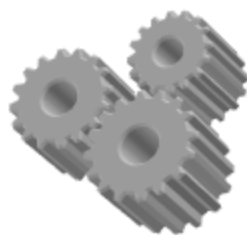




***Universidad de las Ciencias Informáticas
Facultad #5***

***Componente de Interpretación de registros
de pozos petroleros para el sistema AnPer.***



Autor(a):

❖ Yarisleydis Barzaga Pérez

Tutor:

❖ Ing. Yunier Alexander Pimienta Fernández.

Co-Tutor:

❖ Ing. Alexey Díaz Domínguez

Ciudad de La Habana, Junio 2011.

“Año 53 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Yarisleydis Barzaga Pérez

Firma del Autor

Yuniel Alexander Pimienta Fernández

Alexey Díaz Domínguez

Firma del Tutor

Firma del Co-Tutor



A mi mamá por ser una persona tan especial en mi vida, te amo tanto mamá que simplemente no son suficientes todas las palabras buenas que existen en el diccionario para decírtelas. Este trabajo es lo menos que te mereces, y es lo menos que vas a recibir de una hija que te agradece todo lo que es ahora.

A mi papá que me ayudó tanto, que me dio mucho su apoyo y su cariño, te quiero papá.

A mis abuelos por su cariño incomparable, por su amor incondicional y por creer desde un principio que yo sería capaz de lograr todo lo que he logrado hasta ahora y mucho más.

A mi novio por darme todo su apoyo, y aun teniendo un carácter tan diferente al mío fue capaz de comprenderme y de seguir junto a mí durante toda mi estancia en la Universidad.

Un agradecimiento muy especial a mis tres amigas incondicionales Lulú, Nelis y Liset, que me han ayudado, compartido, soportado todas mis locuras en las buenas y en las malas y por confiar en mí, "Las Quiero Mucho".

A mis compañeras de cuarto Eileen, Sulemy, Yaní por soportarme todos estos años y por compartir momentos importantes en mi vida.

A mis compañeras de apartamento Danae, Daylen (Elenita), Annia, Daylen, Lianet y Orly's por soportarme todos estos años, a ellas les agradezco todo el apoyo que me dieron cuando yo más lo necesitaba.

A todo los compañeros del tribunal que me ayudaron a realizar un trabajo profesional, corrigiendo los errores que cometía.





A todos mis compañeros de aula, principalmente a los que vienen conmigo desde primer año, con los que convivo desde 2do año y a los a los que conozco hace poco tiempo.

A los chicos de la Facultad 6: Roselí, Yordan, David (HI), Reynel y Camilo por alegrarme, y apoyarme tanto cuando estaba triste. Por ser tan buenos amigos, por su ayuda y por confiar en mí.

A Henry y a Adriél que aunque me llevo con ellos desde hace poco me han ayudado mucho en mi tesis, pues sin su ayuda creo que no me hubiera dado tiempo graduarme este año, Gracias.

A mis tutores Pimienta y Alexey que se han preocupado mucho por mi tesis, por guiarme por todo este camino tan difícil para mí. Aun cuando las luces se me apagaron y me sentía estresada, sin ganas de continuar, estuvieron ahí para encender aquella lucecita de la esperanza y de la capacidad de confiar en mí misma. Le agradezco mucho su apoyo, de verdad no creo que lo hubiera logrado sin su ayuda. Esos son los tutores que se quieren comprometidos.

A todos los amigos que se han preocupado de alguna manera por mis resultados. Que aunque sea alguna vez me preguntaron ¿Y la tesis, cómo va?



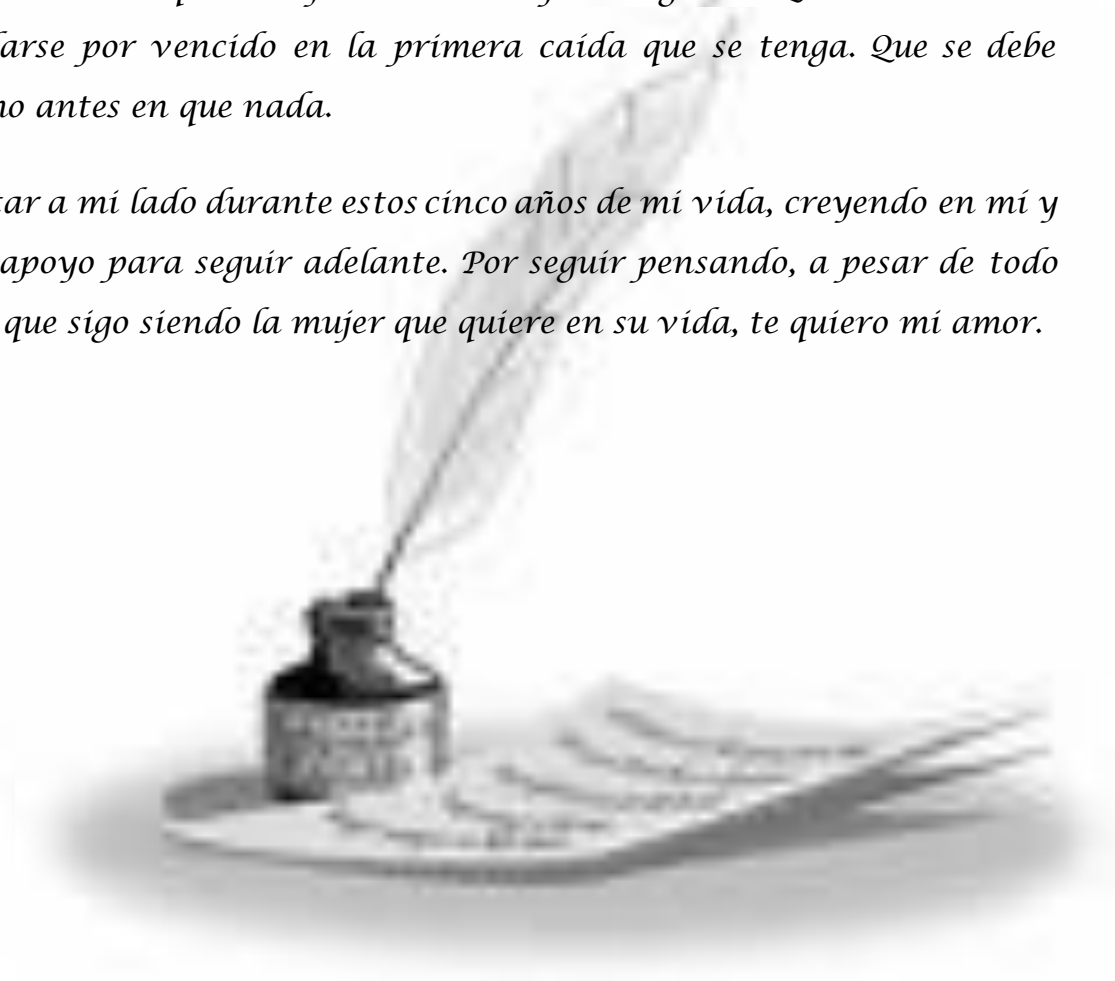


A la mujer, amiga y madre incondicional que ha hecho más allá de lo posible para que yo fuera hoy por hoy, no una niña ignorante sino una mujer formada, dedicada, profesional y de bien. A tí mamá, que siempre confiaste en mí, que en el momento en que ni yo misma me sentía con fuerzas para continuar, tú me abriste los brazos y no permitiste que me rindiera, todo lo contrario, tus palabras: “Confía en tí, estudia para que salgas bien”, fueron más que suficientes para que yo siguiera adelante. Para tí está dedicado este Trabajo de Diploma, te amo con la vida ♥Mamá ♥.

A mis abuelos queridos: Caridad (Cachita), Sabá y Víctor que son los seres más sagrados y especiales en esta vida. Gracias por todo su amor y dedicación hacia mí, los amo.

Para mí papí que me apoyó tanto en mis decisiones, y siempre me enseñó que para ser alguien en esta vida tienes que sacrificarte mucho para lograrlo. Que en la vida se debe luchar, no darse por vencido en la primera caída que se tenga. Que se debe confiar en sí mismo antes en que nada.

