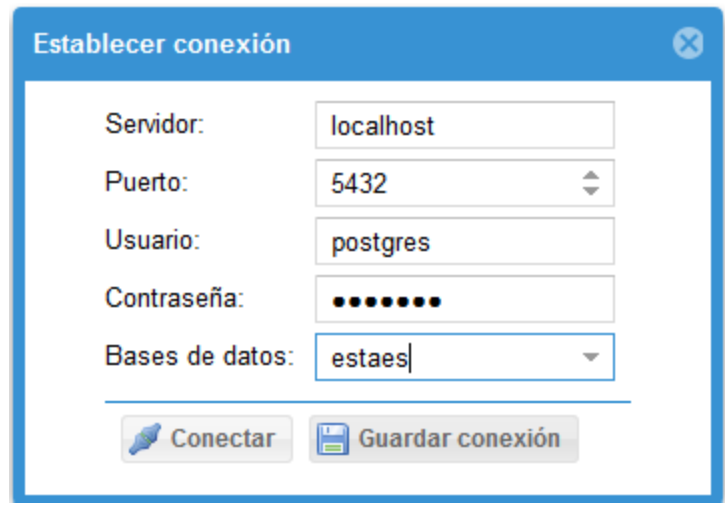


5. El Minero pulsa el botón "Guardar Conexión" al seleccionar la base de datos.

6. Se establece la conexión.

**Prototipo de Interfaz**

Interfaz 3



Flujo Alternativo de Eventos	
5.1 El Minero omite el campo obligatorio de la contraseña para acceder a la aplicación.	6.1 El sistema muestra un mensaje de error: "Verifique que los datos sean correctos."

### 3.3 Conclusiones Parciales

Los artefactos que se generaron en el presente capítulo permiten contar con una documentación organizada de los principales conceptos que se manejan en el sistema informático y sus relaciones, así como las restricciones con las que cuenta el mismo para su futura implementación. Se generó el modelo de dominio del sistema, ya que no se tienen bien definidos los procesos del negocio, lo que permitió identificar los problemas existentes. En el levantamiento de requisitos se obtuvieron las funcionalidades con las que debe contar el sistema informático, y a partir de ellos se modeló la vista de casos de uso del sistema, que permite definir cómo será el comportamiento de los usuarios con respecto a la aplicación. Se considera que luego de haber presentado la propuesta de solución se está en condiciones de pasar a la implementación de la misma.

### **4. Capítulo 4. Diseño, Implementación y Pruebas de la solución propuesta.**

En el presente capítulo se presentan los artefactos ingenieriles relacionados con el diseño, implementación y validación del sistema: el modelo de diseño, diagramas de interacción, el modelo de despliegue y el modelo de casos de prueba. Se abunda sobre el funcionamiento de los framework seleccionados, y se explica cuáles son los patrones utilizados para el desarrollo del sistema.

#### **4.1 Análisis y Diseño**

Análisis y diseño es un flujo de trabajo que tiene como objetivo traducir los requisitos a una especificación que describa como se debe implementar el sistema. Por una parte el análisis consiste en obtener una visión general del sistema, o sea, permite refinar y estructurar los requisitos funcionales para obtener una comprensión más precisa de estos.

Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales que deben estar presentes en el sistema. El equipo de desarrollo puede prescindir del Análisis cuando los requisitos son sencillos y conocidos, si se puede identificar de manera simple la forma que va a adoptar el sistema y se cuenta con la comprensión correcta de los requisitos permitiendo la construcción de un *software* que los represente lo más directamente posible. (30)

En la presente investigación no se realiza el modelo de análisis ya que se presentan las condiciones explicadas anteriormente. Otra de las razones es que se desea evitar el coste en tiempo que implica el Análisis puesto que se logra el objetivo y propósito del mismo de una forma diferente. De esta manera a continuación se presenta el desarrollo del Diseño como parte del flujo de trabajo Análisis y Diseño propuesto por RUP.

#### **4.2 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 sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como una entrada fundamental de las actividades de implementación. (30)



## 4.3 Patrones de diseño utilizados

Los patrones de diseño constituyen la base para la búsqueda de soluciones a problemas comunes en el proceso de desarrollo de *software*. Se caracterizan por su reusabilidad, flexibilidad y aplicabilidad a diferentes problemas de diseño en diversas circunstancias. Por su importancia, a continuación se describen los patrones utilizados que implementa el framework Symfony, como son los patrones GRASP<sup>23</sup>, que describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. (26)

### Patrones GRASP:

- Patrón Creador: Consiste en asignar a un objeto la responsabilidad de crear otro Objeto. Un objeto es responsable de crear una nueva instancia de alguna clase si agrega o contiene objetos de ella, registra las instancias de sus objetos o tiene los datos de inicialización que serán enviados a ella cuando el objeto sea creado. Este patrón se ve reflejado en la clase QueryController de la Aplicación Web.
- Patrón Controlador: El patrón ofrece una guía para tomar decisiones sobre los eventos de entrada, asignando la responsabilidad del manejo de mensajes de los eventos del sistema a una clase controladora, ya que los elementos de interfaz y sus controladores de eventos, no deben ser responsables de controlar los eventos del sistema. Este patrón se ve reflejado en las clases controladoras de la Aplicación Web: DefaultController, ManageConnectionController, ManageQueryController.
- Patrón Alta Cohesión: La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realizan un trabajo enorme. Fomenta la reutilización, mejorando la claridad y facilidad del diseño. Este patrón se ve reflejado en la clase ManageQueryController de la Aplicación Web.

## 4.4 Diagramas de clases del diseño

---

<sup>23</sup>Patrones Generales de Asignación de Responsabilidad de Software (del término en inglés GRASP: General Responsibility Assignment Software Patterns).

# DISEÑO, IMPLEMENTACIÓN Y PRUEBAS DE LA SOLUCIÓN PROPUESTA

Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones del sistema con sus relaciones estructurales, dando así una vista del diseño estático. En el caso de las aplicaciones web el diagrama de clases de diseño representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. (30)

A continuación se realiza el diagrama de clases del diseño para el caso de uso Establecer Conexión. Posteriormente se realiza una descripción de las clases que se representan para un mejor entendimiento de sus funciones y su interrelación.

El resto de los diagramas se pueden encontrar en el **Anexo B**.

