

**UNIVERSIDAD DE LAS CIENCIAS
INFORMÁTICAS
Facultad 6**



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

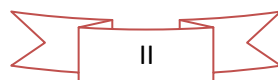
Título: Análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin.

Autora: Madelaynis Quesada Urquiza.
Tutora: Ing. Lisbeth.O López Verdecie.

**Ciudad de la Habana, junio 2011
"Año 53 de la Revolución"**

“Por este mundo pasaré solamente una vez, si hay una buena obra que pueda hacer, si hay una buena palabra que pueda decir; haré esa buena obra y diré esa buena palabra, pues ya nunca volveré a pasar por aquí”

(Edmundo D’ Amicis “Corazón”)



Declaración de autoría

Declaro que soy la única autora de este trabajo y autorizo 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 29 días del mes de junio del año 2011.

Firma del autor:

Madelaynis Quesada Urquiza

Firma del tutor:

Ing. Lisbeth.O López Verdecie



Datos del contacto

Ing. Lisbeth.O López Verdecie

Profesora graduada en el año 2008 de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Profesora de la Facultad # 6. Actualmente imparte la asignatura de Práctica Profesional 1. Desempeña el rol de analista principal en el proyecto Minería del Centro de Desarrollo de Geoinformática y Sistemas Digitales (GEySED) en dicha facultad.

Correo electrónico: lolopez@uci.cu

Dedicatoria

Dedico de manera muy especial este logro al ser más grande de mi vida, mi madre, la persona que con todo su amor, apoyo, dedicación, comprensión y confianza, me dio las fuerzas para seguir adelante y hacer de mi lo que hoy soy, una ingeniera, por ser aquella persona que si alguna vez me faltara, no sabría cómo vivir, a ti te debo mi vida, gracias por existir. Siempre serás mi Ángel mami.

A una persona muy especial que me ha dado su cariño, su apoyo, su ayuda en los buenos y en los malos momentos; y sobre todas las cosas nunca ha dejado de confiar en mí, mi novio Diubert.

A mis abuelos Nancy y Ulicer

A mi bisabuela Rachel

A mis hermanos Eric y Leanis y a mi prima Sandra

A mi padrastro por comportarse como un padre.

También quiero dedicarle este trabajo a alguien que quiero mucho y que lamentablemente la vida no fue tan larga como yo hubiese querido, pero seguro hoy esté donde esté, se siente muy orgulloso de mí, mi tío Cesar.

Agradecimientos

Quiero agradecer especialmente a:

Mi mamá: por ser madre y padre, por su infinito amor, su apoyo incondicional y por confiar en mí, incluso cuando yo misma no confiaba, porque este es su sueño hecho realidad, por ser la luz que me sirvió y servirá de guía a lo largo de mi vida, por los sacrificios que ha hecho para presenciar este momento de mi vida y por todo lo que ha hecho para que yo pudiera salir adelante en mi carrera y para que un día como hoy pueda estar en este lugar, especialmente por regalarme la oportunidad de vivir, gracias por existir y por ser parte de mi vida.

Gracias mima I ♥ you.

A mi novio, ese gran hombre, compañero y amigo que es Diubert Fernández Montero que ha compartido conmigo estos largos años y que nunca ha dejado de creer en mí y por ayudarme a vencer todos los contratiempos que el destino me ha puesto. Por soportarme, por comprenderme, por preocuparse más que yo y animarme a seguir aun cuando yo decía que no podía, por darme tanta felicidad en estos últimos años de la carrera, porque si terminé esta tesis es gracias ti. Muchísimas gracias por todo mi amor.

A mis abuelos Nancy y Ulicer por quererme tanto y estar ahí, siempre en el momento en que los necesito, por su apoyo incondicional por ser tan buenos y comprensivos, por ayudarme y apoyarme siempre, los quiero mucho, ustedes son mis padres.

A mi bisabuela Rachel por todo el sacrificio, apoyo y dedicación desde pequeña y por ser como una madre más para mí.

A mis hermanos Eric y Leanis y a mi prima Sandra, que son la motivación principal de mi buen comportamiento al intentar darles un buen ejemplo a seguir y por ser de las personas más importantes en mi vida.

A mi padrastro por ser como un padre para mí.

A toda mi familia por darme, con su confianza, la seguridad de que lograría graduarme exitosamente y porque de una forma u otra me ayudaron hacer realidad mi sueño.

Especialmente a mi amiga Mao que es como una hermana para mí, por compartir tantos los momentos malos como los buenos, por siempre confiar en mí por haber logrado mantener esta amistad

A mis amistades Milayne y Daniel por compartir tantos recuerdos.

A mis compañeras de apartamentos Liusmila, Eylene, Lisi y en especial a Yas y su novio Inda por compartir esa sabrosa comida que solíamos hacer.

A ese clan que solía llamarse la jungla integrado por Yaima, Noli, Nora, Yanet y Eliadys por compartir tantos momentos de risa.

A mi prima yiyi que a pesar de sus locuras me ayudo mucho al inicio de esta carrera con sus consejos.

Agradezco además a todos mis grandes amigos, los que están cerca y los que no, por toda la ayuda, por compartir buenos y malos momentos.

Agradezco a todas las personas con las cuales he compartido estos maravillosos años, a la gente de mi grupo, a todos aquellos que alguna vez me preguntaron cómo me iba en la tesis, a los que me han ayudado de una forma u otra a realizar este sueño.

Y de forma muy especial a mi tutora Lisbeth.O quien me ha apoyado y ayudado en el transcurso de este trabajo porque sin ella sencillamente no podría ser ingeniera.

Agradezco al tribunal de tesis y a mi oponente por sus precisas acotaciones que le dieron a este trabajo mayor calidad y rigor científico y por haber puesto todo su empeño y dedicación en perfeccionar esta investigación hasta convertirla en lo que es hoy.

A los profesores por la ayuda brindada durante el desarrollo de este trabajo y a todos los que han contribuido a mi formación profesional.

A nuestra Revolución que sin ella hubiera sido más difícil mi formación como profesional.

A dios por haberme dado la posibilidad de existir y no abandonarme en todo lo que he vivido.

Resumen

Hoy con los avances de la informática, el sector minero no tiene por qué quedarse atrás, y obviar la ayuda que significa el uso de las soluciones informáticas mineras. Con el objetivo de seguir desarrollando estos sistemas, en la Universidad de las Ciencias Informáticas el proyecto Minería del departamento de Geoinformática se dedica a desarrollar el primer software geológico minero cubano llamado Geolmin con la colaboración de especialistas de entidades geológica mineras como el grupo empresarial Cuba Níquel, la Oficina Nacional de Recursos Minerales y el Instituto de Geología y Paleontología. En esta propuesta se muestra una investigación que arroja el análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin, teniendo como objetivo generar la documentación para una posterior implementación de dicho módulo.

Palabras Claves: análisis, Geolmin, IGP, módulo, ONRM.

Índice

Introducción	13
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1. Introducción	5
1.1. Caracterización de los procesos involucrados en la aplicación del método de prospección y exploración, pozos de sondeos.....	5
1.2. Análisis de soluciones existentes.....	6
1.2.1. Sistemas internacionales	6
1.3. Metodologías de desarrollo de software	8
1.3.1. Proceso Unificado de Modelado (RUP).....	9
1.4. Lenguaje de Modelado Unificado O UML.....	11
1.5. Herramienta CASE	11
1.5.1. Visual Paradigm	12
1.6. Técnica de estimación	12
1.7. Conclusiones Parciales.....	14
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA PROPUESTO.....	15
1. Introducción.....	15
2.1. Modelo de Dominio.....	15
2.1.1. Descripción del diagrama.....	17
2.2. Especificación de los requisitos de software	17
2.2.1. Requisitos funcionales	17
2.2.2. Especificación de requisitos.....	18
2.2.3. Requisitos no funcionales	20
2.3. Descripción de la integración del módulo con el software Geolmin.....	22
2.4. Definición de los casos de uso del sistema	23
2.4.1. Descripción de los actores del sistema	23
2.4.2. Diagrama de caso de uso del sistema	23
2.4.3. Descripción de los casos de usos del sistema.....	24
2.5. Análisis.....	29
2.5.1. Modelo de análisis	29
2.6. Diagramas de colaboración	30
2.7. Conclusiones Parciales	31

CAPÍTULO 3: ESTUDIO DE FACTIBILIDAD Y VALIDACIÓN DEL SISTEMA PROPUESTO.....	32
3. Introducción	32
3.1. Aplicación del método seleccionado para realizar la estimación.....	32
3.2. Técnicas de validación de requisitos.....	40
3.2.1. Aplicación de las técnicas de validación de requisitos	41
3.3. Conclusiones Parciales	43
Conclusiones Generales	44
Recomendaciones	45
Referencias Bibliográficas.....	46
Bibliografía Consultada	48
Anexos.....	50
Anexo#1: Descripción textual de los casos de uso del sistema.	50
Anexo#2: Diagramas de Clases del Análisis.	62
Anexo#3: Diagramas de Colaboración.	64
Anexo#4: Lista de chequeo de requisitos.	66

Índice de tablas y figuras

Tabla 1: Descripción de los actores del sistema.....	23
Tabla 2: Descripción textual del caso de uso: Configurar conexión con la base de datos	24
Tabla 3: Factor de Peso de los Actores sin Ajustar	33
Tabla 4: Factor de Peso de los Casos de Uso sin Ajustar	34
Tabla 5: Factor de Complejidad Técnica.	35
Tabla 6: Factor de Ambiente.	37
Tabla 7: Esfuerzo del Proyecto	39
Tabla 8: Requisitos Funcionales y Caso de Uso del Sistema.....	42
Tabla 9: Actores del Sistema y Casos de Uso del Sistema	43
Figura 1: Modelo de dominio o modelo conceptual	16
Figura 2: Diagrama de Caso de Uso del Sistema.....	24
Figura 3: Diagrama de Clases del Análisis	30
Figura 4: Diagrama de Colaboración.....	31
Figura 5: La validación en el proceso de requisitos	40

Introducción

La minería es una de las actividades más antiguas en la historia del hombre, es por eso que en la actualidad es la actividad económica primaria, además de que juega un papel trascendental en la economía y desarrollo de los países del mundo por la ingente riqueza que ha producido, por lo que su importancia radica en el abastecimiento de los materiales para producir bienes que simplifiquen la vida. Además es una actividad que día a día necesita renovarse por tal motivo es que se sigue haciendo año a año. **(1)**

Pero en el mundo no solo la minería ha obtenido un gran auge, también las Tecnologías de la Información y las Comunicaciones (TIC), que han avanzado a paso agigantado en cada rama de la economía mundial y la sociedad, debido a la constante evolución que han tenido durante los últimos años. Estas además juegan un papel fundamental ya que permiten un mayor procesamiento y obtención de datos que los medios manuales, con un menor rango de equivocación. Debido al uso de las mismas en la rama de la minería es que existen en el mundo varios sistemas mineros entre los que se encuentran Datamine, Geocom, Surpac, Vulcan, los cuales tienen una aplicación que han hecho a la minería más atractiva y le brinda una mayor facilidad a los profesionales que en ella laboran.

Estos sistemas fueron creados con el objetivo de asegurar el éxito del negocio minero, permiten un trabajo rápido, eficiente, generando así resultados auditables y repetibles a lo largo del tiempo, logrando contar en forma clara y amigable con los datos adecuados y oportunos para la toma de decisiones. Pero tienen como inconveniente que son sistemas propietarios al ser las licencias tan costosas, ya que están dirigidas a un sector donde los ingresos compensan la adquisición de estas, además de imposibilitar realizar cambios en su código fuente. Estas limitaciones provocan para Cuba un alto gasto en la economía debido a la utilización de estos sistemas, ya que los que existen aquí no son capaces de cumplir con todas las fases de la actividad minera, como por ejemplo la fase de prospección y exploración, diseño de la explotación y la planificación.

Cuba es un país que cuenta con una gran diversidad de yacimientos minerales, lo que refleja la complejidad geológica del territorio. Es por esto que se hace necesaria la existencia de un software minero en Cuba, además es un país donde planifican metas ambiciosas que están a la altura de los países del primer mundo, y que no se está muy lejos de poder alcanzar. Un ejemplo de esto es la Industria Cubana del Software, que con la participación de la Universidad de las Ciencias Informáticas UCI y otras empresas productoras de software del país están dando grandes pasos.

La UCI creada en el año 2002 bajo los propósitos de la batalla de ideas, juega un papel importante en el desarrollo de software para Cuba y el mundo, por ser una universidad productiva, cuya misión es producir software y servicios informáticos a partir de la vinculación estudio – trabajo como modelo de formación, la

producción de software y servicios informáticos se basa en la integración de los procesos de formación, investigación y producción en torno a una temática para convertirla en una rama productiva. La producción se concentra en el desarrollo de proyectos donde se destacan resultados en las esferas de salud, educación, software libre, teleformación, sistemas legales, realidad virtual, automatización, bioinformática, procesamiento de imágenes y señales digitales, entre otras **(2)**. En la universidad, se cuenta con el centro de Geoinformática y Señales Digitales (GEySED) perteneciente a la facultad # 6, este centro cuenta con varios proyectos entre los que figura en el departamento de Geoinformática el proyecto de Minería.

Este proyecto está dedicado a la conceptualización y desarrollo de soluciones informáticas para la minería, y especialmente a desarrollar el primer software geológico minero cubano llamado (Geolmin) con la colaboración de especialistas de entidades geológico mineras como el grupo empresarial Cuba Níquel, la Oficina Nacional de Recursos Minerales (ONRM) y el Instituto de Geología y Paleontología (IGP). La principal razón de la creación del sistema, es que el país no cuenta con una herramienta potente para el modelado geológico y la planificación minera, ya que para cumplir esta actividad de forma eficiente utiliza sistemas informáticos propietarios, lo cual produce un alto gasto en la economía por la adquisición de su licencia por tener un costo elevado adquirir la misma y por imposibilitar realizar cambios en su código fuente y adaptarlo a otras necesidades, es por eso que cada día más se lucha por impulsar la independencia y la soberanía tecnológica a partir del uso de tecnologías libres. Por lo que surge la necesidad de realizar un software minero que requiera como etapa inicial el estudio de los procesos involucrados en la captura y procesamiento de la información obtenida durante la actividad minera al aplicar los métodos directos e indirectos de prospección y exploración, pozos de sondeos, además debe facilitar la administración de los datos que se necesita manejar.

A partir de la situación problemática descrita anteriormente, se plantea el siguiente **problema a resolver**: ¿Cómo contribuir a la gestión de la información obtenida al aplicar el método de prospección y exploración, pozos de sondeos?

Por lo que se plantea como **objetivo general**: Realizar el análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin.