A mi novio por estar a mi lado durante estos cinco años de mi vida, creyendo en mí y dándome todo su apoyo para seguir adelante. Por seguir pensando, a pesar de todo este tiempo junto, que sigo siendo la mujer que quiere en su vida, te quiero mi amor.





La Unión Cuba-Petróleo (CUPET) es la entidad encargada del desarrollo de la industria del petróleo en Cuba. Esta cuenta con un Centro de Investigaciones del Petróleo (CEINPET), empresa encargada de dar respuesta de forma integral a toda la actividad petrolera. En el centro se realizan análisis e interpretación de registros, para ello utilizan software privativo, aunque les facilita el trabajo a los especialistas, tienen que pagar grandes sumas de dinero por sus licencias y actualizaciones. Además ocultan el mecanismo de su funcionamiento imposibilitando mejoras y limitando así el conocimiento del tipo de análisis realizado.

Con el fin de resolver estas deficiencias en la Universidad de las Ciencias Informáticas (UCI) se está desarrolla el software “Sistema para el Análisis e Interpretación de los Registros de Pozos” (AnPer). Este actualmente solo contiene los elementos de análisis, ubicándolo como un sistema incompleto. Por tal motivo se implementa el componente de interpretación el cual al integrarlo con AnPer agrupará las funcionalidades principales de los software propietarios. La puesta en marcha de dicho sistema permite cambiar la situación existente y agilizará el proceso de interpretación. Además de contar con servicios adaptados a las necesidades específicas de los especialistas. La implementación de este componente trajo consigo un ahorro en tiempo, mayor usabilidad y que el trabajo sea más cómodo y económico.

Palabras Clave: análisis, componentes, interpretación, petróleo, registros de pozos.



Introducción.....	1
Capítulo 1: Fundamentación Teórica.	6
1.1. Introducción.....	6
1.2. Conceptos asociados al dominio del problema.....	6
1.3. Objeto de estudio	9
1.3.1. Descripción general	10
1.3.2. Descripción actual del dominio del problema.....	11
1.4. Modelo de Interpretación	11
1.4.1. Descripción de las cuatro partes de un Modelo	12
1.4.1.1. Componentes de la formación	12
1.4.1.2. Ecuación de Respuesta.....	13
1.4.1.3. Parámetros	14
1.5. Métodos a utilizar en la Interpretación Petrofísica	14
1.6. Análisis de los sistemas utilizados en el mundo y en Cuba	15
1.6.1. AnPer	16
1.6.2. Petroluke	16
1.6.3. Elemental Analysis (ELAN).....	17
1.6.4. Interactive Petrophysics (IP).....	17
1.6.5. Neurolog.....	17
1.7. Tendencias y tecnologías actuales a utilizar	19
1.7.1. Metodología de desarrollo: Proceso Unificado de Desarrollo (RUP)	19
1.7.2. Lenguaje Unificado de Modelado, UML:	20
1.7.3. Herramienta Case: Visual Paradigm	21
1.7.4. Framework de Desarrollo: QT	23
1.7.5. Lenguaje de Programación: C++.....	25
1.7.6. Paradigma de Programación: Programación Orientada a Objeto (POO)	26
1.7.7. Entorno de Desarrollo: Eclipse	27
1.7.8. Gestor de Base de Datos: PostgreSQL	28
1.8. Conclusiones Parciales	28
Capítulo 2: Descripción y Análisis de la solución.	29
2.1. Introducción.....	29
2.2. Características de la solución propuesta	29



2.3.	Especificación de los requerimientos del componente	29
	Requerimientos Funcionales	29
	Requerimientos no Funcionales	30
2.4.	Modelo de Casos de Uso del Sistema	32
	Descripción de los actores del Sistema	32
	Casos de Uso del Sistema	32
	Diagrama de Casos de Uso del Sistema	33
2.5.	Modelo de Diseño	34
	Diagramas de Clases del Diseño	34
2.6.	Arquitectura de Software	35
	Estilos Arquitectónicos	36
	Arquitectura en Capas	37
2.7.	Visión general de la Arquitectura	38
	Descripción de las Capas y sus componentes	39
2.8.	Patrones de Diseño utilizados	39
2.9.	Algoritmo a utilizar: Descomposición (LU)	41
2.10.	Conclusiones Parciales	43
Capítulo 3:	Implementación y Validación de la solución	44
3.1.	Introducción	44
3.2.	Modelo de Implementación	44
	3.2.1. Vista de Implementación	44
3.3.	Diagrama de Componentes	45
3.4.	Diagrama de Despliegue	47
3.5.	Estructura de la Base de Datos	47
3.6.	Estándares de Codificación	48
3.7.	Validación del sistema propuesto	48
	3.7.1. Pruebas de Caja Negra	49
	3.7.1.1 Casos de Pruebas	49
3.8.	Conclusiones Parciales	58
Conclusiones Generales	60	
Recomendaciones	61	
Referencias Bibliográficas	62	



ANEXOS	66
GLOSARIO DE TÉRMINOS.....	73

**INDICE DE TABLAS**

Tabla 1 Descripción del actor del sistema.....	32
Tabla 2 Casos de Usos del sistema.....	33
Tabla 3 Caso de Prueba de la sección Crear Modelo del caso de uso “Gestionar Modelo”.	50
Tabla 4 Caso de Prueba de la sección Cambiar Nombre del caso de uso “Gestionar Modelo”.	50
Tabla 5 Caso de Prueba de la sección Eliminar Modelo del caso de uso “Gestionar Modelo”.	51
Tabla 6 Caso de Prueba de la sección Definir Volumen del caso de uso “Editar Modelo”.	52
Tabla 7 Caso de Prueba de la sección Definir Ecuación del caso de uso “Editar Modelo”.	53
Tabla 8 Caso de Prueba del caso de uso “Inicializar Parámetros”.	54
Tabla 9 Caso de Prueba del caso de uso “Ejecutar Modelo”.	56
Tabla 10 Tabla de no conformidades.	57
Tabla 11 Descripción del Caso de Uso Gestionar Modelo	68
Tabla 12 Descripción del Caso de Uso Editar Modelo	70
Tabla 13 Descripción del Caso de Uso Inicializar Parámetros.....	71
Tabla 14 Descripción del Caso de Uso Ejecutar Modelo	71
Tabla 15 Descripción del Caso de Uso Visualizar Resultados en gráficas de curvas.....	72

INDICE DE FIGURAS

Figura 1 Ecuación simple de respuesta lineal.....	13
Figura 2 Casos de uso del Negocio.	33
Figura 3 Diagrama de Clases del Diseño.....	35
Figura 4 Vista general de la arquitectura	38
Figura 5 Vista de Implementación	44
Figura 6 Diagrama de Componentes	46
Figura 7 Diagrama de Despliegue	47
Figura 8 Modelo Entidad Relación de la BD.	48

Introducción

El petróleo u oro negro es utilizado desde hace muchos años por el hombre debido a sus innumerables aplicaciones. Sus primeros indicios no se revelaron hasta el siglo XIX donde la Revolución Industrial exigía la búsqueda de un nuevo combustible para su creciente desarrollo. Hoy en día es usado como material energético y como materia prima para la fabricación de otros productos necesarios en la sociedad.

Cuba es uno de los países del mundo que más manifestaciones superficiales de petróleo tiene, principalmente en la mitad septentrional del territorio nacional, donde se ubica el primer descubrimiento en 1881 en la zona de Corralillo, provincia de Villa Clara. Sin embargo, fue después de 1960 que se inició un programa sistemático y detallado de exploración, con investigaciones y estudios geológicos y geofísicos, y la perforación de pozos profundos de carácter estratigráfico y de exploración. De esta forma se han delimitado dos grandes cuencas sedimentarias: la Cuenca Norte y la Cuenca Sur. Desde entonces se han perforado más de 2500 pozos de petróleo. Durante este proceso se genera abundante información, almacenada en diferentes registros de pozo, uno de ellos es el registro eléctrico que ayuda a conocer los tipos de formación y las características físicas de las rocas, tales como densidad, porosidad, contenidos de agua, de petróleo y de gas natural.