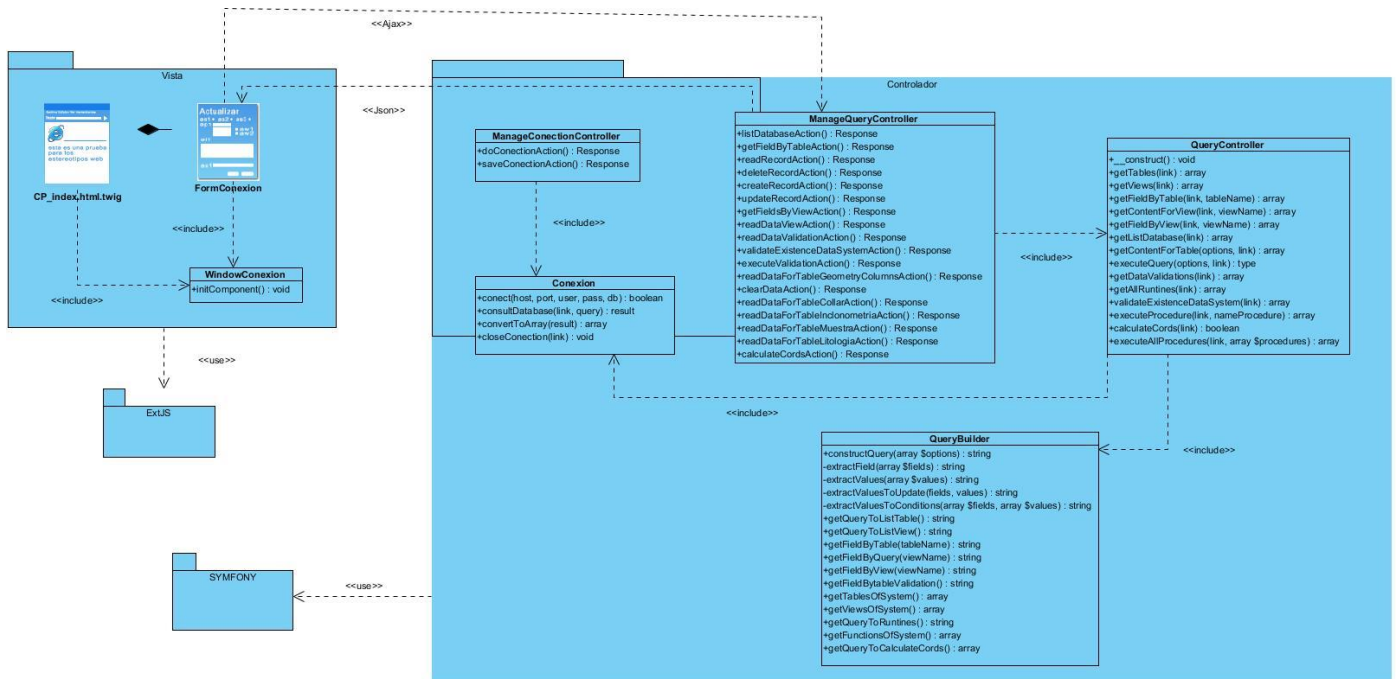


Figura 7. Diagrama de clases del diseño para el CU Establecer Conexión.

## Descripción de clases

Cada uno de los diagramas de clases del diseño que emplean el *framework* Symfony poseen clases que son comunes para la mayoría de ellos, tanto las que son JavaScript como las que son PHP. A continuación serán descritas las clases fundamentales del diagrama y las relaciones de comunicación que se realizan entre ellas.

- Clase ManageConectionController: Es la encargada de realizar y salvar la conexión.
- Clase Conexión: Es la encargada de conectarse a la Base de datos.
- Clase ManageQueryController: Tiene como propósito la lectura, inserción, modificación de los datos etc.
- Clase QueryController: Tiene como propósito recibir las peticiones del controlador y mandar a construir y ejecutar las consultas.
- Clase QueryBuilder: Es la encargada de construir las consultas.

### 4.5 Modelo de despliegue

El modelo de despliegue es un modelo de objetos que ilustra la distribución física del sistema en términos de cómo está distribuida la funcionalidad entre los nodos de cómputo. (30) En la figura 7 se muestra el diagrama de despliegue correspondiente a la solución propuesta.