Lo que trae como **objeto de estudio**: Proceso de gestión de información involucrado en la aplicación del método de prospección y exploración, pozos de sondeos.

Enmarcado en el **campo de acción**: Análisis del proceso de gestión de información involucrado en la aplicación del método de prospección y exploración, pozos de sondeos para el software minero Geolmin.

A partir de lo antes expuesto se formula la siguiente **idea a defender**: El análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin contribuirá al correcto desarrollo del

sistema, logrando mejoras en el proceso de gestión de información involucrado en la aplicación del método de prospección y exploración, pozos de sondeos.

Esto arroja las siguientes **tareas de la investigación**:

- 1 Caracterización de los procesos involucrados en la aplicación del método de prospección y exploración, pozos de sondeos.
- 2 Análisis de las soluciones existentes mineras en el ámbito nacional e internacional.
- 3 Caracterización de la metodología de desarrollo de software y herramientas en las que se apoya la solución, justificando su selección.
- 4 Elaboración de la documentación técnica ingenieril correspondiente al análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin.
- 5 Realización del estudio de factibilidad y la validación del módulo propuesto.

Métodos Científicos

Con el propósito de darle solución a los objetivos trazados durante la investigación, se utilizaron varios métodos científicos:

Métodos Teóricos:

- ✓ **Método Analítico - Sintético:** fue utilizado para descomponer el tema, posibilitando un mejor estudio de la bibliografía encontrada sobre el tema de investigación.
- ✓ **Método Histórico - Lógico:** se utilizó para profundizar en las tendencias, regularidades y cualidades que presenta el objeto de estudio, así como el surgimiento y la evolución de la investigación.
- ✓ **Método Modelación:** se utilizó para modelar los artefactos generados y así obtener un mayor entendimiento con cada uno de los diagramas ya que estos tienen las actividades y procesos del sistema.

Estructura del documento

El documento se ha dividido en tres capítulos. A continuación se hace referencia al contenido expuesto en cada uno de ellos.

Capítulo # 1: "Fundamentación Teórica". En este capítulo se plasman los principales conceptos asociados al dominio del problema, metodología en el que se apoyará el proceso de desarrollo del software que se propone, herramientas que sirven para dar solución al problema, además de hacer una caracterización de los procesos involucrados en la aplicación del método de prospección y exploración, pozos de sondeos.

Capítulo # 2: "Características del sistema propuesto". En este capítulo se especifica todo lo relacionado con el negocio, se exponen los principales conceptos del dominio, se representa el modelo del dominio y se argumenta la especificación de los requisitos de software tanto no funcional como funcional. También se definen los actores y casos de usos del sistema, se hace una descripción textual de cada caso de uso y una representación del diagrama de caso de uso del sistema, además de realizar los diagramas correspondientes al análisis como los diagramas de clases del análisis y los diagramas de colaboración. También se generan todos los artefactos correspondientes como especificación de requerimientos del software, glosario del sistema, modelo de caso de uso del sistema, modelo de dominio, modelo de análisis por ser necesarios para un correcto desarrollo.

Capítulo # 3: "Estudio de factibilidad y validación del sistema propuesto". En este capítulo se analizan las diferentes técnicas de estimación, técnicas de validación así como la justificación de su selección. Además se aplican las técnicas seleccionadas y se determinan si es factible el desarrollo del módulo propuesto.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1. Introducción

El presente capítulo tiene como objetivo garantizar una correcta comprensión del objeto de estudio de la investigación, se mencionan algunos sistemas automatizados en los ámbitos internacionales y nacionales utilizados para la realización de la actividad minera. Se describen las tecnologías, técnicas así como la metodología de desarrollo seleccionada, el lenguaje de modelado y la herramienta CASE a utilizar en el desarrollo del problema propuesto.

1.1. Caracterización de los procesos involucrados en la aplicación del método de prospección y exploración, pozos de sondeos

La actividad minera se realiza en fases como son: la prospección, exploración, planificación, explotación y el cierre. Cada una de estas fases tiene sus impactos ambientales, y se debe prever las medidas ambientales correspondientes para prevenir, mitigar, remediar o compensar los mismos. La prospección y la exploración son fases estrechamente ligadas y a veces se combinan, y en general los geólogos son quienes la realizan. **(3)**

El proceso inicial en la búsqueda de un yacimiento mineral, consiste en realizar la prospección que no es más que la búsqueda de indicios de nuevas áreas mineralizadas y apunta a un reconocimiento general de una región. Luego de completar la etapa de prospección, si ésta ha sido prometedora, viene la etapa de exploración que se ejecuta con técnicas más avanzadas. En esta etapa pueden efectuarse estudios más avanzados de geoquímica, geofísica, y los sondeos diamantinos, o muestreos del terreno por medio de trincheras o canales, además en esta etapa se puede hallar la determinación del tamaño y forma del yacimiento, así como del contenido y calidad del mineral en él existente. La exploración incluye también la evaluación económica del yacimiento y se enfoca en una investigación encauzada de un área claramente definida y más reducida con respecto al área cubierta en la prospección. Por lo que un objetivo común de la prospección y de la exploración es la reducción del área de investigación, además de aumentar las ventajas del área prometedora con respecto a su explotación rentable, como por ejemplo ocuparse de un camino de acceso transitable y de un peritaje del medio ambiente.

Lo más importante en esta etapa es la delimitación de zonas en las cuales los suelos presentan características similares para eso se debe saber donde se encuentra ubicado el mineral a explotar por lo que sigue una serie de pasos, primeramente localizar el depósito mineral tomando fotos aéreas e imágenes a través de satélite es decir se hace la topografía del área, se toman las dimensiones del área de interés, luego se procede a tomar distintos tipos de muestras como del agua de río, del suelo, de plantas o de rocas y se aplica en ellas los análisis del laboratorio correspondiente y se obtiene la información de los minerales del subsuelo entre las que se encuentra la cantidad, la calidad y la distribución de los elementos.

Para tomar estos distintos tipos de muestras se utilizan métodos para la extracción de dicha muestra, entre los que se encuentran el método de pozos a cielo abierto ó calicatas y el método de sondeos. Estos sondeos se pueden realizar de tres formas diferentes mediante el método de exploración preliminar, método de sondeos definitivo o mediante métodos geofísicos. 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 de la orientación y localización de los sondajes, como también las propiedades de las muestras en cuatros ficheros: **(4)**

- ✓ collars.txt(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.
- ✓ surveys.txt (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.
- ✓ assays.txt (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.
- ✓ geology.txt (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.

1.2. Análisis de soluciones existentes

1.2.1. Sistemas internacionales

Diversas alternativas ofrece la industria de los software mineros disponibles en el mercado internacional, desde soluciones orientadas a aplicaciones específicas hasta productos que ofrecen servicios más integrales, que abarcan y controlan diversas áreas del negocio minero. En la actualidad, existen sistemas usados por las más importantes organizaciones mineras del mundo que logran mejores y mayores resultados en el modelado geológico, las actividades de planificación minera y la programación completa de la producción, como la colaboración entre diversas soluciones, entre los que se encuentran el Gemcom, Datamine, Surpac.

Datamine

Datamine es el líder mundial en software integrado para la industria de los recursos naturales, y su principal producto es Datamine Studio 3.0. Los usos más comunes del sistema son: la captura y análisis de la información, exploración, geología, geoquímica, mecánica de rocas, topografía, modelado geológico, diseño de mina a cielo abierto y subterráneo, planeamiento minero, y áreas relacionadas a los estudios ambientales. El rango de productos Datamine le otorga gran flexibilidad al sistema, con un sistema de archivos que conforman una base de datos abierta y relacional.

Datamine brinda varias opciones que posibilitan un mejor procesamiento de la información como son: un buscador de archivos lo cual ahorra espacio en el disco duro y mejora la velocidad de acceso y procesamiento de los datos, por tener los archivos o tablas de la base de datos almacenados en un formato binario. También posibilita la importación de cuatro archivos que contienen la información de la orientación y localización de los sondajes, como también las propiedades de las muestras constituyentes. **(5)**

Gemcom

Las soluciones integradas que desarrolla Gemcom abarcan desde las fases de exploración, evaluación de recursos, diseño de minado, optimización, planeamiento minero y control de leyes de producción, hasta la reconciliación y balance metalúrgico a lo largo de la línea de producción.

Gemcom es un programa que permite reducir la duplicación de datos mediante su fácil conexión con bases de datos corporativas y con los formatos de almacenamiento de datos y archivos usuales. Además de que da la posibilidad de elegir el tipo de base datos entre las que se encuentran: Jet 4 (Access 2000), SQL Server 7.0, SQL. Otras de las ventajas que posee Gemcom es que permite la configuración de las bases de datos y que en su versión Gems 5.1 la gran mayoría de la información se guarda en una base de datos, ahora incluyendo polilíneas, puntos, sólidos y superficies agregándose a las ya existentes como las base de datos de sondajes y polígonos. **(5)**

Surpac

Surpac tiene aplicación en el área geológica minera, específicamente en: exploración geológica, estimación de reservas, diseño en minería subterránea, planificación minera, modelado de relieve, levantamiento topográfico. Además de tener módulos para programar la producción de la mina, se puede programar el desarrollo de la mina, y mezclar producciones buscando una ley objetivo, también está revolucionando la industria del software de secuenciamiento con su capacidad de leer y de escribir datos en los formatos nativos así como de los otros sistemas haciendo más fácil su utilización para programar los datos que se

han elaborado en cualquiera de los sistemas principales. Los resultados que pueden ser obtenidos tienen una interface directa con MS Project, MS Access, y MS Excel, lo cual permite un fácil manejo y manipulación de la información. **(5)**

Estos software tienen mucho prestigio debido a su utilidad y seguridad de los datos, cada uno tiene sus ventajas y desventajas pero la realidad es que su principal inconveniente es que son privativos. Lo cual hace difícil el uso de los mismos en nuestro país por tener un alto costo en la obtención de su licencia y no poder hacer cambios en su código fuente.

1.2.2. Sistemas nacionales

En nuestro país se han desarrollado software para la actividad minera como el NIKEL, debido al uso intensivo del mismo se logró desarrollar una versión más integral para la prospección geológica de níquel llamado MICRONIQ. El cual puso al día las reservas de los yacimientos de níquel y sigue empleándose su algoritmo de modelado de recursos en versión más portable conocida como CORTE para los resultados de los trabajos de prospección, además de que los recursos y reservas certificados de los yacimientos cubanos están calculados con esta tecnología. Estos software y el sistema para SO LEAL para CID 201B, son los primeros software para cálculo de recursos que se confeccionaron con esfuerzo nacional.

En la actualidad estos sistemas mineros programados en el país (MICRONIQ, CORTE) no cumplen con las características requeridas para la actividad geológica minera en Cuba. Ninguno cuenta 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 directos e indirectos de prospección y exploración, pozos de sondeos. Además no posibilitan la determinación del tamaño y forma del yacimiento, así como del contenido y calidad del mineral existente, ni posibilitan la búsqueda de indicios de nuevas áreas mineralizadas. Estos sistemas informáticos cubanos no cuentan con una documentación para el uso de los mismos, lo que imposibilita 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.

1.3. Metodologías de desarrollo de software

“Los estándares o metodologías definen un conjunto de criterios de desarrollo que guían la forma en que se aplica la ingeniería del software” **(6)** en palabras sencillas es la base para la edificación de un proyecto de software, la etapa fundamental para lograr los objetivos buscados con dicho proyecto.

Piattini definió como metodología al “... conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a desarrollar un nuevo software”. **(7)**

Obtener un software con la calidad requerida implica la utilización de una metodología o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, en aras de lograr una mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que eleven la productividad. Su objetivo es controlar de manera transparente todo el proceso de desarrollo, fundamentalmente permite producir lo esperado en el tiempo esperado y con el costo esperado, esto es básico, pues muchas veces se planea hacer algo que finalmente toma más tiempo de lo planeado, por tanto produce más costos por lo que no se logra completar el proyecto según lo esperado.

Ya en estos tiempos se cuenta con varias metodologías, como por ejemplo se habla de metodologías tradicionales o robustas como el Proceso Unificado de Modelado (RUP), Microsoft Solution Framework (MSF) y metodologías ágiles donde se encuentra Programación Extrema (XP).

1.3.1. Proceso Unificado de Modelado (RUP)

Dentro de las metodologías fuertes se puede encontrar RUP. La metodología RUP es un:” proceso para el desarrollo de un proyecto de un software que define claramente quien, cómo, cuándo y qué debe hacerse en el proyecto. Presenta tres características esenciales: está dirigido por los casos de uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere; está centrado en la arquitectura: que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden; y es iterativo e incremental: donde divide el proyecto en mini-proyectos, donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada”. **(8)**

RUP divide en 4 fases el desarrollo del software:

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

Los elementos del RUP son:

- ✓ Actividades: Son los procesos que se llegan a determinar en cada iteración.
- ✓ Trabajadores: Personas o entes involucrados en cada proceso.
- ✓ Artefactos: Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Los flujos de trabajo con los que cuenta RUP son:

- ✓ Modelado de negocios
- ✓ Requisitos
- ✓ Análisis y Diseño
- ✓ Implementación
- ✓ Pruebas
- ✓ Distribución

y en 3 flujos de trabajo llamados de apoyo

- ✓ Administración Cambios y Configuración
- ✓ Administración del Proyecto
- ✓ Ambiente

Para guiar el proceso de desarrollo del software la arquitectura del proyecto seleccionó como metodología de desarrollo a utilizar RUP debido a que es una metodología robusta, sirve de guía a los equipos de proyecto en cómo administrar el desarrollo iterativo de un modo, además de describir los diversos pasos involucrados en la captura de los requerimientos y en el establecimiento de una guía arquitectónica de acuerdo a los requerimientos y a la arquitectura. Proporciona una definición de proceso común y central que todos los miembros del equipo de desarrollo pueden compartir, garantizando siempre una comunicación clara y sin ambigüedades entre los miembros del equipo.

Es una metodología que se adapta a las necesidades del proyecto por lo que no se generarán todos los artefactos que propone dicha metodología, permite generar una documentación para luego continuar el proceso de desarrollo de software sin ningún tipo de problema ya que la duración del proyecto es larga y el capital humano con que cuenta cambiará frecuentemente. También RUP provee un enfoque de ingeniería de software para una mejor asignación de tareas así como las responsabilidades dentro del equipo de desarrollo en que se encuentra el proyecto. Una de las ventajas que ofrece RUP es que aumenta la calidad del equipo de desarrollo ya que permite que no importa cuál sea la responsabilidad de cada individuo en el equipo de desarrollo este podrá entrar a la misma base de datos de conocimientos. Esto proporciona que cada cual pueda compartir el mismo lenguaje, la misma visión, y que se centren en el mismo proceso acerca de cómo desarrollar el software. Estas características fueron las que se tuvieron en cuenta para la selección de dicha metodología y así obtener los artefactos correspondientes.