Por la complejidad de su extracción y la necesidad de contar con la presencia de este en el sector industrial y social; en el mundo existen empresas y transnacionales que se dedican a su extracción, procesamiento y comercialización. Entre muchas de las empresas que brindan servicios para el sector petrolero se encuentra **Schlumberger**. Esta desarrolla la plataforma **Geoframe** que se dedica a la caracterización integrada de yacimientos, permitiendo describir el subsuelo en forma precisa. **Geoframe** cuenta con un módulo de **Evaluación Petrofísica Elan**, que permite realizar una evaluación cuantitativa de formaciones y el **PetroView Plus**, programa de análisis de registro mediante el cual se ejecuta la interpretación determinística sobre un pozo (34).

Otras de las compañías reconocidas mundialmente en la industria petrolera y de gas es Halliburton. La cual ofrece soluciones para la gestión de los datos geológicos, la evaluación de perforación y formación, la construcción y terminación de un pozo, y la optimización de la producción a través de la vida del campo. Proporciona soluciones de evaluación de la formación adecuada para determinado problema (35).

La ejecución de los procesos petroleros es compleja, por lo que el no hacer bien el trabajo eleva los costos dedicados a esta actividad. A raíz de la complejidad de estas actividades, se ha requerido el surgimiento y desarrollo constante de aplicaciones informáticas. Estos sistemas favorecen el trabajo de los especialistas pero su mayoría son desarrollados por grandes grupos extranjeros que responden a empresas líderes de este sector. El costo de adquisición y el pago de licencias de estos software resultan elevados y en ocasiones poco asequibles.

La Unión Cuba-Petróleo (CUPET), es la entidad cubana encargada de extraer, procesar y comercializar el crudo de producción nacional. Esta se organiza fundamentalmente por tres empresas de Perforación y Extracción de Petróleo (EPEPs), tres entidades de refinación, varias instituciones comerciales y el Centro de Investigaciones del Petróleo (CEINPET). Este último da respuesta de forma integral a toda la actividad petrolera, desde la Exploración hasta la Refinación, en el proceso de investigación – desarrollo – producción.

Entre los productos que actualmente utiliza el CEINPET se destacan el Elemental Analysis (ELAN) y el Interactive Petrophysics (IP) que aunque realizan de forma eficiente el análisis e interpretación petrofísica, implican el pago de costosas licencias, así como del sistema operativo donde son ejecutados. Estos productos propietarios son desarrollados por empresas que se dedican a la investigación del petróleo y dentro de esta al análisis petrofísico, pero en muchas ocasiones se necesita de una experiencia avanzada para su uso. Otro inconveniente es que luego de ser adquiridos no se puede acceder a su código fuente para una posible mejora.

A partir de las características presentadas podemos decir que: para obtener sus actualizaciones se hace necesario realizar elevados pagos constituyendo un obstáculo para la institución y debido al bloqueo en muchas ocasiones estas compras se realizan por terceras personas disminuyendo la probabilidad de obtener un producto fiable y con menores costos. En otro de los casos se demora días y hasta semanas la adquisición de las licencias provocando retrasos en los análisis y tomas de decisiones en el campo donde se perfora o intervienen los pozos de petróleo y gas.

En el Centro de Desarrollo de Informática Industrial (CEDIN) perteneciente a la Universidad de las Ciencias Informáticas (UCI), se desarrollan soluciones para la automatización de procesos industriales. Sus resultados han conllevado a establecer encuentros de trabajo con el CEINPET, donde se demostró que los software utilizados por los especialistas petrofísicos son desarrollados bajo políticas de software privativo, deduciendo un elevado costo en la adquisición de sus licencias. Estos sistemas

en su mayoría priorizan aspectos importantes pero en todo momento ocultan el mecanismo de respuesta, limitando así el conocimiento en el tipo de análisis que se realiza. También se identificó que no se cuenta con un sistema que agrupe todas las funcionalidades necesarias para el trabajo de los petrofísicos y la limitación de métodos de interpretación que se realizan a los pozos petroleros en Cuba en el área de la petrofísica.

A raíz de todo lo anterior el CEDIN conjunto con el CEINPET se dieron a la tarea de desarrollar un sistema que cubra inicialmente las necesidades de los especialistas, dando como resultado el Sistema para el Análisis e Interpretación de Registros de Pozos (AnPer). AnPer es un producto que permite el estudio de las propiedades físicas de las rocas y reservorios de hidrocarburos a través de análisis de muestras de núcleos y de registros de pozo. Dentro de los elementos definidos para los sistemas petrofísicos solamente cuenta con los elementos de análisis, identificándolo como un sistema incompleto y con poco uso para los especialistas. AnPer al no contar con un componente de interpretación, impide realizar una interpretación avanzada y limita a los petrofísicos en la toma de decisiones relacionadas con la adquisición de datos de presión, la recolección de muestras de fluidos y las operaciones de entubación y terminación de pozos.

Ante esta **situación problemática** se define el siguiente problema científico:

¿Cómo garantizar la interpretación petrofísica de registros de pozos con el sistema AnPer?

Enmarcándose como **objeto de estudio** los procesos de interpretación de registros de pozos.

Para dar respuesta al problema de la investigación se definió como **objetivo general** implementar un componente de interpretación de registros de pozos que permita agilizar el tiempo de respuesta ante las solicitudes de interpretación realizadas con AnPer y evaluar las propiedades físicas de las rocas.

El **campo de acción** que abarca esta investigación es el proceso de interpretación de registros de pozos para evaluar las propiedades físicas de las rocas.

Se tiene como **idea a defender** que: Implementando un componente para la interpretación de registros de pozos petroleros que permita la evaluación de las propiedades físicas de las rocas, facilitará y agilizará el tiempo de respuesta ante las solicitudes que se realizan de interpretación.

El componente de interpretación realizado al integrarlo con AnPer, incorporará al mismo la capacidad de realizar la interpretación de registros de pozos. Esta funcionalidad posibilita que AnPer se consolide como un producto completo, permitiendo el análisis e interpretación petrofísica. Todo esto minimiza en tiempo y costo el proceso de interpretación, que antes de existir se realizaba con soluciones propietarias y procedimientos engorrosos, posibilitando que el mismo pueda incorporarse y competir en el mercado.

Para cumplir el objetivo de la investigación se tienen las siguientes **tareas investigativas**:

- Análisis de sistemas de interpretación y análisis petrofísico para un mejor entendimiento del proceso de interpretación y análisis.
- Estudio de métodos existentes que posibilitan el análisis e interpretación de registros de pozos con el objetivo de aplicarlos en la implementación del componente.
- Estudio y selección de los modelos y procedimientos para la implementación del componente.
- Análisis de las tecnologías, lenguajes y herramientas propuestas para el desarrollo del componente.
- Codificación del componente para la interpretación y análisis de registros de Pozos.
- Realización de pruebas al componente para su exitoso funcionamiento.

Se utilizan los **métodos científicos** que se enuncian a continuación:

Métodos Teóricos:

Analítico – sintético: este método sirve para analizar y distinguir los elementos relacionados con análisis e interpretación de registros de pozos, componentes de software y proceder a revisar ordenadamente cada uno de ellos por separado, para posteriormente procesar toda la información, poder sintetizar y diferenciar la información más relevante. Además de poder extraer los elementos más importantes teniendo en cuenta el objeto de estudio.

Histórico – lógico: como está vinculado al conocimiento de las distintas etapas de los objetos en su sucesión cronológica, posibilitó conocer los antecedentes y tendencias de las tecnologías de interpretación y desarrollo de software en el mundo y en Cuba. Mediante este método, se analiza la trayectoria concreta de la teoría, su condicionamiento a los diferentes períodos de la historia y al basarse en el desarrollo histórico, pone de manifiesto la lógica interna de su desarrollo, su teoría y encuentra el conocimiento más profundo de esta, su esencia.

Inductivo-deductivo: se utiliza al realizar un estudio de la información recopilada y al seleccionar la tecnología que se va usar.

Método empírico:

Observación: este método tiene como objetivo obtener un registro visual de toda la información referente a los procesos de interpretación y el funcionamiento del CEINPET en una situación real.