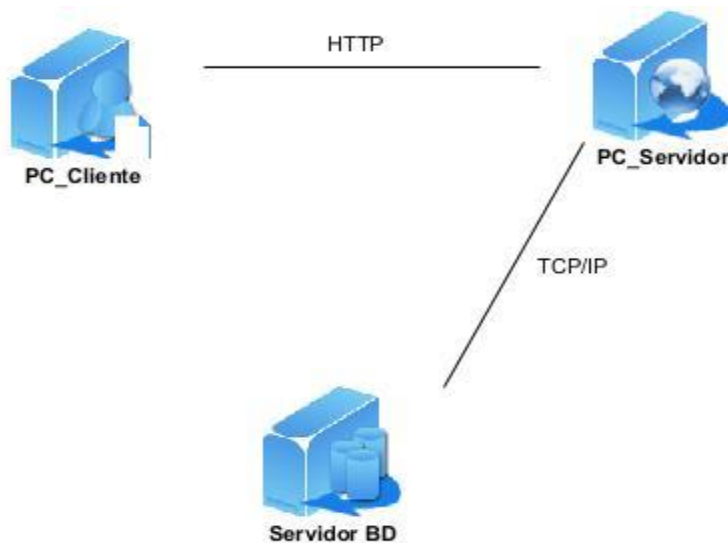
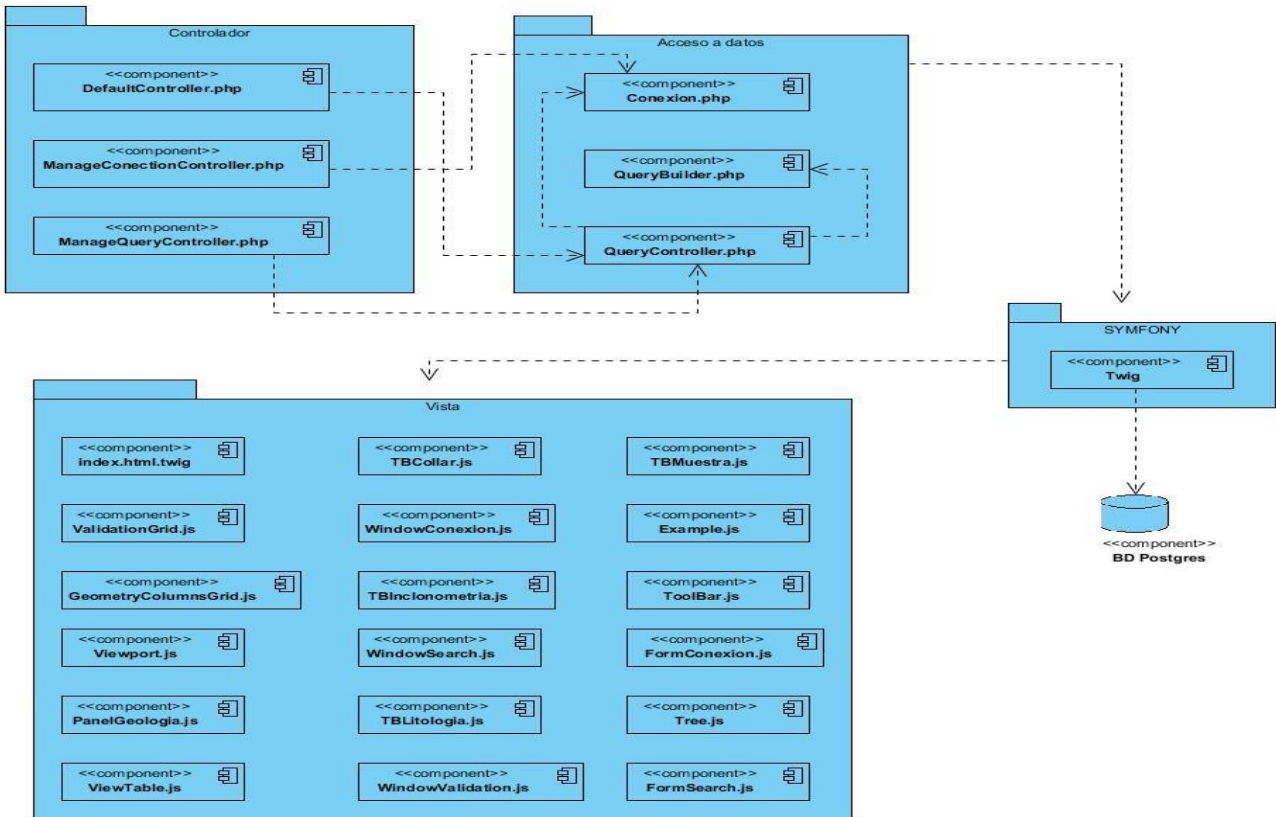


Figura 8. Diagrama de despliegue.

- Pc\_Cliente: Su función es acceder a la aplicación e interactuar con ella. Para ello hará uso del protocolo de transferencia de Hipertexto (HTTP). Contará con cualquier distribución tanto de Windows como de Linux para acceder a la misma, así como el navegador web.
- PC Servidor: Para la instalación del sistema se requiere de un servidor web, Apache. En este servidor se encontrará alojada la aplicación, el cual será accedido a través de un navegador web a partir de una pc cliente. Este será el encargado de escuchar y satisfacer las peticiones HTTP, y devolviendo así mismo la solicitud de la petición al navegador que la realizó. También será el encargado de generar toda la información referente a la aplicación.
- PC Base de datos: Se requiere además de un servidor de base de datos donde se ejecutará el SGBD PostgreSQL. Estos dos servidores se comunicarán a través del protocolo de transmisión de datos TCP/IP.

### 4.6 Diagrama de Componentes

Los diagramas de componentes muestran las dependencias lógicas entre los componentes de *software*, estos pueden ser código fuente, binarios o ejecutables. Los componentes representan los elementos de *software* utilizados en la fabricación de aplicaciones informáticas, desde bibliotecas cargadas dinámicamente, paquetes o archivos. Estos se representan como un grafo de componentes unidos a través de relaciones de dependencia. (30) El diagrama de la figura 9 es un diagrama genérico que representa los principales componentes del sistema utilizados en la implementación.



**Figura 9. Diagrama de componentes de la aplicación web.**

## 4.7 Modelo de Prueba

Una vez finalizada la implementación del sistema, se hace necesario comprobar que su funcionamiento sea adecuado, por lo que se debe verificar que las operaciones internas se ajustan a las especificaciones planteadas. Para lograrlo se aplicarán las pruebas de caja negra.

Esta técnica de prueba se centra en los requisitos funcionales del software, permitiendo obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Las mismas son de gran utilidad pues permiten encontrar en el *software* varios tipos de errores: funciones incorrectas o ausentes, errores de interfaz, errores en estructuras de datos o en accesos a bases de datos externas, errores de rendimiento y errores de inicialización y de terminación. Además, con su aplicación se comprueba si el sistema es sensible a ciertos valores de entrada y qué efectos tiene sobre el mismo la

## DISEÑO, IMPLEMENTACIÓN Y PRUEBAS DE LA SOLUCIÓN PROPUESTA

entrada de ciertas combinaciones de datos, (19) aspectos que son necesarios a la hora de validar el buen funcionamiento del sistema.

En concreto se hará uso de la técnica de partición equivalente, que se basa en una evaluación de las clases de equivalencia para una condición de entrada. (19)

A continuación se realizará el diseño de caso de prueba para el caso de uso Establecer conexión.

### Diseño de las pruebas al sistema

Diseño de las pruebas al sistema

DCP<sup>24</sup> Establecer conexión

Descripción general: Permitir al minero conectarse a la base de datos.

Condiciones de ejecución: La base de datos se encuentre creada.

Escenarios a probar en el Caso de Uso: Establecer conexión.

**Tabla 3. Escenarios del caso de uso Establecer conexión**

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1 Establecer conexión	EC 1.1 Establecer conexión con éxito.	El sistema muestra una interfaz con la configuración de las propiedades generales de la conexión.	Syam web/botón Configurar conexión.
	EC 1.2 Establecer conexión con fallos.	El sistema muestra un mensaje de error: "Verifique que los datos sean correctos."	Syam web/botón Configurar conexión.

Descripción de las variables.