1.4. Lenguaje de Modelado Unificado o UML

“Lenguaje Unificado de Modelado o (UML) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema”. **(9)** Su utilización es independiente del lenguaje de programación y de las características de los proyectos, ya que UML ha sido diseñado para modelar cualquier tipo de proyecto tanto informáticos como de arquitectura, o de cualquier otra rama y que posibilita la modificación de todos sus miembros mediante estereotipos y restricciones. Este es el lenguaje de modelado escogido para elaborar los distintos diagramas por ser el lenguaje que utiliza la metodología seleccionada; a continuación se describe de forma breve sus características principales.

El uso del lenguaje de modelado UML permite:

- ✓ Especificar todas las decisiones de análisis, diseño e implementación, construyéndose así modelos no ambiguos, precisos y completos para darle solución al problema planteado.
- ✓ Documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones).
- ✓ Permite expresar de forma gráfica un sistema de forma que otro lo pueda entender.
- ✓ Permite especificar cuáles son las características de un sistema antes de su construcción.

1.5. Herramienta CASE

En las últimas décadas se ha trabajado en el área de desarrollo de sistemas para encontrar técnicas que permitan incrementar la productividad y el control de calidad en cualquier proceso de elaboración de software, y hoy en día la tecnología de Ingeniería de Software Asistida por Ordenador o Computer Aided Software Engineering por sus siglas en ingles CASE reemplaza al papel y al lápiz por el ordenador, para transformar la actividad de desarrollar software en un proceso automatizado.

“Las herramientas CASE se pueden definir como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software” **(10)**. Estas herramientas nos pueden ayudar en todos los aspectos como el proceso de realizar un diseño del proyecto, cálculo de costos, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

Una de estas herramientas CASE es el Visual Paradigm, la cual soporta el ciclo de vida completo del desarrollo de software y es una de las herramientas más poderosas y potente que cuenta con mucha popularidad y beneficios. A continuación se describen las principales características:

1.5.1. Visual Paradigm

Visual Paradigm es una herramienta CASE que utiliza UML como lenguaje de modelado. “Está diseñada para una amplia gama de usuarios interesados en construir sistemas de software fiables con el uso del paradigma orientado a objetos, incluyendo actividades como ingeniería de software, análisis de sistemas y análisis de negocios”. **(11)**

Esta herramienta ayuda a los equipos de desarrollo de software para sobresalir todo el modelo de acumulación de trabajo y así desplegar el proceso de desarrollo de software, lo que permite maximizar y acelerar tanto las contribuciones individuales como las de equipo. También permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Esta herramienta permite tanto la ingeniería directa como inversa soportada en varios lenguajes de programación, brinda un diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.

La disponibilidad de una licencia gratuita y comercial y de múltiples versiones para cada necesidad y la posibilidad de permitir representar todo tipo de diagramas UML para las distintas fases por la que transita un software en desarrollo como la captura de requisitos, análisis, diseño e implementación fueron los aspectos claves para la selección de Visual Paradigm como la herramienta idónea para el modelado del sistema propuesto, además de que es una excelente herramienta para ser utilizada en un ambiente de software libre. Tiene como beneficios los siguientes:

- ✓ Presenta una licencia gratuita.
- ✓ Fácil de utilizar, instalar y actualizar.
- ✓ Proporciona una gran cantidad de tutoriales y demostraciones interactivas de UML, que constituyen una guía de referencia y consulta para sus usuarios.
- ✓ Disponible en español.

1.6. Técnica de estimación

Realizar la estimación de un proyecto software no es tarea fácil, especialmente cuando no existe un método único viable para cada tipo de proyecto. La planificación, en el mundo de la Industria del Software, se ha convertido en uno de los principales retos para la gestión de proyectos y una actividad fundamental para desarrollar software de alta calidad. Se considera imprescindible para planificar, la aplicación de

buenos métodos de estimación, por la creciente influencia que ejercen en el control preciso, predecible y repetido sobre los procesos de producción y los productos de software. Se investigaron 3 métodos de estimación y se seleccionó uno de ellos. A continuación se describe en qué consiste cada uno de ellos.

El método de Puntos de Función: fue creado como un método de estimación por Allan Albrecht y se basa principalmente en la identificación de los componentes del sistema informático en términos de transacciones y grupos de datos lógicos que son relevantes para el usuario en su negocio. A cada uno de estos componentes les asigna un número de puntos por función basándose en el tipo de componente y su complejidad; y la sumatoria de esto, da los puntos de función sin ajustar. El ajuste es un paso final basándose en las características generales de todo el sistema informático que se está contando. **(12)**

COCOMO: fue propuesto y desarrollado por Barry Boehm en 1981, es uno de los modelos de estimación de costo mejor documentado, estudiado y utilizado en la industria de software. El modelo permite, basándose en un grupo de ecuaciones no lineales obtenidas mediante técnicas de regresión a través de un histórico de proyectos ya realizados; estimar el esfuerzo, costo y tiempo que se requiere en un proyecto de software a partir de una medida del tamaño del mismo, expresada en el número de líneas de código que se estimen generar para la creación del producto software. El modelo original ha evolucionado a un modelo más completo llamado COCOMO III. **(12)**

La estimación mediante el análisis de Puntos de Casos de Uso: es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto de asignación de “pesos” a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores. **(13)**

El método de estimación seleccionado para la siguiente investigación es Análisis de Punto de Caso de Uso por ser un método de estimación y cálculo de tamaño del software basado en cuentas hechas sobre los casos de uso para un sistema de software, es uno de los métodos más efectivos para capturar la funcionalidad de un sistema. Además este método cumple con al menos dos objetivos fundamentales:

- ✓ Permite disponer de una métrica dimensional que con el tiempo podrá ser comparada con las métricas actualmente en uso y se enriquecerán con la historia de los proyectos que se desarrollen bajo la metodología de orientación a objetos.
- ✓ Existencia de ejemplos de uso del método de distintas fuentes.

Este método posee ventajas con respecto a los demás como por ejemplo el método COCOMO no fue seleccionado por tener el inconveniente de que hay que contar con una cantidad de líneas de código fuente a implementar por casos de uso y para lograr este indicador se debe tener experiencia en el desarrollo de aplicaciones. Al igual que el método de Punto de Función no se seleccionó por ser difícil de realizar si no se cuenta con una base histórica de proyectos que provea los coeficientes de conversión.

1.7. Conclusiones Parciales

El análisis y la investigación realizada en este capítulo, ha mostrado de manera clara y se pudo comprender de una forma detallada la necesidad de la existencia de un software que requiera como etapa inicial el estudio de los procesos involucrados en la captura y procesamiento de la información obtenida durante la actividad minera al aplicar los métodos directos e indirectos de prospección y exploración, pozos de sondeos. Además, se realizó un estudio sobre los principales sistemas existentes que se encuentran relacionados con la actividad minera, lo que permitió tener una idea más clara sobre el funcionamiento de los mismos y lograr tener una base para el desarrollo del que se pretende hacer, permitió escoger la metodología de desarrollo que mejor se adecua a la necesidad, así como las herramientas utilizadas para un mejor desarrollo favoreciendo así que se realice con la calidad requerida a favor de los resultados esperados. Todas estas tecnologías fueron seleccionadas teniendo en cuenta los principios de independencia tecnológica que ha venido defendiendo desde sus inicios la UCI.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA PROPUESTO

1. Introducción

En el presente capítulo se describen las características del sistema a construir. Debido a la poca definición de los procesos de negocio se hace necesaria la definición de conceptos y sus relaciones, agrupados en un modelo de dominio. Este sirve de apoyo para la especificación de las condiciones, capacidades y cualidades que el sistema debe tener; las que se especifican en términos de requerimientos y casos de uso, mostrándose estos últimos en un diagrama de casos de uso en conjunto con los actores que interactúan con los mismos.

2.1. Modelo de Dominio

Un modelo de dominio (o modelo conceptual) muestra gráficamente los conceptos (clases de objetos), los atributos y las asociaciones más importantes del dominio del problema. **(14)**

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.

Representa un modelo de conceptos de objetos del dominio, no del software y estos conceptos son importantes a la hora de entender lo que se quiere representar, es decir, un diccionario visual que utiliza la notación UML de diagrama de estructura estática.

Incluye:

- ✓ Clases de objetos.
- ✓ Asociaciones entre clases de objetos.
- ✓ Atributos de las clases de objetos.

En el caso del módulo que permite la gestión de la información geológica para el software minero Geolmin se define un modelo conceptual al no haber información suficiente para establecer procesos de negocio, por lo que se definieron clases conceptuales relativas al dominio del problema. Esto proporcionará a los usuarios, clientes, desarrolladores y otros interesados, la ventaja de utilizar un vocabulario claro y relacionado con el sistema en cuestión y facilitará la captura de los requerimientos para darle solución al problema como se presenta a continuación en la figura 1.

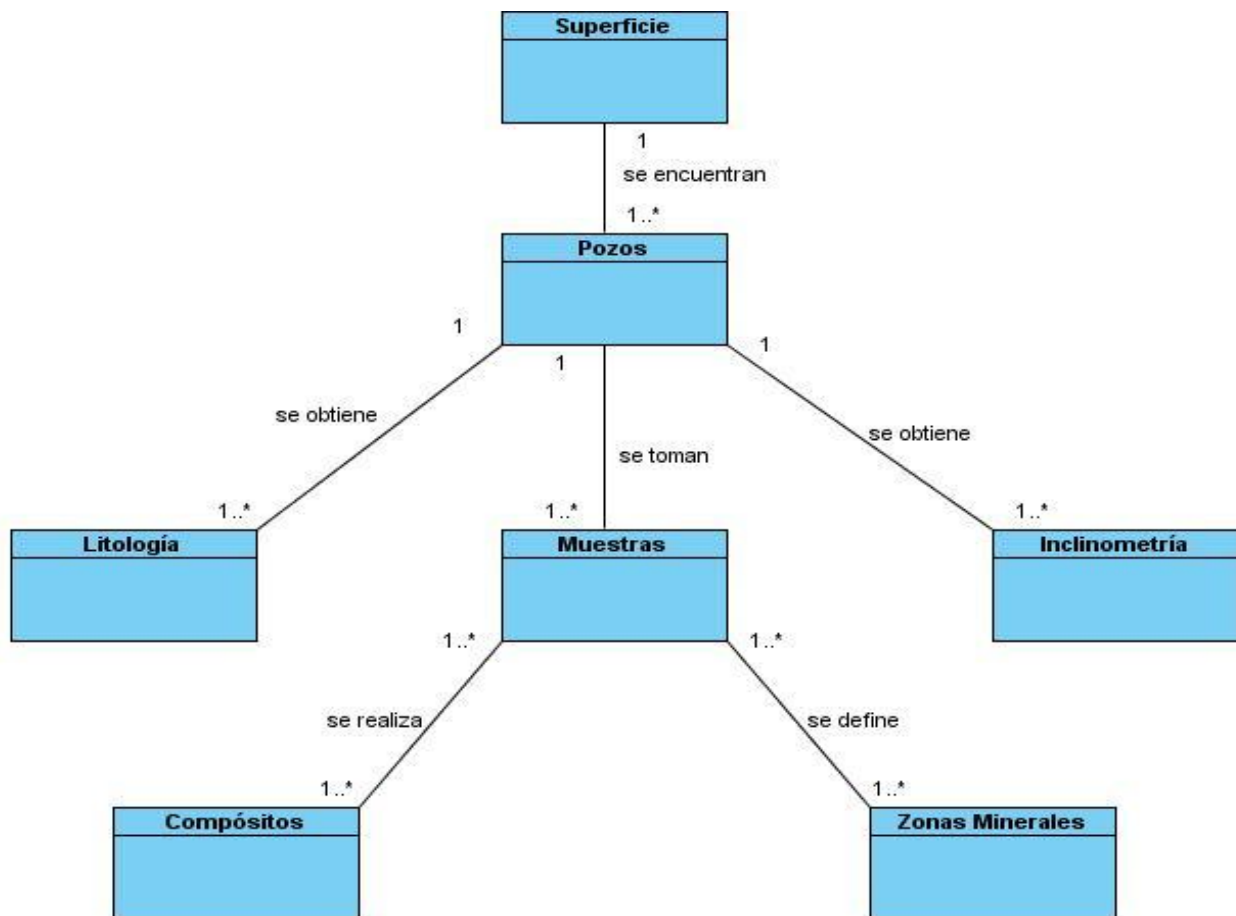


Figura 1: Modelo de dominio o modelo conceptual

Descripción de las clases del modelo de dominio

- ✓ **Inclinometría:** Densidad del mineral existente en los diferentes pozos de sondeos.
- ✓ **Zona Mineral:** Zona específica donde se encuentra el mineral.
- ✓ **Superficie:** Zona donde se pretende hacer el pozo de sondeo.
- ✓ **Compósito:** Acción de regularizar las muestras, es decir que exista el mismo intervalo entre las muestras tomadas.
- ✓ **Muestras:** Es una parte o una porción que se toma de cada mineral.
- ✓ **Litología:** Tipo de roca existente en los pozos de sondeos.
- ✓ **Pozos:** Orificio o túnel vertical perforado en la tierra, hasta una profundidad suficiente para alcanzar el mineral que se busca.

2.1.1. Descripción del diagrama

A través de una superficie que se define como la zona donde se pretende hacer los pozos de sondeo se encuentran varios pozos, a la hora de realizar los mismos se tiene que tomar distintos tipos de muestras, además de que se obtiene la Inclínometría que no es más que la densidad del mineral, también se obtiene la litología es decir el tipo de roca con se cuenta a la hora de realizar el pozo. Una vez tomadas las muestras se realizan los compósitos, entiéndase por compósitos la acción de regularizar las muestras, es decir que exista el mismo intervalo entre las muestras tomadas y luego se define la zona mineral.

2.2. Especificación de los requisitos de software

Los requerimientos del software son las condiciones necesarias para el desarrollo de un sistema. El objetivo primario es documentar todos los requerimientos del sistema para proporcionar una descripción completa y comprensiva de lo deseado por el cliente, con lo cual debe cumplir el software a desarrollar.

La captura de requisitos y el modelado del sistema son las actividades fundamentales que se desarrollan durante la fase de inicio del desarrollo de un software. Precisamente el flujo de trabajo de captura de requisitos se encarga de capturar los requisitos funcionales y no funcionales y modelar el sistema propuesto.

2.2.1. Requisitos funcionales

Los requerimientos funcionales del módulo que permite la gestión de la información geológica para el software minero Geolmin, especifican lo que dicho módulo debe hacer, para lo cual se definió el alcance del módulo teniendo en cuenta las acciones a realizar y el intercambio de datos entre sus diferentes funciones. A continuación se listan los requerimientos funcionales identificados para el módulo:

RF1. Configurar parámetros de conexión del Sistema Gestor de Base de Datos SQLite.