Consulta de expertos: Se utilizó para obtener el criterio de los especialistas para guiar la investigación por el camino correcto.

Este trabajo fue estructurado de la siguiente forma:

Capítulo 1 Fundamentación Teórica: En este capítulo se analiza el estado del arte sobre análisis e interpretación, herramientas para el trabajo con el componente y los conceptos asociados al dominio del problema. Selección de las tecnologías y herramientas actuales a considerar para realizar la selección de aquellas que se van a utilizar en el desarrollo del componente.

Capítulo 2 Descripción y Análisis de la solución: Se propone una solución al problema planteado y se describe el desarrollo de la misma. Se definen y describen los requisitos que debe cumplir, así como los casos de uso del sistema.

Capítulo 3 Implementación y Validación de la solución: Se construye de forma completa la solución propuesta. Se selecciona los estándares de codificación para la implementación del componente. Además de realizar pruebas para comprobar su correcto funcionamiento.

Capítulo 1: Fundamentación Teórica.

1.1.Introducción

Los hidrocarburos están formados por carbono, hidrógeno, oxígeno, nitrógeno y azufre. Del petróleo se fabrica una variedad de productos químicos como la turbosina, que es la gasolina utilizada en aviones jet, la gasolina de aviación para uso de aviones con motores de combustión interna, el diesel de uso común en camiones, ómnibus y el asfalto en la construcción. Es irrefutable la suma importancia que este hidrocarburo tiene a nivel mundial, es por esto que Cuba cuenta con el CEINPET, que se encarga de dar respuesta a toda la actividad petrolera desde la exploración hasta la refinación. En esta institución se realizan varias actividades dentro de las que se destacan el análisis petrofísico de los registros de pozos y reservorios.

El presente capítulo brinda una panorámica sobre el objeto de estudio de la investigación y contempla algunos aspectos básicos relativos de análisis e interpretación de registros pozos de petróleo. Se explican algunos conceptos importantes para un mejor entendimiento de este trabajo. Además se caracteriza de manera breve algunos software utilizadas en el mundo y en Cuba utilizados por el CEINPET para realizar el proceso de análisis e interpretación de registros de pozos. Igualmente se mencionan las tecnologías, herramientas y técnica de programación que se van a usar para desarrollar el componente.

1.2.Conceptos asociados al dominio del problema

Petrofísica: La petrofísica se ocupa del estudio de las propiedades físicas de las rocas como reservorios de hidrocarburos a través de análisis de laboratorio en núcleos y de registros de pozo, es decir, es el estudio de las propiedades físicas y químicas que describen la incidencia y el comportamiento de las rocas, los sólidos y los fluidos (1). Por tanto, la petrofísica es la encargada del estudio de todas las propiedades físicas y químicas de las rocas y describe el comportamiento de los fluidos presentes en las mismas.

Pozo petrolero: Perforación efectuada por medio de barrenas de diferentes diámetros y a diversas profundidades, con el propósito de definir las condiciones geológico-estructurales de la corteza terrestre, para la prospección o explotación de yacimientos petrolíferos. El método más utilizado es el rotatorio, y las perforaciones pueden desarrollarse con o sin recuperación de núcleo (2).

Yacimiento: Los yacimientos son sistemas complejos y heterogéneos compuestos por diversos tipos de rocas depositadas en diferentes ambientes sedimentarios y afectados por una infinidad de procesos geológicos a lo largo de su historia (1).

Reservorio: Son conocidos como roca almacén o roca colectora ya que en esta roca es donde está almacenado el petróleo.

Registro de pozos: Son mediciones que se realizan con sensores remotos constituidos por buzo, cable y registrador de superficie. Estos registros se relacionan con las propiedades físicas, mecánicas, eléctricas y radiactivas de las secuencias de rocas que descubre un pozo (3).

Proporcionan información *in situ*¹ del subsuelo, que difícilmente pueden ser obtenidas a través de otros métodos menos costosos. Pueden definirse como mediciones de diversos parámetros y propiedades físicas de un pozo, tomadas a lo largo del mismo y bajo ciertos intervalos. Consisten en una serie de mediciones o registros geofísicos, obtenidas por una sonda² con varios sensores o antenas transmisoras y receptoras que se introduce en una perforación de barreno para determinar las curvas de cada parámetro que se desea conocer. Con esta técnica se obtiene a diferentes profundidades los parámetros físicos de la formación. Se lleva a cabo para determinar las características físicas de las rocas, de los fluidos que la saturan y de las propiedades de la construcción del pozo. Un Registro de pozos no es más que una forma de llevar un control o agrupar las mediciones hechas a un pozo determinado, de sus propiedades físicas (4).

Evaluación de Formaciones en pozos de petróleo y gas: Es toda la evaluación que se hace de los reservorios y sellos en un contexto geológico, donde se integra toda la información sobre las rocas que constituyen las distintas formaciones.

Porosidad: La porosidad se define como la relación entre el volumen de espacio de poro y el volumen de grano de la roca del yacimiento. Puede ser expresado en fracción o por ciento. Desde el punto de vista del proceso responsable de la formación de la porosidad se clasifica fundamentalmente en dos tipos Primaria (es la porosidad original después de la deposición de los sedimentos y su compactación

¹ Son pruebas realizadas para la determinación de las características geotécnicas de un terreno.

² Es un objeto de manipulación remota cuya misión es llegar a un objetivo prefijado y realizar algún tipo de acción o mandar información.

inicial) y Secundaria (debida a los esfuerzos tectónicos y a la circulación del agua subterránea. Es más importante en rocas carbonatadas debido a su fragilidad y relativa solubilidad) (1). Es una medida de la capacidad de almacenamiento de fluidos que posee una roca y se define como la fracción del volumen total de la roca que corresponde a espacios que pueden almacenar fluidos (3).

Permeabilidad: Es la medición de la capacidad de una roca para transmitir fluidos. Para que sea permeable, la roca debe tener poros o fracturas interconectados, por lo tanto, hay una relación general entre la porosidad y la permeabilidad. Según el Glosario de las reservas de hidrocarburos de México se define como permeabilidad: “facilidad de una roca para dejar pasar fluidos a través de ella. Es un factor que indica si un yacimiento es, o no, de buenas características productoras” (3).

Saturación de agua: Puede ayudar a determinar la probabilidad de que haya hidrocarburos en una formación rocosa y el volumen existente de hidrocarburos en un depósito determinado, es decir, el cálculo de las reservas (1).

Flujo: El flujo de fluidos tiene lugar en una red interconectada de espacios de poro. Las características y las propiedades de esta red porosa están relacionadas con la distribución original del tamaño de los granos de la roca del yacimiento. La comprensión de la estrecha relación existente entre la red de poros, las propiedades de la roca y el flujo constituye la piedra angular de cualquier estudio de un yacimiento (1).

Componente de software. Un componente es una parte no trivial, casi independiente, y reemplazable de un sistema que llena claramente una funcionalidad dentro de un contexto en una arquitectura bien definida. Un componente se conforma y provee la realización física por medio de un conjunto de interfaces.

Un componente de software en tiempo de ejecución es un paquete dinámicamente vinculado con uno o varios programas manejados como una unidad y que son accedidos mediante interfaces bien documentadas que pueden ser descubiertos en tiempo de ejecución.

Un componente de software es una unidad de composición con interfaces contractualmente especificadas y explícitas sólo con dependencias dentro de un contexto. Un componente de software puede ser desplegado independientemente y es sujeto a la composición de terceros (5).

1.3. Objeto de estudio

Durante la perforación de los pozos petroleros, tanto en la etapa de exploración como en la parte de perforación, el proceso de interpretación y análisis petrofísico a partir de la lectura de los registros de pozos es importante para los especialistas del CEINPET. A partir de esta lectura se obtienen propiedades físicas de las rocas tales como: la porosidad, la saturación de agua y la permeabilidad. Todo esto permite que se realice de forma eficiente la exploración y la posterior explotación de los yacimientos de hidrocarburos.