<sup>24</sup> Diseño de Caso de Prueba

Tabla 4. Variables del caso de uso Establecer conexión

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Servidor	Campo de texto	no	Este atributo recibe cadena de caracteres refiriéndose al nombre del servidor o una dirección.
2	Puerto	Campo numérico	no	Este atributo recibe campos numéricos refiriéndose al puerto de la base de datos.
3	Usuario	Campo de texto	no	Este atributo recibe una cadena de caracteres, refiriéndose al nombre del usuario.
4	Contraseña	Campo de texto	no	Este atributo recibe una cadena de caracteres, los valores de la cadena no serán visibles para el Minero.
5	Base de Datos	Lista desplegable	no	Este atributo mostrará una lista con los nombres de las bases de datos que se encuentren

## DISEÑO, IMPLEMENTACIÓN Y PRUEBAS DE LA SOLUCIÓN PROPUESTA

			disponibles.
--	--	--	--------------

Matriz de datos.

SC 1 Establecer conexión.

**Tabla 5. Matriz del caso de uso Establecer conexión**

ID del escenario	Escenario	Variable 1 Servidor	Variable 2 Puerto	Variable 3 Usuario	Variable 4 Contraseña	Variable 5 Base de Datos	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Establecer conexión con éxito.	V <sup>25</sup>	V	V	V	V	El sistema se pudo conectar a la base de datos satisfactoriamente.	Satisfactorio.
EC1.2	Establecer conexión con fallos	I <sup>26</sup>	V	V	V	I	El sistema mostró un mensaje de error	No satisfactorio.
		V	I	V	V	I	"Verifique que los datos sean correctos"	
		V	V	I	V	I		

<sup>25</sup>Valido

<sup>26</sup>Invalido



		V	V	V	I	I	debido a q los campos fueron mal entrados.	
--	--	---	---	---	---	---	---	--

### 4.8 Conclusiones Parciales

Los artefactos generados en el presente capítulo constituyen la base sobre la que se construye la aplicación web para las empresas mineras cubanas. A través del modelo de diseño se logró representar la estructura del sistema teniendo en cuenta los framework sobre los que se desarrolla. Se cuenta con una especie de guía para la instalación del sistema como resultado de generar el modelo de despliegue con los diferentes nodos físicos donde finalmente se instalará el mismo. El modelo de componentes diseñado permitió representar los elementos antes descritos en el diseño en términos de componentes de *software*, permitiendo mostrar así los componentes necesarios para la implementación del sistema. Además, las pruebas realizadas al *software* han mostrado su correcto funcionamiento.

### Conclusiones Generales

Al término de la presente investigación se consideran cumplidos los objetivos trazados al tener desarrollada una aplicación web capaz de gestionar los datos geológicos mineros de un yacimiento mineral. La misma cumple todos los requisitos planteados para su desarrollo, logrando contar con un sistema informático cuyas funcionalidades básicas muestran los resultados esperados.

El sistema fue desarrollado utilizando herramientas multiplataforma, lo que permite que pueda ser desplegada sobre entornos libres. El hecho de haber sido desarrollada sobre una arquitectura web garantiza la explotación de los beneficios de la red, minimizando el costo en cuanto a *hardware* y *software* donde se utilice el producto.

Se creó además una documentación técnica relacionada con la herramienta, en la misma se detallan todos los artefactos generados a partir de la aplicación de la Metodología de Desarrollo de *Software* RUP. Este trabajo representa un aporte importante para las empresas mineras cubanas ya que permite la gestión de la información de los pozos de perforación de una forma atractiva. Logrando elevar la calidad del trabajo en las mismas.

### Recomendaciones

A partir de los beneficios que proporciona este trabajo de diploma, se proponen las siguientes recomendaciones:

- Que se despliegue la primera versión del sistema junto con sus tecnologías correspondientes, en las diferentes empresas mineras cubanas.
- Agregarle otras funcionalidades como la visualización de los pozos en 2D y3D, además de realizarse la implementación de las mismas.
- Realizar una ayuda que facilite la comprensión del sistema para los usuarios.

### Bibliografía Referenciada

1. **Amigot Mora, Thais.** Propuesta de Arquitectura para el Sistema de Software Minero Geolmin. 2011. Disponible en: [http://repositorio\\_institucional.uci.cu/jspui/bitstream/ident/TD\\_04547\\_11/1/TD\\_04547\\_11.pdf](http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_04547_11/1/TD_04547_11.pdf)
2. **Suárez Morales, Dagoberto A., Quesada Rodríguez, Eddy.D.** Sistema Minero Cubano: Un paso de avance hacia la soberanía tecnológica. 2011. Disponible en: <http://revistageonews.wordpress.com/>.
3. **deConceptos.com.** Concepto de Geología. [En línea] 2012. [Citado el: 15 de Noviembre de 2012.] Disponible en: <http://deconceptos.com/ciencias-naturales/geologia>.
4. **Definiciones, Conceptos.** Concepto de, Definición de. [En línea] 15 de Noviembre de 2012. Disponible en: <http://conceptodefinicion.de/mineria/>.
5. **González Partida, Dr. Eduardo.** Yacimientos minerales. [En línea] [Citado el: 15 de Noviembre de 2012.] Disponible en: [http://www.geociencias.unam.mx/~rmolina/Diplomado/yacimientos/yacimientos\\_1.html](http://www.geociencias.unam.mx/~rmolina/Diplomado/yacimientos/yacimientos_1.html).
6. **Vozdvizhenski, B.I, Golubintsev, O.N y Novozhilov, A.A.** *Perforación de Exploración.* s.l. : Mir. Moscú.
7. **Sánchez Falero, Héctor René.** Gestión de Nomencladores para el Programa Nacional de Informatización del Conocimiento Geológico. 2009.
8. Tendencias, Tecnologías y soluciones para la exploración minera. [En línea] [Citado el: 17 de Noviembre de 2012.] Disponible en: [http://www.geosoft.com/media/uploads/resources/brochures/fp\\_ebook\\_2011\\_09\\_es\\_web.pdf](http://www.geosoft.com/media/uploads/resources/brochures/fp_ebook_2011_09_es_web.pdf).
9. **Innova, Web.** Web Innova, Soluciones en Tecnologías y Comunicaciones. [En línea] [Citado el: 17 de Noviembre de 2012.] Disponible en: <http://www.webinnova.com.co>.
10. **Alarcon de Quesada, Ricardo.** Asamblea Nacional del Poder Popular. [En línea] [Citado el: 20 de Noviembre de 2012.] Disponible en: <http://www.parlamentocubano.cu>.
11. Informática Aplicada a la Minería. [En línea] [Citado el: 20 de Noviembre de 2012.] Disponible en: <http://www.cae.com/en/mining/products.asp>.