RF2. Configurar parámetros de conexión del Sistema Gestor de Base de Datos PostgreSQL.

RF3. Validar la información de los pozos contenida en ficheros de texto.

RF4. Modificar la información de las muestras almacenada en la base de datos.

RF5. Modificar la información de la litología almacenada en la base de datos.

RF6. Modificar la información de la Inclínometría almacenada en la base de datos.

RF7. Modificar la información de las coordenadas almacenada en la base de datos.

RF8. Mostrar la información de las muestras almacenada en la base de datos.

- RF9. Mostrar la información de la litología almacenada en la base de datos.
- RF10. Mostrar la información de la Inclínometría almacenada en la base de datos.
- RF11. Mostrar la información de las coordenadas almacenada en la base de datos.
- RF12. Exportar archivos.
- RF13. Importar archivos.
- RF14. Insertar fila a una tabla de la base de datos.
- RF15. Eliminar fila a una tabla de la base de datos.
- RF16. Insertar columna a una tabla de la base de datos.
- RF17. Eliminar columna a una tabla de la base de datos.

2.2.2. Especificación de requisitos

Configurar parámetros de conexión del Sistema Gestor de Base de Datos (SGBD) SQLite

En el módulo se contará con la opción de identificar el tipo de SGBD que es lo que permite hacer la conexión con la base de datos, en el caso de seleccionar como SGBD SQLite permite llenar datos como nombre y lugar físico.

Configurar parámetros de conexión del Sistema Gestor de Base de Datos (SGBD) PostgreSQL

En el módulo se contará con la opción de identificar el tipo de SGBD que es lo que permite hacer la conexión con la base de datos, en el caso de seleccionar como SGBD PostgreSQL permite llenar datos como servidor, puerto, usuario, contraseña entre otros.

Validar la información de los pozos contenida en ficheros de texto.

En el módulo se podrá realizar la operación de validar la información referente a los pozos contenida en los cuatros ficheros. Con esto se tendría una mayor seguridad de la información contenida en cada fichero.

Modificar la información de las muestras almacenada en la base de datos.

El módulo debe permitir realizar la operación modificar la información contenida en el fichero muestras donde se modificarán los cambios que el usuario desee hacer a la información registrada de cada muestra.

Modificar la información de la litología almacenada en la base de datos.

El módulo debe permitir realizar la operación modificar la información contenida en el fichero litología donde se modificarán los cambios que el usuario desee hacer a la información referente a la litología.

Modificar la información de la Inclínometría almacenada en la base de datos.

El módulo debe permitir realizar la operación modificar la información contenida en el fichero Inclínometría donde se modificarán los cambios que el usuario desee hacer a la información referente a la Inclínometría.

Modificar la información de las coordenadas almacenada en la base de datos.

El módulo debe permitir realizar la operación modificar la información contenida en el fichero coordenada donde se modificarán los cambios que el usuario desee hacer a la información registrada de cada coordenada.

Mostrar la información de las muestras almacenada en la base de datos.

En el módulo se realizará la operación de mostrar la información de las muestras permitiendo al usuario seleccionar la información que desea que se le muestre, así puede satisfacer sus requerimientos y necesidades.

Mostrar la información de la litología almacenada en la base de datos.

En el módulo se realizará la operación de mostrar la información de la litología permitiendo al usuario seleccionar la información que desea que se le muestre, así puede satisfacer sus requerimientos y necesidades.

Mostrar la información de la Inclínometría almacenada en la base de datos.

En el módulo se realizará la operación de mostrar la información de la Inclínometría permitiendo al usuario seleccionar la información que desea que se le muestre, así puede satisfacer sus requerimientos y necesidades.

Mostrar la información de las coordenadas almacenada en la base de datos.

En el módulo se realizará la operación de mostrar la información de las coordenadas permitiendo al usuario seleccionar la información que desea que se le muestre, así puede satisfacer sus requerimientos y necesidades.

Exportar archivos.

El módulo permitirá al usuario exportar archivos en formato de texto limitado o en XML con esto se puede exportar cualquier información necesaria.

Importar archivos.

El módulo permitirá al usuario importar archivos en formato de texto limitado o en XML que no es más que importar un archivo para un posterior trabajo con el mismo.

Insertar fila a una tabla de la base de datos.

El módulo debe contar con la opción de insertar una fila a una tabla de la base de datos para poder contar con más información.

Eliminar fila a una tabla de la base de datos.

El módulo debe contar con la opción de eliminar una fila a una tabla de la base de datos y así poder eliminar la información no deseada.

Insertar columna a una tabla de la base de datos.

El módulo ofrece la opción de insertar una columna a una tabla de la base de datos lográndose crear un campo mas de la tabla.

Eliminar columna a una tabla de la base de datos.

El módulo ofrece la opción de eliminar una columna a una tabla de la base de datos lográndose eliminar la información que desee.

2.2.3. Requisitos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener **(15)**. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable, además de que los requerimientos no funcionales forman una parte significativa de la

especificación y normalmente están vinculados a requerimientos funcionales. A continuación se listan los requerimientos no funcionales que el sistema debe tener:

Requerimientos de usabilidad

- ✓ El módulo debe poseer una interfaz amigable, que contenga los colores característicos de la minería.
- ✓ El módulo debe poder ser usado por cualquier persona que tenga conocimientos básicos de sistemas mineros.
- ✓ El módulo debe proporcionar un acceso fácil y rápido al usuario, con el objetivo de proveer familiarización con el uso del mismo.

Requerimientos de apariencia o interfaz externa

- ✓ El módulo debe tener una interfaz sencilla, amigable, intuitiva y de fácil navegación por el usuario.
- ✓ El diseño de la interfaz del sistema responderá a la ejecución de acciones de una manera rápida, minimizando los pasos a dar en cada proceso.
- ✓ El módulo debe tener una apariencia profesional y un diseño gráfico sencillo, composición homogénea de color, estilos y fuentes en toda la interfaz.

Requerimientos de portabilidad y operatividad

- ✓ La aplicación debe ser compatible con los Sistemas Operativos: Windows XP y GNU/Linux.

Restricciones en el diseño y la implementación

- ✓ Para el desarrollo del análisis y el diseño del módulo deberá ser utilizada la metodología RUP, usando el lenguaje de modelado UML y como herramienta para llevarlo a cabo el Visual Paradigm en su versión 6.4.
- ✓ Para la implementación se utilizará como lenguaje de programación C++.

Requerimientos de Soporte

- ✓ El Sistema deberá tener:
 - Fácil instalación, configuración y puesta en marcha.

Requerimientos de software

- ✓ Las computadoras que utilizará el software deben tener instalado:
 - Windows XP Profesional o GNU/Linux en cualquier distribución.

Requerimientos de hardware

- ✓ Para garantizar el correcto funcionamiento del módulo, se necesita una computadora con un procesador Pentium IV o superior y una memoria RAM de 512 MB como mínimo.

2.3. Descripción de la integración del módulo con el software Geolmin

El desarrollo del software geológico minero cubano es una idea propuesta por distintas entidades de esta rama. El proyecto Minería es un proyecto de colaboración entre la UCI y entidades geólogo-mineras pertenecientes al MINBAS. Este sistema enmarcado como una solución informática, denominado Geolmin, es desarrollado en el Centro de Desarrollo de Software GEySED de la universidad, y debe permitir de manera general modelar geológicamente un depósito mineral, estimar los recursos y reservas de un depósito mineral y optimizar, diseñar y planificar la actividad minera. Este sistema está compuesto por varios módulos entre los que se encuentran: módulo base o núcleo de la aplicación, módulo que se encarga de la visualización gráfica (3D) de la información del sistema, módulo que permite la creación y manipulación de modelos de bloques, módulo que permitirá realizar análisis estadísticos y geoestadísticos de la información y el módulo que permite la gestión de la información geológica en la base de datos del sistema.

Este último módulo es una parte importante en el desarrollo del software minero que se desea realizar. Tiene como propósito principal gestionar toda la información de la base de datos obtenida a partir de los sondeos realizados a los pozos de perforación, además de facilitar la administración de los datos que se necesita manejar. También presenta una capa de abstracción de base de datos que permite la interacción con distintos Sistemas Gestores de Base de Datos (SGBD) y usa por defecto el SGBD SQLite, por lo que

se hace necesario la integración de dicho módulo al sistema para que este pueda brindar soluciones que sean las más acertadas, y de esta forma hacer más eficientes el trabajo con la información obtenida en la aplicación del método de prospección y exploración, pozos de sondeos.

2.4. Definición de los casos de uso del sistema

Los casos de usos del sistema representan una unidad funcional coherente del sistema, subsistema o clases y son artefactos narrativos que describen el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades que debe cumplir el sistema.

2.4.1. Descripción de los actores del sistema

Los actores del sistema “pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado”. **(15)** Es decir intercambian información con el sistema pero no son parte de él. A continuación se muestra la descripción de los actores identificados, los cuales tienen una interacción directa con el sistema a implementar.

Tabla 1: Descripción de los actores del sistema

Actor	Descripción
Cliente	Interactúa con el sistema con el objetivo de brindarle al sistema los datos necesarios para que este pueda mostrar, validar y modificar la información, así como exportar dicha información.

2.4.2. Diagrama de caso de uso del sistema

El modelo de caso de uso permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requerimientos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. El mismo contiene actores, casos de uso y sus relaciones y constituye una entrada para el análisis, el diseño y las pruebas.

La representación gráfica de las interacciones entre casos de uso y actores del sistema se define como “Diagrama de Casos de Uso del Sistema”, a continuación se muestra en la figura 2 el diagrama modelado para el módulo que se propone implementar.

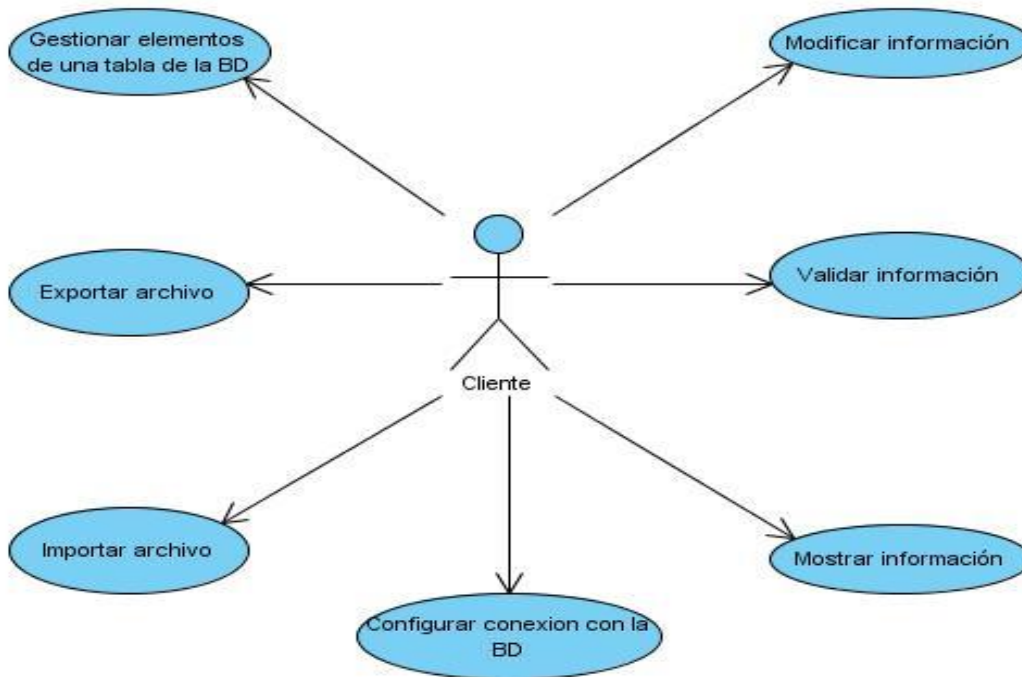


Figura 2: Diagrama de Caso de Uso del Sistema

2.4.3. Descripción de los casos de usos 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 presenta la descripción textual del caso de uso configurar conexión con la base de datos las demás se pueden encontrar en [Anexo # 1](#)

Descripción del caso de uso: Configurar conexión con la base de datos.

Tabla 2: Descripción textual del caso de uso: Configurar conexión con la base de datos

Caso de Uso:	Configurar conexión con la base de datos.
Actores:	Cliente.
Resumen:	El caso de uso comienza cuando el usuario selecciona la opción Configurar

	conexión con la base de datos, dentro del mismo selecciona proveedor PostgreSQL o proveedor SQLite y finaliza cuando el sistema estable la conexión con la base de datos.
Precondiciones:	Se tiene que haber creado el proyecto.
Referencias	RF1, RF2.
Prioridad	Crítico.

Flujo Normal de Eventos

1. El caso de uso comienza cuando el usuario selecciona en el menú base de datos la opción de configurar conexión con la base de datos.	2. El sistema muestra la interfaz para seleccionar el proveedor de la base de datos y las opciones de cancelar y aceptar, la opción de aceptar estará bloqueada hasta que no se realice las opciones de configurar el proveedor de base de datos seleccionado.
3. El usuario selecciona el proveedor de base de datos. Si el usuario selecciona como proveedor PostgreSQL ver Sección “Proveedor PostgreSQL”. Si el usuario selecciona como proveedor SQLite ver Sección “Proveedor SQLite”.	

Sección “Proveedor PostgreSQL”

Acción del Actor	Respuesta del Sistema
	1.El sistema muestra los campos de configuración : <ul style="list-style-type: none"> • Servidor: • Puerto: • Usuario:

	<ul style="list-style-type: none"> • Contraseña: • Base de Datos: • La opción para escoger el protocolo seguro de comunicación de la red SSL: • Puerto SSL: • La opción para establecer la prueba de conexión:
2. El usuario llena los campos correspondiente.	3. El sistema valida a medida que el usuario llena los campos que los datos introducidos sean correctos.
4. El usuario selecciona la opción "prueba de conexión".	5. El sistema verifica que los parámetros introducidos por el usuario sean correctos y emite un mensaje "Conexión Satisfactoria" y desbloquea la opción aceptar.
6. El usuario da clic en la opción aceptar.	7. El caso de uso termina cuando el sistema establece la conexión con la base de datos.
Prototipo de interfaz	

Flujos Alternos

	3. Si los datos introducidos son incorrectos el sistema muestra un mensaje “Los datos son incorrectos”.
	5. Si los parámetros introducidos no son correctos el sistema muestra un mensaje “Conexión no satisfactoria”.
6. Si el usuario no da clic en aceptar y da cancelar.	7. El sistema cierra la interfaz.