En los últimos 20 años, las nuevas mediciones obtenidas de los registros han mejorado la interpretación. Estas mejoras pueden dividirse en dos tipos; las que se concentran en una nueva definición de los fluidos y las enfocadas en una mejor definición de los sólidos. La definición directa de los fluidos ha mejorado sustancialmente gracias a los desarrollos logrados en las herramientas de adquisición de registros de resonancia magnética nuclear (RMN). Dado que las principales propiedades de interés son porosidad, saturación y permeabilidad se relacionan con los fluidos, esta técnica RMN parecería la mejor opción. Sin embargo, existen limitaciones, particularmente con las interpretaciones de los valores de saturación porque las mediciones se obtienen en la zona invadida, cerca del pozo, y porque las señales indicadoras de la presencia de petróleo y agua a veces no están claramente separadas.

La más utilizada es la que consiste en definir los volúmenes de sólidos y luego aplicar las ecuaciones conocidas para determinar las propiedades principales de los yacimientos a partir de otras mediciones. El proceso de definición de los sólidos se basa en que cada medición posee una ecuación de respuesta que puede ser expresada como un conjunto de volúmenes de formación desconocidos, cada uno de los cuales es multiplicado por un parámetro. Cuando existen como mínimo tantas ecuaciones como volúmenes desconocidos, estos últimos pueden hallarse mediante métodos de inversión comunes. La solución puede ser restringida; por ejemplo, no permitiendo que la porosidad exceda un valor específico.

Cualquiera que sea el método seleccionado, las tareas principales de la interpretación asistida por las computadoras siguen siendo las mismas. En primer lugar, los registros de entrada necesarios son editados, ajustados en profundidad y corregidos por efectos ambientales. En segundo lugar, se deben elegir los parámetros y cuando se dispone la opción, el modelo de formación que no son más que los minerales, rocas y fluidos que probablemente se puedan encontrar en cantidades apreciables. Por

último, es preciso controlar los resultados para verificar su calidad y modificar los parámetros o el modelo hasta que la interpretación resulte satisfactoria (6).

1.3.1. Descripción general

Todo el proceso de análisis petrofísico se realiza a través de estudios en laboratorios de los núcleos y registros de pozo. La información del estudio de las condiciones de los pozos de petróleo a diferentes profundidades es guardada en los registros o ficheros con diferentes extensiones (.las, .lis, .dlis)³ de pozos. Al inicio todo el trabajo fue rústico, se utilizaban cables y sondas. En la actualidad el trabajo de forma manual con los registros de pozos es casi imposible porque el volumen de datos recuperados sobre la información petrofísica es elevado y el procesamiento se vuelve engorroso si no se cuenta con un software para ello.

La información que contienen los registros de pozos es interpretada mediante productos de software especializados, que permiten la visualización de las curvas que representan las mediciones contenidas en los registros y el cálculo de las propiedades físicas. La interpretación de registros es el proceso por el cual los parámetros⁴ que pueden ser medibles se traducen a los parámetros petrofísicos deseados de porosidad, saturación de hidrocarburos, permeabilidad, litología, etc. Se realiza a través de relaciones matemáticas que involucran las mediciones de los registros y los análisis en laboratorios de los núcleos. Además permite evaluar donde es que se encuentran las rocas que pueden producir petróleo y gas. La meta de la interpretación cuantitativa de registros es proporcionar las ecuaciones y técnicas para que dichos cálculos puedan llevarse a cabo. En realidad, las premisas básicas de la interpretación de registros son pocas y de concepto sencillo (7).

Para llevar a cabo la interpretación es necesario tener en cuenta los siguientes pasos:

1. Estimar un conjunto de repuestas que corresponden a los parámetros de la formación que se van a determinar. La definición de este conjunto de repuestas depende del modelo de interpretación seleccionado por el analista de registros.

³ **Log ASCII Standard** (.las), **Log information Standard** (.lis), **Digital log information Standard** (.dlis): formatos de archivos estándar común en el petróleo y el gas de la industria petrolera para almacenar la información de los registros de pozo.

⁴ Los parámetros pueden ser la resistividad, la densidad, el tiempo de tránsito, el potencial espontáneo, la radioactividad natural y el contenido de hidrógeno de la roca.

2. Los registros "reconstruidos" se derivan de este conjunto de respuestas por medio de ecuaciones de respuestas del software; se emplea una para cada registro.
3. Los registros reconstruidos se comparan con los registros reales y una función de "incoherencia" mide la calidad de la concordancia entre los dos conjuntos de registros (6).

1.3.2. Descripción actual del dominio del problema

Actualmente en el CEINPET se realiza el proceso de análisis e interpretación petrofísica a través de software propietarios. Dicha entidad no cuenta con un software libre para interpretar y analizar los registros de pozos. Estos sistemas informáticos al ser propietarios y al no cumplir con las necesidades de los especialistas, se encuentran enfrascados en la búsqueda de soluciones que les facilite y agilice el trabajo.

Actualmente, en colaboración de la Universidad de las Ciencias Informáticas y el Centro de Investigaciones del Petróleo, se está desarrollando el software AnPer el cual pretende agrupar la mayoría de las funcionalidades que presentan estas aplicaciones. Pero en estos momentos solo cuenta con los elementos de análisis, ubicándolo como un sistema incompleto y sin poder competitivo alguno con demás sistemas de este tipo en el mercado. La otra funcionalidad que debe poseer AnPer, es la interpretación de registros, pues es uno de los elementos fundamentales con que debe contar un software petrofísico. Actualmente este software no posee un componente para realizar dicha interpretación e impide que los petrofísicos puedan tomar decisiones de gran envergadura con respecto a los pozos de petróleo y gas del territorio cubano, basándose solo en los resultados mostrados hoy día con el sistema AnPer.

1.4. Modelo de Interpretación

El proceso de interpretación incluye un conjunto de modelos que permiten representar la información y describir el problema para ser solucionado. Estos modelos constan de un conjunto de ecuaciones de respuesta (curvas), un set de componentes de formación o volúmenes (rocas, minerales, fluidos), parámetros (matrices) y restricciones.

1. Los componentes de la formación son los constituyentes de los cuales se necesitan los volúmenes.
2. Las ecuaciones de respuesta son las ecuaciones a resolver y sus datos de entrada asociados.
3. Los parámetros son los parámetros globales y los parámetros de control programados, los parámetros de respuesta y los parámetros obligatorios.

4. Las restricciones son los límites que los volúmenes resultantes deben cumplir.

Además implícitamente contienen curvas asociadas, parámetros de respuesta, y otros parámetros globales y específicos de cada modelo. Para establecer los modelos petrofísicos es necesario tener en cuenta los registros o perfiles y todas las investigaciones que se realizan en los pozos con los cuales se esté trabajando (27).

1.4.1. Descripción de las cuatro partes de un Modelo

En el desarrollo del componente solo se van a utilizar las tres primeras partes (ecuaciones, componentes de formación, parámetros) de un modelo. Para comprender mejor lo expresado anteriormente, a continuación se describen cada una de las tres partes de un modelo

1.4.1.1. Componentes de la formación

Los componentes de formación está formado por tres grupos: minerales, rocas y fluidos.

- Los minerales son sólidos descritos por una fórmula química; por ejemplo SiO_2 , CaCO_3 ó CaSO_4 . Debido a su estructura bien definida, es posible usualmente utilizar los parámetros de los minerales por defecto.
- Las rocas se consideran una combinación de minerales definida por el usuario como cieno, carbonatos y rocas ígneas. Las rocas no tienen parámetros por defecto.
- Los fluidos son las sustancias que rellenan el espacio vacío de huecos en una roca que puede contener agua, petróleo, gas y otros fluidos especiales. Frecuentemente es posible obtener valores por defecto de los fluidos, resultando unos mejores que otros. La porosidad por defecto de 1 para el neutrón es exacta para el agua, pero la de 0.8 g/cc para la zona lavada puede variar notablemente de un pozo a otro (27).

Los componentes de formación seleccionados para integrar el modelo deben ser solamente aquellos que se esperen estar presentes en una cantidad apreciable. El número de componentes a modelar nunca debe exceder al número total de ecuaciones en uso, en caso contrario no podría llevarse a cabo la interpretación puesto que no se tienen los datos necesario para determinar los valores de los componentes de formación.