12. Causa el bloqueo cuantiosos daños a la minería cubana. [En línea] [Citado el: 22 de Noviembre de 2012.] Disponible en: <http://www.cubavsbloqueo.cu>.
13. Software Propietarios. [En línea] [Citado el: 22 de Noviembre de 2012.] Disponible en: [http://www.slideshare.net/erito22/software-propietario-13701264?from\\_search=10](http://www.slideshare.net/erito22/software-propietario-13701264?from_search=10).
14. **Len, BASS, Paul, CLEMENTS y Rick, KAZMAN.** *Software architecture in practice.* s.l. : Addison Wesley.
15. **ISW, Colectivo de Asignatura.** *Diseño, arquitectura e integración de sistemas. Arquitectura Cliente-Servidor.*
16. **Carlos, Reynoso.** *Estilos y Patrones en la Estrategia de Arquitectura de Microsoft Version 1.0.*
17. **Vignaga, Andres y Perovich, Daniel.** *Enfoque Metodológico para el desarrollo basado en componentes.*
18. **González Cornejo, José Enrique.** DocIRs. [En línea] [Citado el: 3 de Diciembre de 2012.] Disponible en: <http://www.docirs.cl/uml.htm>.
19. **Pressman, Roger.** *Ingeniería del Software: Un enfoque práctico. 5ta Edición.*
20. PHP: Hypertext Preprocessor. [En línea] [Citado el: 10 de Diciembre de 2012.] Disponible en: <http://www.php.net/manual/es/intro-what-is.php>.
21. Libros Web. [En línea] [Citado el: 11 de Diciembre de 2012.] Disponible en: <http://www.librosweb.es/javascript/capitulo1.html>.
22. Bases de datos Weblog. [En línea] [Citado el: 12 de Diciembre de 2012.] Disponible en: <http://basesdedatos.wordpress.com/10-%C2%BFque-es-un-sgbd/>.
23. **PostgreSQL, Equipo Global de Desarrollo de.** PostgreSQL 9.1 Press Kit. [En línea] [Citado el: 8 de Enero de 2013.] Disponible en: <http://www.postgresql.org/about/press/presskit91/es/>.
24. **Ubuntu, De Guía.** PgAdmin III. [En línea] [Citado el: 8 de Enero de 2013.] Disponible en: [http://www.guia-ubuntu.com/index.php?title=PgAdmin\\_III](http://www.guia-ubuntu.com/index.php?title=PgAdmin_III).

25. Ciberaula. [En línea] [Citado el: 10 de Enero de 2013.] Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro](http://linux.ciberaula.com/articulo/linux_apache_intro).
26. **Potencier, Fabien y Zaninotto, Francois.** *Symfony la guía definitiva.* [En línea] [Citado el: 11 de Enero de 2013.] Disponible en: <http://www.slideshare.net/nereosa/symfony-guia-definitiva>.
27. **Ardissone, Juan.** *Maestros del Web. Introducción a Symfony 2.* [En línea] [Citado el: 11 de Enero de 2013.] Disponible en: <http://www.maestrosdelweb.com/editorial/curso-symfony2-introduccion-instalacion/>.
28. **Utria, Dianet y Santiesteban, Jose Carlos.** *Subsistema de consulta de la información referente a los pozos en la Oficina Nacional de Recursos Minerales.* . 2008.
29. NetBeans. [En línea] [Citado el: 12 de Enero de 2013.] Disponible en: [https://netbeans.org/index\\_es.html](https://netbeans.org/index_es.html).
30. **JACOBSON, Ivar.** *El Proceso Unificado de Desarrollo de Software.* 2000.
31. **Sommerville, Ian.** *Ingeniería de Software. 7ma Edición.* 2005.

**Bibliografía**

1. **Amigot Mora, Thais.** Propuesta de Arquitectura para el Sistema de Software Minero Geolmin. 2011. Disponible en: [http://repositorio\\_institucional.uci.cu/jspui/bitstream/ident/TD\\_04547\\_11/1/TD\\_04547\\_11.pdf](http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_04547_11/1/TD_04547_11.pdf)
2. **Alarcon de Quesada, Ricardo.** Asamblea Nacional del Poder Popular. [En línea] [Citado el: 20 de Noviembre de 2012.] Disponible en: <http://www.parlamentocubano.cu>.
3. **Ardissone, Juan.** Maestros del Web. *Introducción a Symfony 2*. [En línea] [Citado el: 11 de Enero de 2013.] Disponible en: <http://www.maestrosdelweb.com/editorial/curso-symfony2-introduccion-instalacion/>.
4. Bases de datos Weblog. [En línea] [Citado el: 12 de Diciembre de 2012.] Disponible en: <http://basesdedatos.wordpress.com/10-%C2%BFque-es-un-sgbd/>.
5. Causa el bloqueo cuantiosos daños a la minería cubana. [En línea] [Citado el: 22 de Noviembre de 2012.] Disponible en: <http://www.cubavsbloqueo.cu>.
6. **Carlos, Reynoso.** *Estilos y Patrones en la Estrategia de Arquitectura de Microsoft Version 1.0*.
7. Ciberaula. [En línea] [Citado el: 10 de Enero de 2013.] Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro](http://linux.ciberaula.com/articulo/linux_apache_intro).
8. **deConceptos.com.** Concepto de Geología. [En línea] 2012. [Citado el: 15 de Noviembre de 2012.] Disponible en: <http://deconceptos.com/ciencias-naturales/geologia>.
9. **Definiciones, Conceptos.** Concepto de, Definición de. [En línea] 15 de Noviembre de 2012. Disponible en: <http://conceptodefinicion.de/mineria/>.
10. **González Partida, Dr. Eduardo.** Yacimientos minerales. [En línea] [Citado el: 15 de Noviembre de 2012.] Disponible en: [http://www.geociencias.unam.mx/~rmolina/Diplomado/yacimientos/yacimientos\\_1.html](http://www.geociencias.unam.mx/~rmolina/Diplomado/yacimientos/yacimientos_1.html).
11. **González Cornejo, José Enrique.** DocIRS. [En línea] [Citado el: 3 de Diciembre de 2012.] Disponible en: <http://www.docirs.cl/uml.htm>.