Sección “Proveedor SQLite”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	<p>1.El sistema muestra los campos de configuración:</p> <ul style="list-style-type: none"> • Nombre: • Lugar específico:
<p>2. El usuario llena los campos correspondientes.</p>	<p>3. El sistema verifica que los parámetros introducidos por el usuario sean correctos y desbloquea la opción aceptar.</p>
<p>4. El usuario da clip en la opción “aceptar”.</p>	<p>5. El caso de uso termina cuando el sistema establece la conexión con la base de datos.</p>

Prototipo de interfaz



Flujos Alternos

	<p>3. Si los parámetros introducidos no son correctos el sistema muestra un mensaje “Conexión no satisfactoria”.</p>
<p>4. Si el usuario no da clic en aceptar y da cancelar.</p>	<p>5. El sistema cierra la interfaz.</p>

Poscondiciones	Se configura la conexión con la base de datos.
-----------------------	--

2.5. Análisis

El análisis dice lo que hace el sistema, comprende el análisis de requerimientos donde se genera una fluida comunicación con el usuario, se revelan las necesidades del sistema, se definen sus objetivos, su alcance y se establecen todas sus funciones a contemplar.

El análisis consiste en obtener una visión inicial de cómo la implementación debe ser, por lo que es una guía indiscutible para los programadores. En él se definen las clases que estarán asociadas con el contexto del dominio del problema por lo que representan conceptos y relaciones. **(16)**

Clases Interfaz

Las clases interfaz representan la interacción entre el sistema y sus actores un ejemplo de esto es la clase interfaz validar, modificar, mostrar y exportar información lo cual fueron las definidas para este módulo.

Clases entidad

Estas clases modelan información referente a fenómenos, conceptos y sucesos que ocurren en la vida real, los cuales poseen larga vida y a menudo son persistentes, definiéndose como clases entidad las siguientes: conexión, validar, modificar, mostrar, elementos, archivo, importar.

Clases de control

Las clases de control coordinan el trabajo de uno o unos pocos casos de uso, coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso, esto sirvió de ejemplo para definir como clases de control para dicho módulo las siguientes: gestionar información, base de datos.

2.5.1. Modelo de análisis

Es usado para representar la estructura global del sistema, describe la realización de casos de uso, sirve como una abstracción del modelo de diseño y se centra en los requerimientos funcionales.

Este modelo no es un diagrama final que describe todos los posibles conceptos y sus relaciones, es un primer intento por definir los conceptos claves que describen el sistema. Este artefacto es opcional, pero también tiene a su vez la propiedad de ser temporal en el caso en que se planea su desarrollo. Su utilidad radica en que permite una apreciación global conceptual del sistema. **(16)**

Para realizar el análisis se definieron cuáles iban a ser las clases interfaz, controladora y las entidades, al ser este modelo interno del negocio por describir la realización de los casos de uso, es decir, la interacción del cliente con las diferentes entidades del sistema. A continuación se presenta en la figura 3 el siguiente diagrama de clases del análisis del caso de uso configurar conexión con la base de datos los demás se pueden encontrar en el [Anexo # 2](#)

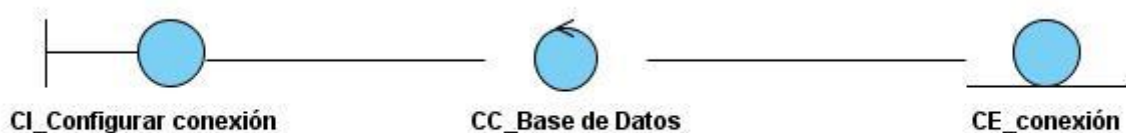


Figura 3: Diagrama de Clases del Análisis

2.6. Diagramas de colaboración

Los diagramas de colaboración son una forma de representar interacción entre objetos por lo que muestran cómo las instancias específicas de las clases trabajan juntas para conseguir un objetivo común, además implementan las asociaciones del diagrama de clases mediante el paso de mensajes de un objeto a otro. (17)

Para poder hacer la realización de los mismos se tiene que tener en cuenta la dependencia de otros artefactos como son:

- ✓ Los casos de uso (expandidos).
- ✓ Diagrama de secuencias.
- ✓ Diagrama de Clases.

Utilizándose este último como el artefacto principal para hacer la realización de los mismos. A continuación se presenta en la figura 4 el diagrama de colaboración: configurar conexión con la base de datos los demás se pueden encontrar en el [Anexo # 3](#)



Figura 4: Diagrama de Colaboración

2.7. Conclusiones Parciales

En este capítulo se desarrollaron los dos primeros flujos de trabajo propuestos por RUP como metodología de software utilizada para guiar el proceso de desarrollo. Primeramente se obtuvo un modelo conceptual por la poca descripción de los procesos del negocio logrando así obtener un entendimiento claro referente a todo el proceso del negocio. Se describen los casos de uso del sistema a través del establecimiento de los requerimientos funcionales y se definieron los requerimientos no funcionales para el éxito del sistema cuando esté terminado. Se obtiene una visión general del sistema es decir una propuesta de cómo pretende que quede el sistema a través del diagrama de clases del análisis.

La realización de estos flujos de trabajo permitió obtener una vista externa del sistema, que describe lo que el cliente espera obtener a través de casos de uso. Una vez cumplido este objetivo, es posible adentrarse al estudio de factibilidad y validación del módulo propuesto.

CAPÍTULO 3: ESTUDIO DE FACTIBILIDAD Y VALIDACIÓN DEL SISTEMA PROPUESTO

3. Introducción

En las últimas décadas han surgido numerosas propuestas que permiten hacer la validación a los sistemas que se construyen, ya que así se obtiene un mejor resultado en cuanto al resultado de los mismos. Por tal razón es que para el desarrollo de un proyecto de software y lograr el éxito del mismo, es crucial cumplir con el paso de definición de requisitos. Otro de los aspectos importantes a tratar será el estudio de la factibilidad del módulo que permite la gestión de la información geológica para el software minero Geolmin, ya que es de suma importancia por lo que se realiza con el objetivo de auxiliar a una organización a lograr sus metas y a su vez de cubrir sus fines con los recursos actuales. En el presente capítulo se plantean algunas alternativas posibles para la estimación del esfuerzo en proyectos.

3.1. Aplicación del método seleccionado para realizar la estimación

El método de puntos en casos de uso, es un método de estimación prometedor que se adapta bien al enfoque de caso de uso para la descripción de los requisitos. El cual fue el seleccionado para ser aplicado en esta investigación para realizar la estimación y está estructurado en 3 paso fundamentales, el primero es calcular Puntos de CU sin Ajustar, el segundo es obtener los Puntos de CU Ajustados y por último el Cálculo del Esfuerzo de Implementación – Estimación Final.

En sus bases yace el modelo de caso de uso, que consiste en actores y casos de uso. El número y el peso de los casos de uso identificados representan el componente más importante para el cálculo de los llamados puntos de caso de uso sin ajustar. El tamaño de un sistema se calcula a partir de los puntos de caso de uso sin ajustar, ajustándolos según el factor de complejidad técnico obtenido tras considerar las propiedades técnicas del sistema.

Una vez que se tiene una estimación del tamaño del sistema, se puede comenzar a pensar en el cálculo del esfuerzo. Es posible hacer esto calculando el factor ambiental (FE) a partir de las calificaciones del equipo y otras influencias ambientales. A continuación se explica detalladamente cada paso en la aplicación de este método. **(18)**

➤ Puntos de caso de uso sin ajustar (UUCP)

Este cálculo se lleva a cabo a través de la ecuación: $UUCP = UAW + UUCW$

Donde:

UUCP: Puntos de Casos de Uso sin Ajustar.

UAW: Factor de Peso de los Actores sin Ajustar.

UUCW: Factor de Peso de los Casos de Uso sin Ajustar.

➤ **Factor de peso de los actores sin ajustar (UAW)**

Este valor del **factor de peso de los actores sin ajustar (UAW)** se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema y en segundo lugar, la forma en la que el actor interactúa con el sistema. Calculándose de la siguiente manera:

(18)

Tabla 3: Factor de Peso de los Actores sin Ajustar

Tipo de Actor	Descripción	Factor de Peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0
Medio	Sistema con sistema mediante protocolo de interfaz basada en texto.	2	0	0
Complejo	Persona que interactúa con el sistema mediante interfaz gráfica.	3	1	3
	Total		1	3

Por tanto: $UAW = \Sigma \text{ cant actores} * \text{ peso}$

$$UAW = 1 * 3$$

$$UAW=3$$

➤ **Factor de Peso de los Casos de Uso sin ajustar (UUCW)**

El valor **factor de peso de los casos de uso sin ajustar (UUCW)** se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el

mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. Los criterios se muestran en la siguiente tabla: **(18)**

Tabla 4: Factor de Peso de los Casos de Uso sin Ajustar

Tipo de CU	Descripción	Peso	Cantidad de CU	Total
Simple	El Caso de Uso contiene de 1 a 3 transacciones	5	7	35
Medio	El Caso de Uso contiene de 4 a 7 transacciones	10	0	0
Complejo	El Caso de Uso contiene más de 8 transacciones	15	0	0
	Total		6	30

Por tanto: $UUCW = \sum \text{cant CU} * \text{Peso}$

$$UUCW = 6 * 5$$

$$UUCW = 35$$

Una vez calculado el valor del **factor de peso de los actores sin ajustar** y el **peso de los casos de usos sin ajustar** se puede calcular los **puntos de casos de uso sin ajustar**:

$$UUCP = UAW + UUCW$$

$$UUCP = 3 + 35$$

$$UUCP = 38$$

➤ **Puntos de caso de uso ajustados (UCP)**

Una vez que se tienen los puntos de casos de uso sin ajustar, se debe ajustar éste valor mediante la siguiente ecuación: **$UCP = UUCP * TCF * EF$**

Donde:

UCP: Puntos de Casos de Uso Ajustados.

UUCP: Puntos de Casos de Uso sin Ajustar.

TCF: Factor de Complejidad Técnica.

EF: Factor de Ambiente.

El **factor de complejidad técnica (TCF)** se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante). **(18)**

Tabla 5: Factor de Complejidad Técnica.

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido.	2	2	4
T2	Objetivo de performance o tiempo de respuesta.	1	4	4
T3	Eficiencia del usuario final.	1	4	4
T4	Procesamiento interno complejo.	1	3	3
T5	El código debe ser reutilizable.	1	4	4
T6	Facilidad de instalación.	0.5	5	2,5
T7	Facilidad de uso.	0.5	5	2,5
T8	Portabilidad.	2	5	10
T9	Facilidad de cambio.	1	3	3
T10	Concurrencia.	1	2	2
T11	Incluye objetivos especiales de seguridad.	1	4	4
T12	Provee acceso directo a terceras partes.	1	5	5
T13	Se requieren facilidades especiales de entrenamiento a usuarios.	1	1	1
	Total		47	49

Significado de los valores:

- 0: No presente o sin influencia.
- 1: Influencia o presencia incidental.
- 2: Influencia o presencia moderada.
- 3: Influencia o presencia media.
- 4: Influencia o presencia significativa.

5: Influencia o presencia fuerte.

Comentarios:

T1: El sistema no es distribuido.

T2: Los tiempos de respuesta deben ser muy altos, debe ofrecer un buen rendimiento.

T3: El usuario final requiere la adecuada preparación para usar el sistema.

T4: El procesamiento interno es medio, puesto que para la gestión de documentos se utiliza un framework como subsistema base.

T5: El código es reutilizable, ya que puede ser utilizado en desarrollos similares.

T6: Aunque son varios los módulos a instalar, el trabajo de instalación es muy sencillo.

T7: El sistema cuenta con una interfaz gráfica y amigable, muy sencilla de utilizar.

T8: El sistema debe ser multiplataforma, siendo usado bajo los Sistemas Operativos Windows y Linux.

T9: Se le podrá realizar algún cambio específico, no todos los cambios de forma general.

T10: Algunos procesos a los que accede el usuario dependen de que esté funcionando otro.

T11: Se crean diferentes cuentas de usuario y se asignan a cada uno los permisos pertinentes, mostrándose a cada usuario sólo las funcionalidades del sistema sobre las cuales tiene permiso de acceso.

T12: No provee acceso a terceras partes.

T13: No es tan necesario dale preparación a los usuarios para poder usar el sistema.

Por lo que: $TCF = 0.6 + 0.01 * \Sigma (\text{peso} * \text{valor asignado})$

$$TCF = 0.6 + 0.01 * 49$$

$$TCF = 0.6 + 0.49$$

$$TCF = 1.09$$

➤ **Factor de ambiente (EF)**

El **factor de ambiente (EF)** está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante). Es importante tener en cuenta:

- ✓ Para los factores E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- ✓ Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.

- ✓ Para el factor E6, 0 significa requerimientos extremadamente inestables, 3 estabilidad media y 5 requerimientos estables sin posibilidad de cambios.
- ✓ Para el factor E7, 0 significa que no hay personal part-time (es decir todos son full-time), 3 significa mitad y mitad y 5 significa que todo el personal es part-time (nadie es full-time).
- ✓ Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil. **(18)**

Tabla 6: Factor de Ambiente.

Factor	Descripción	Peso	Valor asignado	Total
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	3	4,5
E2	Experiencia en la aplicación.	0.5	1	0.5
E3	Experiencia en orientación a objetos.	1	4	4
E4	Capacidad del analista líder.	0.5	5	2,5
E5	Motivación.	1	4	4
E6	Estabilidad de los requerimientos.	2	2	4
E7	Personal part-time.	-1	2	-2
E8	Dificultad del lenguaje de programación.	-1	3	-3
	Total		24	14.5

Comentarios:

- E1:** El equipo se encuentra familiarizado con el modelo utilizado.
- E2:** No se tiene experiencia en el trabajo con aplicaciones similares.
- E3:** Hay una experiencia moderada con la programación orientada a objetos.
- E4:** El analista líder es una persona capacitada.
- E5:** Hay una alta motivación en el equipo de trabajo para la construcción del producto final.
- E6:** Los requerimientos no cambian, ni cuando se implemente el sistema.
- E7:** Los miembros no trabajan a tiempo completo.
- E8:** Se programa con C++, un lenguaje con mucha complejidad.

Por lo que: $EF = 1.4 - 0.03 * \Sigma$ (peso*valor asignado)

$$EF = 1.4 - 0.03 * 14.5$$

$$EF = 1.4 - 0.435$$

$$EF = 0.965$$

Una vez que se tiene el **factor de ambiente** y el **factor de complejidad técnica**, conjuntamente con los **puntos de casos de uso sin ajustar** que se habían calculado en el paso anterior, se puede calcular entonces los **puntos de casos de uso ajustados**:

$$UCP = UUCP * TCF * EF$$

$$UCP = 38 * 1.09 * 0.965$$

$$UCP = 39.970$$

➤ **Esfuerzo horas-hombre (E)**

El esfuerzo en horas-hombre viene dado por: **$E = UCP * CF$**

Donde:

E: Esfuerzo Estimado en Horas-Hombre

UCP: Puntos de Casos de Uso Ajustados

CF: Factor de Conversión

Para obtener el factor de conversión (CF) se cuentan cuántos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3) y los que están por arriba de la media para los restantes (E7, E8).