1.4.1.2. Ecuación de Respuesta

Las ecuación de respuesta es una descripción matemática de cómo una medición obtenida varía respecto a cada componente de la formación. Las ecuaciones más simples de respuesta lineal son de la forma:

$$measurement = \sum_{i=1}^{nfc} V_i \times R_i$$

Figura 1 Ecuación simple de respuesta lineal.⁵

Aunque algunas ecuaciones lineales incluyen términos adicionales, y las ecuaciones no lineales son más complejas, el concepto general es el mismo: las mediciones totales observadas están determinadas por el volumen de cada componente de formación, y cómo ecuación reacciona frente a ese componente de la formación (27). En cualquier solución simultánea de un sistema de ecuaciones, siempre debe haber al menos tantas ecuaciones como incógnitas. Se utilizará para la solución de ecuaciones en el componente, los sistemas de ecuaciones lineales determinados.

- **Determinado**

Si hay exactamente tantas ecuaciones independientes como incógnitas, el sistema se dice que es determinista. Un sistema determinado es exactamente un conjunto de valores donde las incógnitas satisfacen las ecuaciones (27).

- **Sobre determinado**

Si hay más ecuaciones que incógnitas, el sistema está sobre determinado, por lo que se utiliza un método de ajuste como el de mínimos cuadrados (27).

- **Indeterminado**

⁵ donde:

V_i = volumen del componente de formación i .

R_i = parámetro de respuesta para el componente de formación i .

Si hay menos ecuaciones que incógnitas, el sistema es indeterminado y no puede ser resuelto hasta que el problema se reorganice mediante la adición de ecuaciones independientes o reduciendo el número de incógnitas (27).

1.4.1.3. Parámetros

Los parámetros darán el control sobre como el programa se comporta, qué resultados se producen y que datos de utilizan. Los parámetros de control global controlan en qué dirección del programa se llevará a determinadas ramas principales de la lógica. Usted puede pensar en ellos como para determinar en qué modo el programa funcionará. Estos adquieren un valor único en todo el intervalo de procesamiento y los parámetros obligatorios son aquellos que no deben cambiar para no afectar el modelo.

1.5. Métodos a utilizar en la Interpretación Petrofísica

Para la interpretación petrofísica existen diferentes tipos de métodos que se pueden utilizar para evaluar las propiedades físicas de las rocas. En cada uno de ellos existe un grupo de factores que influyen en las curvas o informan sobre las secuencias de rocas, entre estos métodos se encuentran:

Los Métodos Auxiliares:

- **Mecánicos** (*Cáliper, Velocidad de Perforación, Inclínometría*): miden el diámetro del pozo, dan información del tipo de roca y puede obtenerse el ángulo de inclinación de los pozos.
- **Químicos** (*Carotaje Gaseoso, Calcimetría*): se miden los gases que van fluyendo de las rocas y el contenido de carbonatos (calcita y dolomita) que las componen.
- **Termometría**: conocer la temperatura de la formación a cada profundidad (9).

Los Métodos Convencionales:

- **Métodos Eléctricos** (*Potencial Espontáneo, Resistividad, Conductividad*): miden las propiedades eléctricas de las rocas, es decir el potencial eléctrico (voltaje) producido por la interacción del agua de formación innata, el fluido de perforación conductivo y las arcillas, por ser selectivas a los iones. El Efecto Fotoeléctrico que consiste en la emisión de electrones por un material cuando se le ilumina con radiación electromagnética (luz visible o ultravioleta, en general), en este caso la luz de los rayos gamma.

- **Métodos Radiactivos** (*Rayos Gamma, Neutrónicos, Lito Densidad*): mediante estos se pueden indicar la radiactividad natural de las formaciones, obtener información sobre el contenido de hidrógeno, relacionado con la porosidad, presencia de gas en las formaciones y contenido de arcilla en las rocas. También obtener mediciones de la densidad total de las rocas y el efecto fotoeléctrico.
- **Métodos Sónicos (Incluye el VSP)**: se puede obtener información sobre la porosidad de la matriz de las rocas y la porosidad secundaria de las rocas (9).

Los Métodos Especiales:

- **Registros de Imágenes** (*FMS y FMI*): mediante ellos se pueden detectar anomalías estructurales, análisis de unidades sedimentarias y estudio de ambientes de deposición.
- **Registros Magnéticos Nucleares** (*NMR*).
- **Registro Sónico** (*DSI*) (9).

Los métodos de exploración cambian el resultado de la interpretación de registros, es decir, en dependencia de los métodos que se seleccionen para realizar la interpretación se pueden obtener diferentes resultados. Cada uno tiene una ecuación de respuesta diferente respecto a cada volumen.

1.6. Análisis de los sistemas utilizados en el mundo y en Cuba

Hace más de medio siglo se introdujo el registro eléctrico de pozos en la industria petrolera. Desde entonces, se han desarrollado y utilizado, en forma general, mucho más y mejores dispositivos de registro. A medida que la ciencia de los registros de pozos petroleros avanzaba, también lo hacía el arte de la interpretación de datos. Hoy día, el análisis detallado de un conjunto de perfiles cuidadosamente elegidos, provee un método para derivar e inferir valores precisos para las saturaciones de hidrocarburos y de agua, la porosidad, el índice de permeabilidad, y la litología de las rocas del yacimiento (6).

En la actualidad existen sistemas que agilizan el proceso de interpretación y análisis de registros de pozos, entre los que se destacan:

1. Sistema para el Análisis e Interpretación de Registros de Pozo (AnPer).
2. Petroluke.
3. Elemental Analysis(ELAN)

4. InteractivePetrophysics.
5. NeuraLog.
6. PetroLog.

1.6.1. AnPer

El Proyecto Productivo Sistema para el Análisis e Interpretación de Registros de Pozo (AnPer) de manera general, tiene como propósito obtener un sistema cubano que permita a los trabajadores del CEINPET realizar el análisis petrofísico de los registros de pozos extraídos de los yacimientos petroleros.

Con el desarrollo de la aplicación se garantizará el desarrollo de actividades relacionadas con los procesos de análisis petrofísico, permitiendo realizar: la salva, importación y exportación de los estándares de registros de pozos (.LAS, .LIS, .TXT), creación de proyectos de prueba que permiten realizar estudios de casos para cada entorno identificado. Interpretar ficheros de registro de pozos y representar la información en diferentes tipos de gráficas 2D y 3D (graficas de curvas, histogramas, cossplot). Se garantizará una eficiente y rápida toma de decisiones de un pozo o de un área determinada, permitiendo realizar comparaciones a partir de datos históricos y se podrá realizar los análisis correlacionales entre varios registros existentes. Actualmente solo cuenta con los elementos de análisis, ausentándose el componente de interpretación para realizar el análisis del comportamiento de una determinada propiedad de un pozo (Porosidad, Permeabilidad y Saturación) a partir de modelos y métodos científicos definidos para conocer la calidad del reservorio que se estudia.

1.6.2. Petroluke

PETROLUKE es capaz de cargar archivos en formatos LAS y ASCII de forma eficiente y correcta, interpretarlos, guardarlos en archivos MAT y luego abrirlo una vez más para continuar la interpretación, lo cual constituye una ventaja para el intérprete y facilita este proceso. Este software cuenta con diferentes módulos, como el Editor de Ecuaciones, que permite de forma versátil y amigable, la mayor parte de los cálculos necesarios para realizar una interpretación petrofísica avanzada. Por otra parte, pese a no ser aún capaz de calcular resistividades del agua de formación, el módulo de Crossplots demostró ser útil para discriminar litología y la posible presencia o ausencia de Gas (10).

1.6.3. Elemental Analysis (ELAN)

El ELAN⁶ es un programa de análisis de registros multi-mineral (litología compleja) capaz de calcular la formación más probable de minerales y volúmenes de fluido de poros. Es en particular valioso en áreas de litología variada, minerales especiales y sistemas de porosidad duales (10). ELANPlus es la versión en ELAN GeoFrame. ELANPlus proporciona un nivel por nivel multi-mineral (litología compleja), evaluación de la formación cuantitativa con una resolución de ecuaciones simultáneas y el modelo optimizado, método de combinación. Resuelve un conjunto de ecuaciones (formado por los registros, la formación de volumen de los componentes y parámetros) para obtener el volumen de cada uno de los componentes de la primera formación. Calcula las propiedades de formación (tales como la porosidad, la saturación de agua, el volumen de la pizarra) de los volúmenes derivados, no calcula las propiedades de formación de una serie de fórmulas fijas paso a paso como PetroViewPlus (10).