12. **Innova, Web.** Web Innova, Soluciones en Tecnologías y Comunicaciones. [En línea] [Citado el: 17 de Noviembre de 2012.] Disponible en: <http://www.webinnova.com.co>.
13. Informática Aplicada a la Minería. [En línea] [Citado el: 20 de Noviembre de 2012.] Disponible en: <http://www.cae.com/en/mining/products.asp>.
14. **ISW, Colectivo de Asignatura.** *Diseño, arquitectura e integración de sistemas. Arquitectura Cliente-Servidor.*
15. **JACOBSON, Ivar.** *El Proceso Unificado de Desarrollo de Software.* 2000.
16. **Len, BASS, Paul, CLEMENTS y Rick, KAZMAN.** *Software architecture in practice.* s.l. : Addison Wesley.
17. Libros Web. [En línea] [Citado el: 11 de Diciembre de 2012.] Disponible en: <http://www.librosweb.es/javascript/capitulo1.html>.
18. NetBeans. [En línea] [Citado el: 12 de Enero de 2013.] Disponible en: [https://netbeans.org/index\\_es.html](https://netbeans.org/index_es.html).
19. **Pressman, Roger.** *Ingeniería del Software: Un enfoque práctico. 5ta Edición.*
20. PHP: Hypertext Preprocessor. [En línea] [Citado el: 10 de Diciembre de 2012.] Disponible en: <http://www.php.net/manual/es/intro-what-is.php>.
21. **PostgreSQL, Equipo Global de Desarrollo de.** PostgreSQL 9.1 Press Kit. [En línea] [Citado el: 8 de Enero de 2013.] Disponible en: <http://www.postgresql.org/about/press/presskit91/es/>.
22. **Potencier, Fabien y Zaninotto, Francois.** *Symfony la guía definitiva.* [En línea] [Citado el: 11 de Enero de 2013.] Disponible en: <http://www.slideshare.net/nereosa/symfony-guia-definitiva>.
23. **Sánchez Falero, Héctor René.** *Gestión de Nomencladores para el Programa Nacional de Informatización del Conocimiento Geológico.* 2009.
24. Software Propietarios. [En línea] [Citado el: 22 de Noviembre de 2012.] Disponible en: [http://www.slideshare.net/erito22/software-propietario-13701264?from\\_search=10](http://www.slideshare.net/erito22/software-propietario-13701264?from_search=10).



25. **Sommerville, Ian.** *Ingeniería de Software. 7ma Edición.* 2005.
26. **Suárez Morales, Dagoberto A, Quesada Rodríguez, Eddy.D.** Sistema Minero Cubano: Un paso de avance hacia la soberanía tecnológica. 2011. Disponible en: <http://revistageonews.wordpress.com/>.
27. Tendencias, Tecnologías y soluciones para la exploración minera. [En línea] [Citado el: 17 de Noviembre de 2012.] Disponible en: [http://www.geosoft.com/media/uploads/resources/brochures/fp\\_ebook\\_2011\\_09\\_es\\_web.pdf](http://www.geosoft.com/media/uploads/resources/brochures/fp_ebook_2011_09_es_web.pdf).
28. **Ubuntu, De Guía.** PgAdmin III. [En línea] [Citado el: 8 de Enero de 2013.] Disponible en: [http://www.guia-ubuntu.com/index.php?title=PgAdmin\\_III](http://www.guia-ubuntu.com/index.php?title=PgAdmin_III).
29. **Utria, Dianet y Santiesteban, Jose Carlos.** *Subsistema de consulta de la información referente a los pozos en la Oficina Nacional de Recursos Minerales.* 2008.
30. **Vozdvizhenski, B.I, Golubintsev, O.N y Novozhilov, A.A.** *Perforación de Exploración.* s.l. : Mir. Moscú.
31. **Vignaga, Andres y Perovich, Daniel.** *Enfoque Metodológico para el desarrollo basado en componentes.*

## Anexos

## Anexo A: Descripción textual de los casos de uso del sistema.

## Descripción del caso de uso: Gestionar datos tabla

Tabla 6. Descripción del caso de uso Gestionar datos tabla

<b>Caso de Uso:</b>	Gestionar datos tabla
<b>Actores:</b>	Minero
<b>Propósito:</b>	Posibilitar la inserción, modificación o eliminación de registros (datos de los pozos de perforación).
<b>Resumen:</b>	El caso de uso inicia cuando el minero selecciona la tabla a la cual desea realizar acciones para adicionar, modificación y suprimir registros.
<b>Precondiciones:</b>	El minero debe estar previamente conectado a la base de datos.
<b>Postcondiciones:</b>	Se adiciona, modifica o elimina los registros de la tabla satisfactoriamente.
<b>Referencias:</b>	RF5, RF6, RF7.
<b>Prioridad:</b>	Crítico.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Minero selecciona la tabla.	El sistema muestra un listado con los registros la tabla seleccionada.
2. <ul style="list-style-type: none"> <li>• El Minero selecciona la opción Adicionar.</li> <li>• El Minero selecciona la opción Eliminar.</li> <li>• El Minero da doble clic sobre el registro que desea modificar.</li> </ul>	<ul style="list-style-type: none"> <li>• Si el Minero selecciona la opción Adicionar ver la sección 1.</li> <li>• Si el Minero selecciona la opción Eliminar ver la sección 2.</li> <li>• Si el da doble clic sobre el registro que desea modificar ver la sección 3.</li> </ul>
<b>Sección 1.</b> Adicionar.	

3. El Minero selecciona la opción Adicionar.	4. El sistema muestra una nueva fila para que el actor pueda introducir los nuevos datos.
5. El Minero selecciona la opción Editar.	6. El sistema guarda el nuevo registro en la tabla seleccionada.
5. El Minero selecciona la opción Cancelar.	6. Es sistema cancela la operación para adicionar un nuevo registro y actualiza la tabla.
<b>Sección 2. Eliminar.</b>	
3. El Minero selecciona la opción Eliminar.	4. El sistema elimina la fila seleccionada y muestra el mensaje "Operación realizada satisfactoriamente".
<b>Sección 2. Modificar.</b>	
3. El Minero da doble clic sobre el registro que desea modificar.	4. El sistema habilita un formulario con los datos del registro seleccionado.
5. El Minero selecciona la opción Editar.	6. El sistema guarda en la base de datos los datos actuales del registro.
5. El Minero selecciona la opción Cancelar.	6. Es sistema cancela la operación para modificar el registro y oculta el formulario.
<b>Flujo Alternativo de Eventos</b>	