- ✓ Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso.
- ✓ Si el total es 3 ó 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso.
- ✓ Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

Total EF = Cantidad EF < 3 (entre E1 –E6) + Cantidad EF > 3 (entre E7, E8)

Total EF = 2 + 0

Total EF =2

En este caso se puede decir que: **CF = 20 Horas-Hombre / Punto de Casos de uso**

Por lo que:

$$E = UCP * CF$$

$$E = 39.970 * 20$$

$$E = 799.4 \text{ Horas-Hombre}$$

➤ **Esfuerzo de todo el proyecto**

Para una estimación más completa de la duración total del módulo, hay que agregar a la estimación del esfuerzo obtenida por los puntos de casos de uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de dicho módulo, según la siguiente aproximación: **(18)**

Tabla 7: Esfuerzo del Proyecto

Actividad	Porcentaje	Horas-Hombre
Análisis	10.00 %	199.85
Diseño	20.00 %	399.70
Programación	40.00 %	799.40
Pruebas	15.00 %	299.77
Sobrecarga(otras actividades)	15.00 %	299.77
Total	100 %	1998.51

Suponiendo que la jornada laboral del proyecto es de 8 horas, y un mes tiene como promedio 24 días laborales; la cantidad de horas que puede trabajar una persona en 1 mes es 192 horas.

Si el esfuerzo de todo el proyecto es de **1998.51 horas hombre** y por cada 192 horas se tiene 1 mes, eso daría un **TEF = 10 mes-hombre**.

Después de observar y estudiar los resultados obtenidos se puede llegar a la conclusión de que una persona se demoraría en realizar el módulo 10 meses. Pero como en el módulo se cuenta con más

personal asignado para el desarrollo del mismo se espera que se realice en menos tiempo de no existir contratiempos.

3.2. Técnicas de validación de requisitos

La validación es la etapa final y su objetivo es, ratificar los requerimientos, es decir, verificar todos los requerimientos que aparecen en el documento especificado para asegurarse que representan una descripción, por lo menos, aceptable del sistema que se debe implementar. La validación de requisitos es una actividad muy importante, pues un levantamiento de requisitos con errores que no se detecten a tiempo, además de no conducir a resultados inesperados provoca costos excesivos y gran pérdida de tiempo es por eso que los costos de errores en los requerimientos son altos, por lo cual la validación es muy importante.

Además es un proceso que tiene como misión demostrar que la definición de los requisitos define realmente lo que el cliente necesita, revisa que no se haya omitido ninguno, que no sean ambiguos, inconsistentes o redundantes, realistas, verificables y garantiza que todos los requerimientos presentes en el documento de especificación sigan los estándares de calidad. La actividad de validación tiene como entrada el documento de requisitos, los estándares relacionados y el conocimiento de la organización, y como salida se obtiene una lista de problemas y una lista de acciones recomendadas tal y como se muestra la figura # 5. (19)

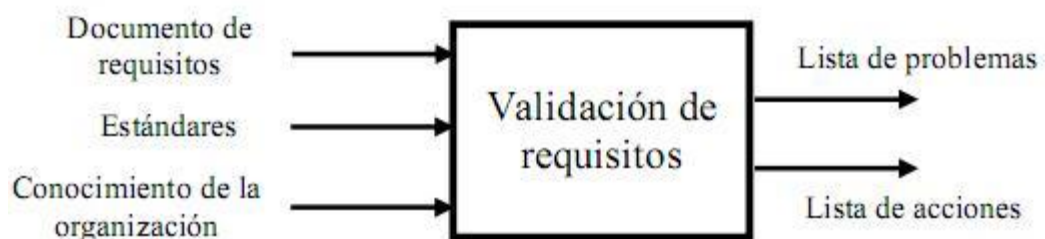


Figura 5: La validación en el proceso de requisitos

Hoy en día existen diferentes técnicas y métodos de validación entre los que se encuentran:

- ✓ Revisiones de requerimientos
- ✓ Construcción de prototipos
- ✓ Listas de chequeo
- ✓ Matrices de trazabilidad
- ✓ Auditorías
- ✓ Generación de casos de prueba

3.2.1. Aplicación de las técnicas de validación de requisitos

Las técnicas de validación seleccionadas y aplicadas en la presente investigación para validar los requisitos fueron las listas de chequeos y las matrices de trazabilidad caracterizadas por sus buenos resultados, con ello se probaran si los requisitos definen el sistema propuesto y saber que requerimientos quedan cubiertos por una prueba, además de que se usan para ver si los casos de uso que se tienen satisfacen todos los requerimientos.

En la UCI existe un grupo de trabajo el cual se encarga de todo lo relacionado con la calidad, el mismo lleva por nombre Calisoft. Entre las tareas que desarrollan está la revisión de la documentación a través de listas de chequeo definidas, propias para cada documento o artefacto generado por RUP.

Estas listas de chequeo no son más que una serie de preguntas a realizar, o una serie de aspectos a considerar, los cuales pretenden apoyar al analista durante la identificación de defectos en la especificación. A través de ellas se verifica si existe necesidad de la elaboración de determinadas disposiciones, además del cumplimiento de algunas reglas establecidas. Ayudan a validar y desarrollar nuevos requerimientos. Permite comprender aquellos requerimientos que no estén muy claros y que sean de alta volatilidad. Debido a la experiencia obtenida al aplicar las listas de chequeos se muestra que son las que proporcionan siempre los mejores resultados. En el [Anexo # 4](#) se puede encontrar las listas de chequeos aplicadas a los requisitos obtenidos con el objetivo de verificar que estén correctos. Una vez validados, los requisitos están cerrados y listos para ser enviados al equipo de análisis y desarrollo.

Obteniéndose como resultado que los requisitos definidos realmente eran los que el cliente deseaba, y que las funcionalidades tengan la integridad necesaria. Además se garantizó que los requisitos cumplieran cada elemento que debe describirse en la especificación de requisitos.

Otra de las técnicas aplicada fue la matriz de trazabilidad que consiste en marcar los casos de uso del sistema y chequearlos contra los requisitos del mismo. Es necesario ir viendo que cada caso de uso cubre cada requisito, de esta forma se podrán detectar inconsistencias u objetivos no cubiertos.

La matriz de trazabilidad es una técnica que liga los requisitos a los diferentes elementos del desarrollo. Por ejemplo se puede hacer una matriz de trazabilidad de cada requisito con los casos de uso en los que aparece. Además se pueden trazar varios tipos de matrices de trazabilidad. Algunos ejemplos pueden ser:

(20)

- ✓ Requisitos frente fases de desarrollo del sistema.
- ✓ Requisitos frente Actores.

- ✓ Requisitos frente casos de uso.
- ✓ Requisitos frente entidades del sistema.
- ✓ Requisitos frente procesos de negocio.
- ✓ Requisitos frente interfaces del sistema.
- ✓ Requisitos frente módulos.

En este caso se realizaron los ejemplos de requisitos frente actores y requisitos frente caso de uso por posibilitar ver el avance en el desarrollo de los requerimientos, además de que se puede verificar si los casos de uso son consistentes con los requerimientos. A continuación se muestran las matrices de trazabilidad realizadas en la presente investigación como se muestra en la tabla 8 y 9.

Tabla 8: Requisitos Funcionales y Caso de Uso del Sistema

Requisitos	CU1(CC)	CU2(VI)	CU3(MI)	CU4(MI)	CU5(EA)	CU6(IA)	CU7(GE)
RF1	x						
RF2	x						
RF3		x					
RF4			x				
RF5			x				
RF6			x				
RF7			x				
RF8				x			
RF9				x			
RF10				x			
RF11				x			
RF12					x		
RF13						x	
RF14							x
RF15							x
RF16							x
RF17							x

Actores	CU1	CU2	CU3	CU4	CU5	CU6	CU7
Cliente	X	x	x	x	x	x	x

Tabla 9: Actores del Sistema y Casos de Uso del Sistema

3.3. Conclusiones Parciales

En el presente capítulo se comprobó la calidad de los trabajos realizados en los capítulos anteriores, mediante aplicación de las técnicas de validación y método de estimación, lo que permitió que la documentación necesaria se generara con gran calidad. Esto permitió que los desarrolladores aprobaran el diagrama de casos de uso del sistema, hilo principal por el cual se guían estos para desarrollar el módulo. También se arribó a la conclusión de que si es factible el desarrollo del módulo propuesto ya que se obtuvo una cifra considerable en el esfuerzo total del proyecto.

Conclusiones Generales

Luego de concluido todo el proceso de análisis del módulo que permite la gestión de la información geológica para el software minero Geolmin, como objetivo principal de esta investigación, se arriban a las siguientes conclusiones:

- ✓ Se realizó una valoración de los sistemas mineros existentes en el mundo y en el país, determinándose que estos sistemas no cumplían con las características de la actividad minera en general y con los requerimientos del sistema a desarrollar.
- ✓ Se analizó las tendencias, tecnologías y herramientas propuestas por el grupo de arquitectura del proyecto para realizar el modelado, teniéndose como herramientas: Visual Paradigm para el modelado y como metodología de desarrollo RUP y UML como lenguaje de modelado. Lo cual contribuyeron y tributaron a los principios de independencia tecnológica que la Universidad ha defendido desde sus inicios.
- ✓ Una vez analizadas las tendencias, tecnologías y herramientas, se procedió al levantamiento de requisitos, permitiendo obtener las funcionalidades, y se realizó la documentación referente al análisis de sistema, logrando la descripción textual de todos los casos de usos.
- ✓ Por último este trabajo permitió obtener una evaluación de la calidad de los procesos realizados mediante técnicas de validación para la calidad de la especificación de los requisitos, además de que se aplicaron métodos de estimación para la factibilidad del proyecto.
- ✓ De modo general, permitió la elaboración de todos los artefactos correspondientes al rol de analista de sistemas, logrando que se realice la implementación sin ningún tipo de inconveniencia. Obteniendo como resultado una amplia y organizada documentación sobre el módulo que permite la gestión de la información geológica para el software minero Geolmin y cumpliendo con el objetivo propuesto al iniciar la investigación.

Recomendaciones

Partiendo de la experiencia acumulada durante todo el período de desarrollo de la investigación y de los resultados obtenidos, se proponen las siguientes recomendaciones:

- ✓ La utilización de esta investigación como bibliografía para posibles investigaciones referentes al tema de la minería.
- ✓ Continuar la investigación enfocada a redefinir los requisitos funcionales con el cliente, con el objetivo de aumentar las funcionalidades del sistema a desarrollar y lograr una mayor satisfacción del cliente.
- ✓ Que se siga con el resto de las fases de desarrollo de la metodología RUP para que pueda ser puesto en funcionamiento.
- ✓ Se actualice la documentación en correspondencia con las nuevas necesidades que plantee el cliente.

Referencias Bibliográficas

1. **Herbert., Juan Herrera.** *"Tecnología Minera"*. 2006.
2. **UCI.** *Portal UCI.* [En línea] <http://www.uci.cu/>.
3. **Susanne Griem Klee.** [En línea] <http://plata.uda.cl/minas/index.html>.
4. **Informática Aplicada a la Minería.** [En línea] 8 de 3 del 2010.
<http://www.cae.com/en/mining/products.asp>.
5. **Radomiro Tomic.** *El mercado de los Software mineros.* [En línea] 2008.
<http://informaticaminera.blogspot.com/2008/08/el-mercado-de-los-software-mineros.html>.
6. **Zulma Cataldi. Facultad de Ingeniería - Universidad de Buenos Aires.** [En línea] 2000.
<http://laboratorios.fi.uba.ar/lsi/cataldi-tesisdemagistereninformatica.pdf>.
7. **Juan Pablo Gómez Gallego. Scribd.** [En línea] 9 de 16 del 2007.
<http://www.scribd.com/doc/297224/RUP>.
8. **María A. Sánchez Mendoza.** *Informatizate.* [En línea] 6 de 7 del 2007.
http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
9. **Booch, G. Rumbaugh, J y Jacobson, I.** *"El Lenguaje Unificado de Modelado"*. 2000.
10. **Capítulo I HERRAMIENTAS CASE . Ingeniería de Software I.** [En línea] 5 de 22 del 2008.
<http://www.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>.
11. **Producto, Web Site Oficial. Visual Paradigm.** [En línea] <http://www.visual-paradigm.com/product/vpuml/>.
12. **Escuela de Postgrado. Instituto Tecnológico de Buenos Aires. Centro de Ingeniería del Software e Ingeniería del Conocimiento (CAPIS).** [En línea] <http://www.itba.edu.ar/capis/webcapis/planma.html>.
13. **Hernández, Sigifredo.E Badani.** *Métricas de estimación de tamaño.* 2002.
14. **Conferencia 2. Fase de Inicio. Modelo del Negocio. Ingeniería de Software I.** Informáticas., Universidad de las Ciencias. curso 2007-2008.
15. **Conferencia 3: Flujo de trabajo de requerimientos. Ingeniería de Software II.** . Informáticas., Universidad de las Ciencias. curso 2009-2010.
16. **Conferencia 6. Fase de Inicio. Flujo de Análisis y Diseño. Modelo de Análisis. Ingeniería de Software I.** Informáticas., Universidad de las Ciencias. curso 2007-2008.
17. [En línea] <http://vital.usalesiana.edu.bo..>
18. **(CAPIS), Escuela de Postgrado. Instituto Tecnológico de Buenos Aires. Centro de Ingeniería del Software e Ingeniería del Conocimiento.** [En línea] <http://www.itba.edu.ar/capis/webcapis/planma.html>.

19. **Distancia, Master de Ingeniería de Software – a. Módulo II Requisitos de sistemas de software.**
20. **Michael Arias Chaves Revista InterSedes © Universidad de Costa Rica ISSN 1409-4746. La Ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software.** [En línea] 26 de 07 del 2007. <http://www.intersedes.ucr.ac.cr>.