1.6.4. Interactive Petrophysics (IP)

Interactive Petrophysics es un software para el análisis de las propiedades y depósitos de petróleo. Es ideal tanto para petrofísicos como para geólogos porque permite flujos de trabajo sencillos tales como: la calidad del control de los datos de registros para realizar tareas complejas de varias zonas y varios análisis. Usa los modelos deterministas y probabilísticos para calcular la porosidad, la saturación de agua, los volúmenes de sales y otras propiedades en las zonas definidas por el usuario. Los usuarios pueden elegir los parámetros y criterios de valoración como crossplot directamente en las parcelas. Como parámetros se seleccionan de las parcelas de registro, crossplots interactivos e histogramas, los resultados de su análisis son actualizado al instante. Interactive Petrophysics posibilita la interpretación en tiempo real, permitiendo la conectividad para leer datos de una red de Transferencia de Información del Pozo en el Lenguaje de Mercado Estándar (con sus siglas en ingles WITSML). Y con las pantallas 3D y utilidad de Generador de informes, la presentación profesional de sus resultados (10).

1.6.5. Neurolog

El software Neuralog es usado por la comunidad Geo científica cuando los requerimientos de calidad y el tiempo son críticos. Cuenta con nuevos sistemas de digitalización automatizada que hacen la captura de datos fáciles, rápidos y de inmejorable calidad. Acepta imágenes TIFF a color, escala de grises o blanco y negro. Igualmente permite la importación de archivos formato LAS (11).

⁶ Traducción: Análisis Elemental de Registros.

Neuralog incluye las siguientes funcionalidades especiales para garantizar la captura de datos:

- ✓ Definiciones de varios tracks y curvas.
- ✓ Corrección de deformación de imágenes (estiramientos, encogimientos, etc). Soporta la vectorización de registros de lodos y Dipmeters. Soporta la vectorización de escalas lineales, logarítmicas y multilineales. Función para el copiado y pegado de curvas. Cálculo automático entre las curvas de resistividad y conductividad. Opción para el manejo de fuera de escalas.
- ✓ Vectorización de curvas a color.
- ✓ Generador de estadísticas de archivos LAS.
- ✓ Herramientas para la verificación de la calidad de vectorización (11).

Este software al ser propietario cuenta con una llave única para su uso. Estos restringen su utilización solamente a una computadora a la vez, dejando al resto del personal inactivo. Solamente puede ser usado en varias computadoras en dependencia de la cantidad de licencias que la empresa cuente del software en cuestión, que tienen por característica precios elevadísimos y sin la posibilidad de actualizar sus servicios.

Luego del análisis de los software anteriores, se concluye que muchos podrían en cierta medida satisfacer y resolver la problemática actual, sin embargo debido a la situación económica en la que se encuentra el país no se puede hacer uso pleno de los mismos. El desarrollo de estos sistemas es realizado por empresas privadas constituyendo una desventaja, pues debido al bloqueo se dificulta e imposibilita el acceso a ellos, a lo cual también se suma el elevado precio de licencias y actualizaciones.

Con la integración de un módulo de interpretación al sistema que se está desarrollando en la UCI (AnPer) se reducirían los gastos de la industria cubana del petróleo por concepto de importaciones de software especializados. Al estar desarrollado con tecnología de software libre garantiza dependencia tecnológica. La inclusión de estándares internacionales y resultados de investigaciones desarrolladas en las entidades clientes hace al producto una herramienta más creíble y confiable; pero al no contar con las funcionalidades de interpretación lo aleja del mercado y lo refleja como un sistema parcial. Por todas las razones explicadas anteriormente se propone el desarrollo de un componente de análisis e interpretación registros de pozos para el Sistema AnPer.

1.7. Tendencias y tecnologías actuales a utilizar

Las tecnologías están presentes en todos los procesos de la sociedad y de la economía del ser humano. Su principal propósito es transformar el entorno, para adaptarlo mejor a las necesidades y deseos del hombre. Las tecnologías de la Información y Comunicación (TIC) son aquellas herramientas computacionales e informáticas que procesan, almacenan, sintetizan, recuperan y presentan información representada de la más variada forma.

1.7.1. Metodología de desarrollo: Proceso Unificado de Desarrollo (RUP)

Hoy en día existe una gran variedad de estudios y estándares que exponen los principios que se deben seguir para la mejora de los procesos de software, con diferentes metodologías compuestas por filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación sumamente importantes. Un proceso proporciona normas para el desarrollo eficiente de software de calidad. Captura y presenta las mejores prácticas que el estado actual de la tecnología permite. En consecuencia reduce el riesgo y hace el proyecto más predecible (13).

RUP como metodología para el proceso de desarrollo de software se caracteriza por ser iterativo e incremental, centrada en la arquitectura y dirigida por los casos de uso. En cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software. Utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema de software (13).

RUP es una metodología que divide el desarrollo de software en 4 fases: la fase de inicio, en la que se determina la visión del proyecto; la de elaboración, en la que se determina la arquitectura óptima del software; la de construcción, en la que se obtiene una capacidad operacional inicial; y la de transición, donde se obtiene un reléase del producto. Cada una de las etapas mencionadas es desarrollada mediante un ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes (13).

En cada una de las iteraciones se llevan a cabo los siguientes flujos de trabajo:

- ✓ Modelo de Negocio: describe los procesos de negocio de una empresa en términos de casos de uso y actores.
- ✓ Requerimientos: descripción de condiciones o capacidades que el sistema debe de cumplir.

- ✓ Análisis y Diseño: se trasladan los requerimientos dentro de la arquitectura de software.
- ✓ Implementación: crea el software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- ✓ Pruebas: asegura que el comportamiento requerido es el correcto y que todo lo solicitado está presente (13).

El uso de RUP como metodología de desarrollo permitió guiar el proceso de desarrollo de AnPer1.0. Las razones de selección estuvieron dadas a partir de las características que ella presenta. También por las necesidades de generar la mayor cantidad de documentación por lo cambiante que pueden ser los equipos de trabajo y por la costosa relación que existe entre sus miembros, reflejándose en los diferentes puntos de trabajo. Además de que los integrantes del proyecto AnPer se encuentran familiarizados con esta metodología, lo que permite agilizar el proceso de desarrollo del componente.

1.7.2. Lenguaje Unificado de Modelado, UML:

UML (Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema de software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE (14).

UML define una notación que se expresa como diagramas que sirven para representar modelos/subsistemas o parte de ellos, define una estructura para ir del análisis al diseño y de este a la implementación (15).

El objetivo de utilizarlo es capturar las partes esenciales del sistema a desarrollar. Su uso permite combinar diversos elementos gráficos mediante reglas para finalmente conformar diagramas y se logra un modelado visual independiente del lenguaje de implementación, de tal forma que los diseños realizados usando este lenguaje se puede implementar en cualquier lenguaje de programación que soporte las posibilidades de UML (principalmente lenguajes orientados a objetos).

Es además un método formal de modelado que aporta las siguientes ventajas:

- ✓ Mayor rigor en la especificación.
- ✓ Permite realizar una verificación y validación del modelado realizado.

- ✓ Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos).
- ✓ Posibilita que el modelo y el código estén actualizados, con el que siempre se puede mantener la visión en el diseño de mas alto nivel de la estructura del proyecto.
- ✓ Proporciona un vocabulario y reglas para permitir la comunicación y se centra en la representación gráfica del sistema o sea con los diagramas se obtiene la representación de diversas perspectivas del sistema, a las cuales se le conoce como modelo. Se especifican cuales son sus características antes de su construcción y los propios elementos gráficos sirven como documentación pudiendo utilizarse para su futura versión (16).