### Descripción del caso de uso: Calcular Coordenadas

Tabla 7. Descripción del caso de uso Calcular Coordenadas

<b>Caso de Uso:</b>	Calcular Coordenadas
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permitir a partir de los datos presentes en la tabla de collares y de inclinometría, transformarlos a datos geo-espaciales para su posterior visualización.
<b>Resumen:</b>	El caso de uso inicia cuando el minero ejecuta la opción "Calcular

	coordenadas” en la barra de herramientas de la aplicación y se calculan las coordenadas geométricas.
<b>Precondiciones:</b>	Las tablas de collares e inclinometría deben poseer datos.
<b>Postcondiciones:</b>	
<b>Referencias:</b>	RF 16, RF 17, RF 18.
<b>Prioridad:</b>	Medio.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Minero selecciona la opción en la barra de herramientas “Calcular Coordenadas”.	2. El sistema calcula las coordenadas geométricas y muestra el mensaje “Operación realizada satisfactoriamente”.
<b>Flujo Alternativo de Eventos</b>	

### Descripción del caso de uso: Validar Datos

Tabla 8. Descripción del caso de uso Validar Datos

<b>Caso de Uso:</b>	Validar Datos
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permitir al minero validar y visualizar los pozos de perforación que presentan errores en la base de datos.
<b>Resumen:</b>	El caso de uso inicia cuando el minero selecciona la opción de “Validar datos” en la barra de herramientas de la aplicación y selecciona las validaciones que desee realizar.
<b>Precondiciones:</b>	Al menos una tabla debe contener datos.
<b>Postcondiciones:</b>	Se deben visualizar los errores encontrados.
<b>Referencias:</b>	RF 8, RF 9, RF 10, RF 11, RF 12, RF 13, RF 19.

<b>Prioridad:</b>	Medio.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Minero selecciona la opción "Validar datos" en la barra de herramientas.	2. El sistema muestra una nueva pestaña con las validaciones disponibles
3. El Minero selecciona la opción "Ejecutar validaciones realizadas".	4. El sistema valida los registros de las tablas y muestra los errores encontrados.
3. El Minero selecciona la opción "Recargar".	4. El sistema actualiza las validaciones disponibles y las muestra en la pestaña actual.
<b>Flujo Alternativo de Eventos</b>	

### Descripción del caso de uso: Filtrar datos

Tabla 9. Descripción del caso de uso Filtrar datos

<b>Caso de Uso:</b>	Filtrar datos
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permitir al Minero encontrar un registro específico en la tabla seleccionada.
<b>Resumen:</b>	El caso de uso inicia cuando el Minero selecciona la opción de "Búsqueda normal" o "Búsqueda Avanzada".
<b>Precondiciones:</b>	Debe tener activa una pestaña de alguna tabla.

<b>Postcondiciones:</b>	El sistema debe mostrar los registros que coincidan con la opción de búsqueda seleccionada.
<b>Referencias:</b>	RF 4.
<b>Prioridad:</b>	Medio.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1.</p> <ul style="list-style-type: none"> <li>El Minero introduce criterio para la Búsqueda normal y selecciona esta opción.</li> <li>El Minero Selecciona la opción de “Búsqueda avanzada”</li> </ul>	<p>Si el minero selecciona la opción de “Búsqueda Normal” ver sección 1.</p> <p>Si el minero selecciona la opción de “Búsqueda Avanzada” ver sección 2.</p>
<b>Sección 1. Búsqueda normal.</b>	
<p>2. El Minero introduce criterio para la Búsqueda normal y selecciona esta opción.</p>	<p>3. El sistema muestra los registros encontrados que coinciden con el criterio de búsqueda introducido.</p>
<b>Sección 2. Búsqueda avanzada.</b>	
<p>2.</p> <ul style="list-style-type: none"> <li>El minero selecciona la opción de “B Búsqueda Avanzada”.</li> <li>El Minero</li> </ul>	<p>3. El sistema muestra un formulario con las opciones de búsqueda avanzada.</p>

selecciona la opción de búsqueda avanzada “Por columnas”.	
<p>4.</p> <ul style="list-style-type: none"> <li>• El Minero selecciona la opción de búsqueda avanzada “Por columnas”.</li> <li>• El Minero selecciona la opción de búsqueda avanzada “Por rango numérico”.</li> </ul>	<ul style="list-style-type: none"> <li>• Si el minero secciona la opción de búsqueda avanzada “Por columnas” ver sección 2.1.</li> <li>• Si el minero secciona la opción de búsqueda avanzada “Por rango numérico” ver sección 2.2.</li> </ul>
<b>Sección 2.1.</b> Búsqueda avanzada - Por columnas.	
5. El Minero selecciona la opción de búsqueda avanzada “Por columnas”.	6. El sistema muestra un combobox con los campos de la tabla y un campo de texto para que el usuario introduzca el criterio de búsqueda.
7. El usuario selecciona la opción “Buscar”.	8. El sistema muestra el resulta de la búsqueda.
<b>Sección 2.2.</b> Búsqueda avanzada.	
5. El Minero selecciona la opción de búsqueda avanzada “Por	6. El sistema muestra un combobox con los campos de la tabla que son numéricos y dos campos de textos para que el minero introduzca los valores del intervalo de búsqueda.

rango numérico”.	
7. El usuario selecciona la opción “Buscar”.	8. El sistema muestra el resulta de la búsqueda.
<b>Flujo Alternativo de Eventos</b>	
El usuario selecciona la opción “Buscar”.	El sistema muestra un mensaje de error en los campos de textos si no se ha introducido ningún criterio de búsqueda.

### Descripción del caso de uso: Identificar Tablas

Tabla 10. Descripción del caso de uso Identificar Tablas

<b>Caso de Uso:</b>	Identificar Tablas
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permite al Minero identificar las tablas que contienen datos geométricos.
<b>Resumen:</b>	El caso de uso inicia cuando el Minero selecciona la opción de visualizar “Tablas con columnas geométricas” y el sistema muestra en una pestaña las tablas que cumplen con esta condición.
<b>Precondiciones:</b>	Al menos una tabla debe poseer datos geométricos.
<b>Postcondiciones:</b>	Mostrar las tablas con datos geométricos.
<b>Referencias:</b>	RF 20.
<b>Prioridad:</b>	Baja.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Minero selecciona la opción “Tablas con columnas geométricas” en la barra de herramientas.	2. El sistema muestra en una nueva pestaña las tablas que poseen datos geométricos.