Bibliografía Consultada

- **Herrera Herbert, Juan.** *"Tecnología Minera"*. 2006. [Citado el: 10 de 11 de 2010.]
- **Plá Ortiz de Urbina, Fernando.** *"Tendencias del laboreo de minas"*. 1994.
- **Crespo Martínez, Gilberto.** *"Industria Minera"*. México: s.n., 1903.
- **José R. Vilar.** 2008. El papel de las TIC en la empresa del SXXI. *Tecnobiz*. [En línea] 2008 <http://www.tecnobiz.com/el-papel-de-las-tic-en-las-empresas>.
- **Sampieri, Roberto Hernández, Collado, Carlos Fernández y Lucio, Pilar Baptista.** Metodología de la Investigación. [Libro digital] México, D. F.: MCGRAW-HILL INTERAMERICANA EDITORES, S. A. de C. V., 1998. ISBN 970-10-1899-0.
- **Rodríguez, H.** *"Premisas Geológicas para la Automatización integral de los trabajos de Prospección en Yacimientos Niquelíferos de Intemperismo"*. 1990.
- **Tomic, Radomiro.** *El mercado del software minero*. 2008. [Citado el: 23 de 11 de 2010.]
- **Pressman, Roger S.** 2005. *Ingeniería del Software .Un enfoque práctico. Quinta Edición*. 2005.
- **RUP.** *Ayuda RUP Español 2007*. 2006.
- **Int. RUP.** 2005. Portal Desarrollo Software. [En línea] 22 de 09 de 2005. <https://pid.dsic.upv.es/C1/Material/Documentos Disponibles/Introducción a RUP.doc>.
- **Metodologías - RUP.** 2007. Consultoría en Metodologías de Desarrollo de Software - RUP y las mejores prácticas para el desarrollo de software. *Hista Internacional S.A.* [En línea] 27 de 2 de 2007 <http://www.histaintl.com/servicios/consulting/rup.php>.
- **Sánchez, María A. Mendoza.** 2007. Informatizate. [En línea] 7 de 6 de 2007. [Citado el: 14 de 1 de 2011.] http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
- **Capítulo I HERRAMIENTAS CASE.** Ingeniería De Software I. [En línea] 22 de mayo de 2008. [Citado el: 19 de 1 de 2011.] <http://www.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>.
- **Booch, Grady, Rumbaugh, Jim y Jacobson, Ivar.** El Lenguaje Unificado de Modelado. 2000.
- **Visual Paradigm.** [En línea] <http://www.visual-paradigm.com/>
- **Producto, Web Site Oficial.** Visual Paradigm. [En línea] [Citado el: 22 de 1 de 2011.] <http://www.visual-paradigm.com/product/vpuml/>.

- **Personales.unican.** *Trabajando con Visual Paradigm para UML.* [En línea]
<http://personales.unican.es/ruizfr/is1/doc/lab/01/is1-p01-trans.pdf>.
- **Conferencia 1.Introducción a la Ingeniería de Software.** Preparación para las Pruebas de Nivel de Ingeniería de Software. [En línea] 2007-2008. <http://eva.uci.cu/course/view.php?id=58>.
- **Conferencia 1: Culminación de la Fase de Elaboración.** Flujo de Trabajo Análisis y Diseño. Ingeniería de Software II. Universidad de las Ciencias Informáticas: s.n., curso 2009-2010.
- **Conferencia 2. Fase de Inicio. Modelo del Negocio.** Ingeniería de Software I. Universidad de las Ciencias Informáticas, curso 2007-2008. [Citado el: 12 de 2 de 2011.]
- **Conferencia 3: Flujo de trabajo de requerimientos.** Ingeniería de Software II. Universidad de las Ciencias Informáticas : s.n., curso 2009-2010.[Citado el: 18 de 3 de 2011]
- **Conferencia 6. Fase de Inicio. Flujo de Análisis y Diseño. Modelo de Análisis.** Ingeniería de Software I. Universidad de las Ciencias Informáticas: s.n., curso 2007-2008. [Citado el: 27 de 3 de 2011]
- **Escuela de Postgrado. Instituto Tecnológico de Buenos Aires.** Centro de Ingeniería del Software e Ingeniería del Conocimiento (CAPIS) [Citado el: 14 de 04 de 2011.]
<http://www.itba.edu.ar/capis/webcapis/planma.html>
- **Michael Arias Chaves Revista InterSedes © Universidad de Costa Rica ISSN 1409-4746.** La Ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. . [En línea] 26 de 07 de 2007 [Citado el: 5 de 05 de 2011.] <http://www.intersedes.ucr.ac.cr>
- **Ian Sommerville** 1995 5ta Edición Capitulo 4 Ingeniería de Software.
- **Máster de Ingeniería de Software – a Distancia.** Módulo II Requisitos de sistemas de software. [Citado el: 7 de 05 de 2011.]

Anexos

Anexo#1: Descripción textual de los casos de uso del sistema.

Descripción del caso de uso: Validar información

Caso de Uso:	Validar la información.	
Actores:	Cliente	
Resumen:	El caso de uso se inicia cuando el usuario selecciona la opción de Validar información donde se validara que los ficheros no tengan intervalos solapados y finaliza cuando el sistema emite un mensaje de que la información es correcta.	
Precondiciones:	Se tiene que haber cargado los ficheros.	
Referencias	RF3.	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando el usuario selecciona en el menú base de datos la opción de validar información.	2. El sistema en los ficheros de Litología, Inclínometría y Muestras valida que no existan intervalos solapados.	
	3. El caso de uso termina cuando el sistema emite un mensaje "La información es correcta" sino existen intervalos solapados.	
Prototipo de interfaz		

ID	FROM	TO	ROCK
DH2675	82.0	100.0	6.0
DH2675	100.0	112.0	8.0
DH2737	64.0	86.0	6.0
DH2737	86.0	116.0	8.0
DH2812	65.6	78.0	6.0
DH2812	78.0	106.0	8.0
DH2813	87.4	102.0	6.0

Validar

Flujos Alternos

2. Si existen intervalos solapados el sistema muestra un mensaje especificando los intervalos solapados y en el fichero que se encuentra.

Poscondiciones Se valida la información.

Descripción del caso de uso: Mostrar información

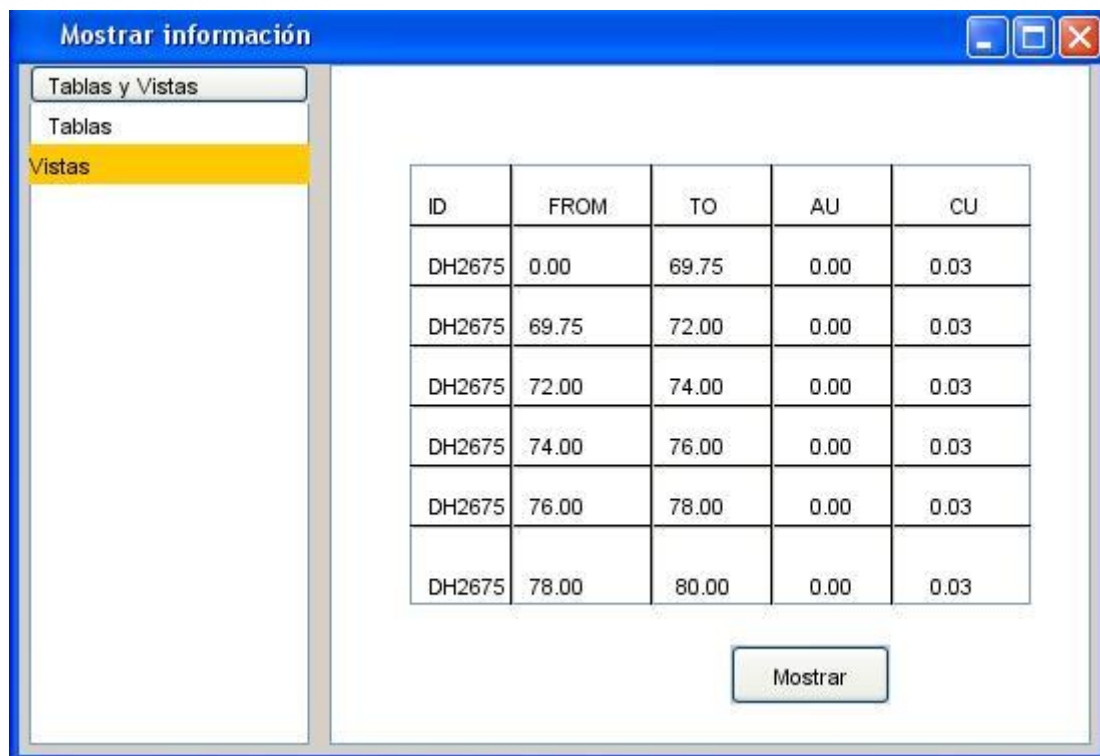
Caso de Uso:	Mostrar información.
Actores:	Cliente.
Resumen:	El caso de uso se inicia cuando el usuario selecciona la opción de Mostrar información donde se debe de especificar la información que quiere que se muestre y finaliza cuando el sistema le muestra la información seleccionada.
Precondiciones:	Se tiene que contar con la información para mostrar.

Referencias	RF8, RF9, RF10, RF11.
Prioridad	Crítico.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el usuario selecciona en el panel de tablas y vistas de la pestaña base de datos la opción "Vistas".	2. El sistema muestra en forma de árbol la información.
3. El usuario selecciona la información que desea mostrar sin poder editar ningún campo.	4. El caso de uso finaliza cuando el sistema muestra en otro panel la información.

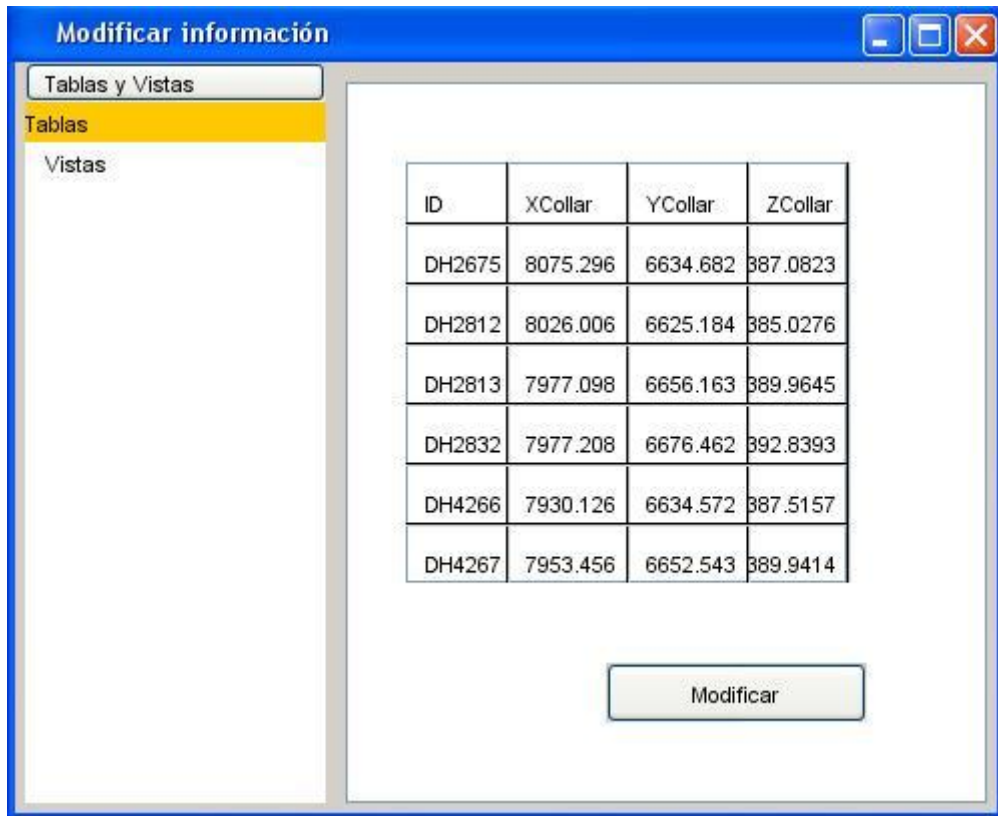
Prototipo de interfaz



Poscondiciones	Se muestra la información deseada por el usuario.
-----------------------	---

Descripción del caso de uso: Modificar información

Caso de Uso:	Modificar la información.	
Actores:	Cliente.	
Resumen:	El caso de uso se inicia cuando el usuario selecciona la opción de Modificar información y luego selecciona la información a modificar y finaliza cuando el sistema modifica la información en la base de datos.	
Precondiciones:	Se tiene que contar con la información para modificar.	
Referencias	RF4, RF5, RF6, RF7.	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando el usuario selecciona en el panel tablas y vistas de la pestaña base de datos la opción "Tablas".	2. El sistema despliega en forma de árbol las tablas que podrán ser editadas en la base de datos (Coordenada, Muestra, Litología e Inclínometría).	
3. El usuario selecciona la tabla en la que desea modificar la información.	4. El sistema muestra en otro panel la tabla con la información dejando editable solo los campos que pueden ser modificados.	
5. El usuario modifica la información deseada.	6. El sistema verifica que los datos insertados sean correctos.	
	7. El caso de uso termina cuando el sistema modifica la información en la base de datos de ser correctos los datos insertados.	
Prototipo de interfaz		



Flujos Alternos

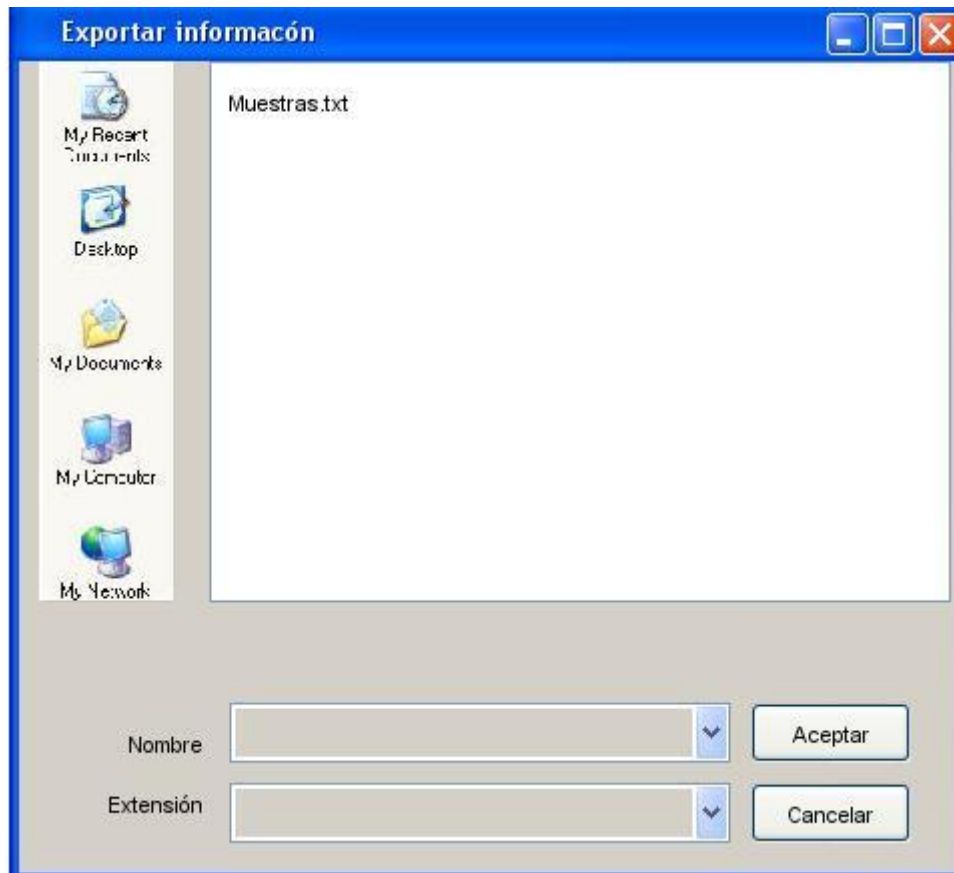
6. Si los datos insertados no son correctos el sistema muestra un mensaje especificando: “Los datos introducidos no son correctos”.