<b>Flujo Alternativo de Eventos</b>

### Descripción del caso de uso: Refrescar Información

Tabla 11. Descripción del caso de uso Refrescar Información

<b>Caso de Uso:</b>	Refrescar Información
<b>Actores:</b>	Minero
<b>Propósito:</b>	Permite al minero visualizar los cambios realizados en otras estaciones de trabajo.
<b>Resumen:</b>	El caso de uso inicia cuando el Minero de recargar los datos de la tabla actualmente visualizada.
<b>Precondiciones:</b>	Debe tener activa alguna tabla en pestaña.
<b>Postcondiciones:</b>	Visualizar los registros de la tabla.
<b>Referencias:</b>	RF 14, RF 15.
<b>Prioridad:</b>	Media.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. <ul style="list-style-type: none"> <li>• El Minero selecciona la opción de “Recargar”.</li> <li>• Si el Minero selecciona la opción “Recargar periódicamente”.</li> </ul>	<ul style="list-style-type: none"> <li>• Si el Minero selecciona la opción “Recargar” ver sección 1.</li> <li>• Si el Minero selecciona la opción “Recargar periódicamente” ver sección 2.</li> </ul>
<b>Sección 1. Recargar.</b>	
2. El Minero selecciona la opción de	El sistema visualiza los registros de la tabla actualizados.

---

“Recargar”.	
<b>Sección 2.</b> Recargar periódicamente.	
2. El Minero selecciona la opción de “Recargar”.	El sistema actualiza periódicamente (1 minuto) los registros de la tabla.
<b>Flujo Alternativo de Eventos</b>	

Anexo B: Diagramas de clases del diseño.

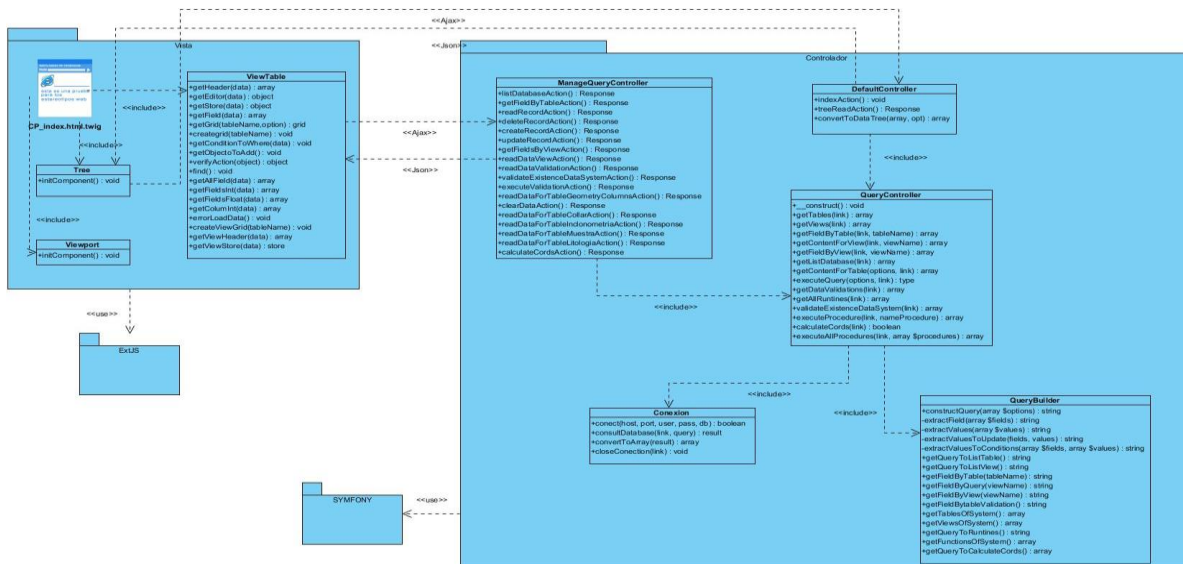


Figura 10. Diagrama de clases del diseño para el CU Gestionar datos Tablas.

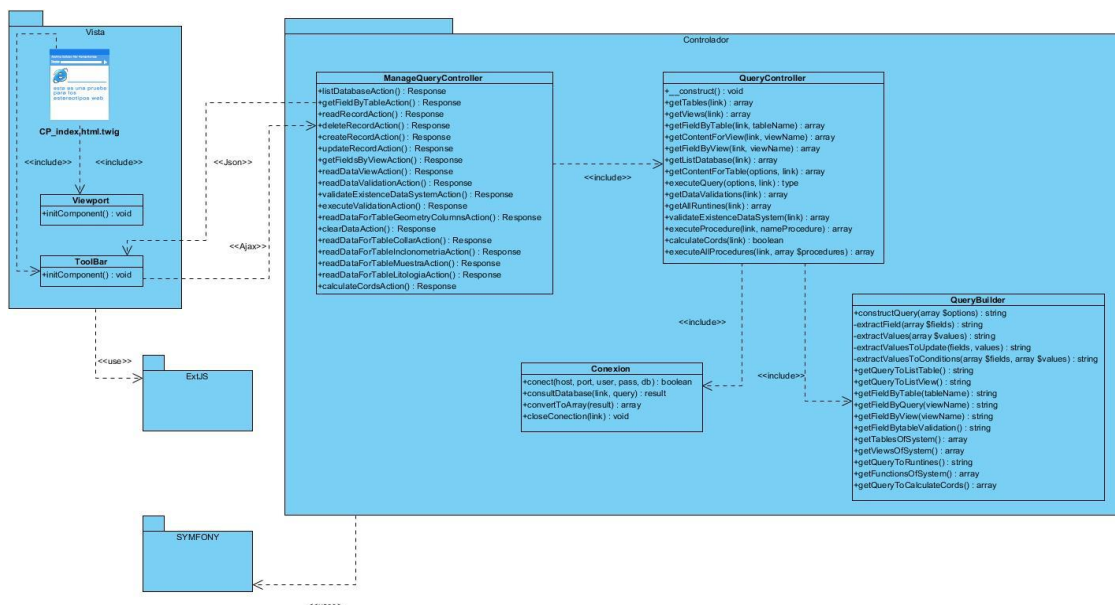


Figura 11. Diagrama de clases del diseño para el CU Calcular Coordenadas.