Poscondiciones Se modifica la información en la base de datos.

Descripción del caso de uso: Exportar archivos

Caso de Uso:	Exportar archivos.
Actores:	Cliente
Resumen:	El caso de uso se inicia cuando el usuario selecciona la opción de Exportar archivos donde se exportaran los ficheros y finaliza cuando el sistema exporta el archivo.

Precondiciones:	Se tiene que contar con los archivos a exportar.	
Referencias	RF12.	
Prioridad	Secundario.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando el usuario selecciona en el menú Importar/Exportar la opción de exportar archivos.	2. El sistema muestra una interfaz con los objetos para que el usuario escoja el objeto que desea exportar.	
3. El usuario selecciona el objeto que desea exportar.	3. El sistema muestra una interfaz para seleccionar la extensión a la que se va a exportar que puede ser texto limitado o XML, nombre y el directorio	
5. El usuario selecciona la extensión, nombre, directorio y da click en "Aceptar".	6. El caso de uso finaliza cuando el sistema exporta el archivo.	
Prototipo de interfaz		



Flujos Alternos

5. Si el usuario no da clic en aceptar y da en cancelar. 6. El sistema cierra la interfaz.

Poscondiciones Se exportan los archivos.

Descripción del caso de uso: Importar archivos

Caso de Uso:	Importar archivos.	
Actores:	Cliente	
Resumen:	El caso de uso se inicia cuando el usuario selecciona la opción de Importar archivos donde se importara información para luego trabajar con dicha información y finaliza cuando el sistema importa el archivo.	
Precondiciones:	Se tiene que contar con los archivos a importar.	
Referencias	RF13.	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando el usuario selecciona en el menú Importar/Exportar la opción de importar archivos.	2. El sistema muestra una interfaz con los objetos para que el usuario escoja el objeto que desea importar.	
	3. El sistema muestra una interfaz para seleccionar la extensión a la que se va a importar que puede ser texto limitado o XML, nombre y el directorio.	
4. El usuario selecciona el archivo a importar y da clic en "Aceptar".	5. El sistema valida los datos del archivo e importa el archivo.	
	6. El caso de uso finaliza cuando el sistema muestra el archivo importado.	
Prototipo de interfaz		



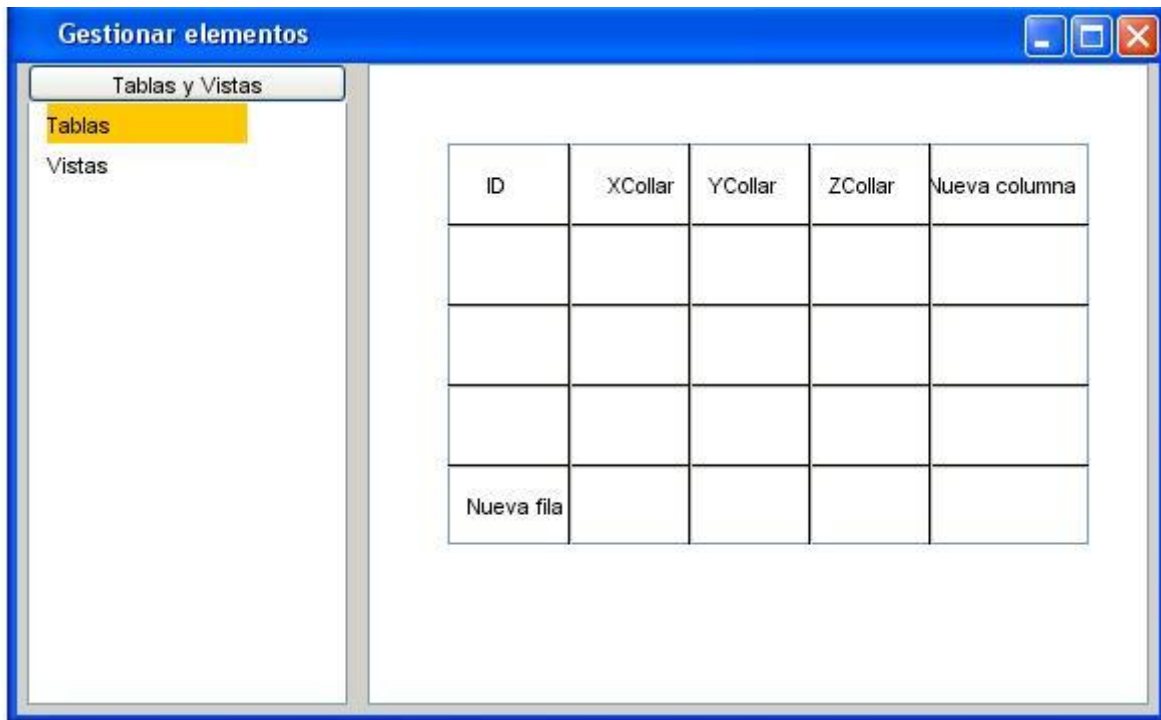
Flujos Alternos

4. El usuario da clic en "Cancelar".	El sistema cierra la interfaz.
	5. Si los datos validados no son correctos el sistema muestra un mensaje especificando: "Los datos validados no son correctos".
Poscondiciones	Se importan los archivos.

Descripción del caso de uso: Gestionar elementos de tabla de la BD

Caso de Uso:	Gestionar elementos de una tabla de la base de datos.
Actores:	Cliente.
Resumen:	El caso de uso comienza cuando el usuario selecciona la opción de Gestionar elementos de una tabla de la base de datos, dentro del mismo se selecciona insertar o eliminar fila o insertar o eliminar columna y finaliza cuando el sistema termina las operaciones guardando los cambios.
Precondiciones:	-
Referencias	RF14, RF15, RF16, RF17.

Prioridad	Crítico
Flujo Normal de Eventos	
1. El caso de uso comienza cuando el usuario decide gestionar un elemento de una tabla de la BD.	2. El sistema muestra otro panel con las tablas y las opciones de insertar fila, eliminar fila, insertar columna y eliminar columna de una tabla. - Si el usuario escoge Insertar fila de una tabla ver Sección "Insertar fila". - Si el usuario escoge Eliminar fila de una tabla ver Sección "Eliminar fila". - Si el usuario escoge Eliminar columna de una tabla ver Sección "Eliminar columna". - Si el usuario escoge Insertar columna de una tabla ver Sección "Insertar columna".
Sección "Insertar fila"	
Acción del Actor	Respuesta del Sistema
1. El usuario desea adicionar una fila a una tabla.	
2. El usuario da clic en "Insertar", adiciona una nueva fila.	3. El caso de eso finaliza cuando el sistema añade la nueva fila a la tabla.
Prototipo de interfaz	



Sección “ Eliminar fila”

Acción del Actor	Respuesta del Sistema
1. El usuario desea eliminar una fila de una tabla.	2. El sistema muestra las filas existentes y habilita la opción Eliminar.
3. El usuario selecciona la fila que desea eliminar da clic en “Eliminar”.	4. El caso de uso finaliza cuando el sistema elimina la fila seleccionada.

Flujos Alternos

3. El usuario no da clic en Eliminar.	4. El sistema muestra el mensaje “No se ha eliminado la fila” y retorna al paso #3 del flujo central de la sección.
---------------------------------------	---

Sección “Eliminar columna”

Acción del Actor	Respuesta del Sistema
1. El usuario desea eliminar una columna	2. El sistema muestra las columnas existentes y habilita

de una tabla.	la opción Eliminar.
3. El usuario selecciona la columna que desea eliminar da clic en "Eliminar".	4. El caso de uso finaliza cuando el sistema elimina la columna seleccionada.
Flujos Alternos	
3. El usuario no da clic en Eliminar.	4. El sistema muestra el mensaje "No se ha eliminado la fila" y retorna al paso #3 del flujo central de la sección.
Sección "Insertar columna"	
Acción del Actor	Respuesta del Sistema
1. El usuario decide insertar una columna a una tabla de la BD.	2. El sistema muestra una interfaz para que introduzca los parámetros: <ul style="list-style-type: none"> • Nombre: • Tipo de dato: • La opción de marcar si es no nulo o nulo:
3. El usuario introduce los parámetros y da clic en "Aceptar".	4. El sistema valida los parámetros introducidos e inserta la columna
Flujos Alternos	
3. El usuario da clic en el botón "Cancelar".	El sistema cierra la interfaz.
	4. Si los datos introducidos son incorrectos el sistema muestra el mensaje "Debe introducir correctamente los parámetros" y retorna al paso #3 del flujo central de la sección.
Poscondiciones	Se gestionan los elementos de una tabla.

Anexo#2: Diagramas de Clases del Análisis.

Diagrama de clases del análisis: Validar información

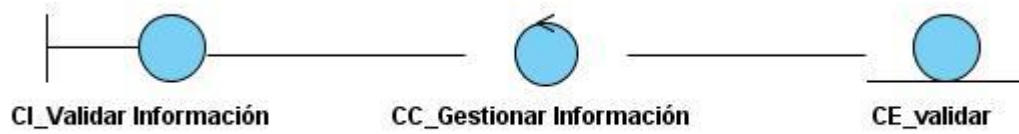


Diagrama de clases del análisis: Mostrar información



Diagrama de clases del análisis: Modificar información



Diagrama de clases del análisis: Exportar archivo



Diagrama de clases del análisis: Importar archivo



Diagrama de clases del análisis: Gestionar elementos de una tabla de la BD



Anexo#3: Diagramas de Colaboración.

Diagrama de colaboración: Validar información



Diagrama de colaboración: Mostrar información



Diagrama de colaboración: Modificar Información



Diagrama de colaboración: Exportar archivo

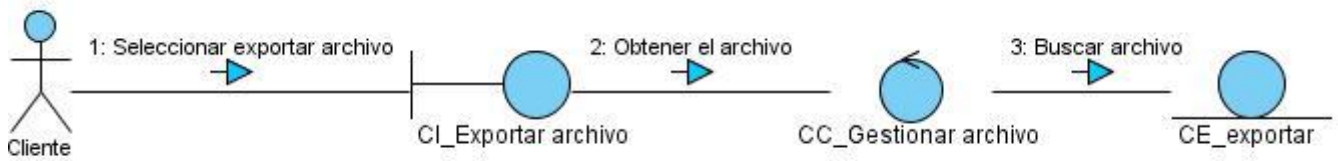


Diagrama de colaboración: Importar archivo

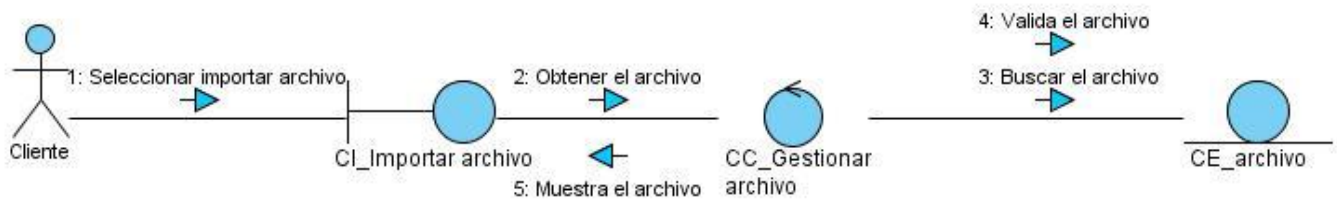


Diagrama de colaboración: Gestionar elementos de una tabla de la BD



Anexo#4: Lista de chequeo de requisitos.

Evidencia	Nivel de importancia	Procedimiento	Respuesta
1. ¿Se refleja el seguimiento del documento?	B	Revisión del documento	Sí
2. ¿Se especifica el número de cada requisito evaluado?	A	Revisión del documento	Sí
3. ¿Se especifica el nombre de cada Requisito Solicitado?	A	Revisión del documento	Sí
4. ¿Se especifica si están identificados los elementos de entrada de cada requisito?	A	Revisión del documento	Sí
5. ¿Se especifica si están identificados los elementos de salida de cada requisito?	A	Revisión del documento	Sí
6. ¿Se especifica si cada requisito no es ambiguo?	A	Revisión del documento	Sí
7. ¿Se especifica si cada requisito es correcto?	A	Revisión del documento	Sí
8. ¿El requisito puede ser verificado?	A	Revisión del documento	Sí

Lista de Chequeo de Especificación de Requerimiento

Evidencia	Nivel de importancia	Procedimiento	Respuesta
¿Existe(n) el(los) documento(s) de "Especificación de Requisitos"?	A	Revisión del documento	Sí
¿Se realiza el registro de control de versiones de los cambios del documento?	M	Revisión del documento	Sí

¿Se define el alcance del documento?	B	Revisión del documento	Sí
¿Se realiza una descripción de la funcionalidad de los requisitos en lenguaje natural?	M	Revisión del documento	Sí
¿Están descritos todos y cada uno de los requisitos?	A	Revisión del documento	Sí
¿Se identifican los requisitos que afectan la usabilidad, fiabilidad, eficiencia o soporte del sistema?	M	Revisión del documento	Sí
¿Están especificadas, si existen, todas las restricciones relacionadas con el diseño del sistema?	M	Revisión del documento	Sí
¿Están especificados los requisitos para la documentación y ayuda de los usuarios?	M	Revisión del documento	Sí
¿Se identifican los componentes que han sido adquiridos o comprados para incorporarlos al sistema?	M	Revisión del documento	Sí
¿Están definidas todas las interfaces que deberán ser soportadas por el sistema (Usuario, Hardware, Software, Comunicación)?	M	Revisión del documento	Sí
¿Están especificados todos los requisitos legales, tales como patentes, derechos de autor, marca comercial, logotipo entre otras así como cualquier requisito de licencia o restricción de uso?	M	Revisión del documento	Sí
¿Existe evidencia o referencias de las normas o estándares aplicables al sistema, por ejemplo estándares legales, de calidad, normas de usabilidad, entre otros?	M	Revisión del documento	Sí