En general se puede decir que un modelo UML describe lo que supuestamente hará un sistema pero no dice como implementarlo. Se escogió como lenguaje de modelado UML ("Unified Modeling Language") porque es posible establecer una serie de requerimientos y estructuras necesarias para modelar un sistema de software previo al proceso intensivo de escribir código. UML es un lenguaje que posee más características visuales que programáticas, las que facilitan a integrantes de un equipo participar e intercomunicarse fácilmente, siendo estos integrantes los analistas, diseñadores, especialistas de área y desde luego los programadores. Una de las características más importantes que hacen que se escoja UML como lenguaje de modelado es que está diseñado para uso con software orientado a objetos, y tiene un uso limitado en otro tipo de cuestiones de programación.

1.7.3. Herramienta Case: Visual Paradigm

Las herramientas CASE de modelado con UML nos brindan la posibilidad de aplicar una metodología de análisis y diseño orientados a objetos y abstraernos del código fuente, de manera que la arquitectura y el diseño se tornan más obvios y más fáciles de entender y modificar. Cuanto más grande y complejo es un proyecto, resulta más importante utilizar una herramienta CASE ya que aporta grandes beneficios para todos los involucrados en un proyecto, por ejemplo, jefe del proyecto, analistas, arquitectos, desarrolladores y otros. Tienen como objetivo fundamental automatizar los aspectos clave en el proceso de desarrollo (13).

Dentro de las características más importantes de las herramientas CASE están:

- ✓ Alto rendimiento.
- ✓ Calidad de software desarrollado.
- ✓ Archivos de datos de conocimiento.

- ✓ Facilidades de uso. (Herramientas CASE y RAD11, 2006)

El Visual Paradigm constituye una herramienta CASE y está diseñada para Análisis y Diseño, utiliza el Lenguaje Unificado de Modelado (UML). Tiene disponibilidad para disímiles versiones y para integrarse en múltiples plataformas. Permite que se genere código en varios lenguajes, entre ellos C++.

Licencia: Gratuita y Comercial.

Características:

- ✓ Producto de calidad.
- ✓ Soporta aplicaciones web.
- ✓ Las imágenes y reportes generados, no son de muy buena calidad.
- ✓ Varios idiomas.
- ✓ Generación de código para Java y exportación como HTML.
- ✓ Fácil de instalar y actualizar.
- ✓ Compatibilidad entre ediciones.

Además Visual Paradigm ofrece:

- ✓ Entorno de creación de diagramas para UML 2.0
- ✓ Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.
- ✓ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✓ Capacidades de ingeniería directa (versión profesional) e inversa.
- ✓ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✓ Disponibilidad de múltiples versiones, para cada necesidad.
- ✓ Disponibilidad de integrarse en los principales IDEs.
- ✓ Disponibilidad en múltiples plataformas (15).

Esta herramienta permite la documentación de todo el ciclo de vida del componente siguiendo la metodología seleccionada, RUP en este caso, haciendo uso de UML como lenguaje de modelado. Ofrece facilidades en cuanto al trabajo con las mismas. Se eligió como herramienta Case a usar el Visual Paradigm en su versión 6.4, definido anteriormente por el arquitecto del proyecto. Aunque necesite de requerimientos computacionales, mediante el mismo se puede generar un producto de mayor calidad, el diseño se centra en los casos de uso y se enfoca en el negocio. Hay que señalar su robustez, usabilidad y portabilidad. Está diseñada para dar soporte a arquitectos de sistemas, diseñadores, desarrolladores, analistas de procesos de negocio y modeladores de datos en los

procesos de desarrollo de software. La Universidad de las Ciencias informáticas enseña la asignatura de Ingeniería de Software mediante el empleo de esta herramienta luego de haber pagado la licencia de la misma.

1.7.4. Framework de Desarrollo: QT

Qt es un framework⁷ de desarrollo para aplicaciones multiplataforma que simplifica mucho el desarrollo de aplicaciones en C++ de forma nativa, también puede ser utilizado en otros lenguajes, funciona en las principales plataformas y tiene un amplio apoyo. Es una biblioteca para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas sin interfaz gráfica como herramientas de consola y servidores (33).

Este framework está compuesto por librerías Qt (clases en C++), además se pueden crear formularios visualmente con Qt Designer y presenta un acceso rápido a la documentación a través de Qt Assistant. Permite una traducción rápida con su Qt Linguist y simplifica el proceso de construcción de proyectos en las diferentes plataformas soportadas. La Interfaz de Programación de Aplicaciones(API) de la biblioteca cuenta con métodos para acceder a bases de datos mediante SQL, así como uso de XML; cuenta con una API multiplataforma unificada para la manipulación de archivos y otras para el manejo de ficheros, además de estructuras de datos tradicionales (33).

Otras Características:

- ✓ QT es una librería para la creación de interfaces gráficas. Se distribuye bajo una licencia libre GPL (o [QPL](#)), además de la LGPL, que nos permite incorporar las QT en nuestras aplicaciones open-source.
- ✓ Se encuentra disponible para una gran número de plataformas: Linux, MacOS X, Solaris, HP-UX, UNIX con X11, Windows.
- ✓ Es orientado a objetos y el lenguaje para el que se encuentra disponible es C++.
- ✓ Es una biblioteca que se basa en los conceptos de widgets (objetos), Señales-Slots y Eventos (Ej.: clic del ratón).
- ✓ Las señales y los slots es el mecanismo para que unos widgets se comuniquen con otros.
- ✓ Los widgets pueden contener cualquier número de hijos. El widget superior (también conocido como top-level), puede ser cualquiera, sea ventana, botón, etc.

⁷ Framework: en el desarrollo de software es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

- ✓ Compatibilidad multiplataforma con un sólo código fuente.
- ✓ Fácil de internacionalizar.
- ✓ Arquitectura lista para plugins.

Con todo lo anterior expuesto se puede decir que Qt dispone de tres grandes ventajas ante otras librerías rivales:

- Qt es completamente gratuito para aplicaciones de código abierto, y una de las licencias mencionadas anteriormente, la LGPL, permite que se utilice gratuitamente con fines comerciales.
- Las herramientas, librerías y clases están disponibles para casi todas las plataformas Unix y sus derivados (como Linux, MacOS X, Solaris, etc.) como también para la familia Windows, por lo que una aplicación puede ser compilada y utilizada en cualquier plataforma sin necesidad de cambiar el código y la aplicación se verá y actuará mejor que una aplicación nativa.
- Qt tiene una extensa librería con clases y herramientas para la creación de aplicaciones con interfaz de usuario enriquecidas. Estas librerías y clases están bien documentadas, son muy fáciles de usar y tienen una gran herencia de programación orientada a objetos lo cual hace de la programación de interfaces gráficas un trabajo más ameno.

A continuación otras de las características más importantes de este framework que ayudan a la selección del mismo:

- ✓ Compatibilidad multiplataforma con un sólo código fuente.
- ✓ Rendimiento en C++.
- ✓ Disponibilidad del código fuente.
- ✓ Excelente documentación.
- ✓ Arquitectura lista para plugins.

Con este framework se pueden crear de forma rápida y sencilla interfaces gráficas de usuario, provee componentes y librerías como **QtSql** que facilitan la integración con base de datos usando SQL y **QtXml** permitiendo guardar la información como extensión xml. Además tiene disponible como lenguaje C++, siendo este el lenguaje seleccionado para la implementación del componente. También el uso de esta biblioteca ahorra una enorme cantidad de tiempo y esfuerzo, pues de lo contrario se tendrían que desarrollar algunos componentes desde cero. Estas son las principales características que debe tener la aplicación que se desarrolla, lo cual dio paso a la selección del mismo para la realización de la aplicación de este trabajo.

1.7.5. Lenguaje de Programación: C++

Ante la presencia de tantos lenguajes, ¿en qué se basa un ingeniero de software para adoptar en un proyecto el uso de uno en concreto? Generalmente, el lenguaje se elige por preferencia personal de los clientes de una empresa, o bien, porque los desarrolladores poseen un conocimiento del mismo, lo cual es razonable. En este caso el arquitecto en conjunto con el equipo de desarrollo ha seleccionado como lenguaje de programación a C++ y de esta manera lograr alto nivel de eficiencia.

Es un lenguaje de programación orientado a objetos que toma la base del lenguaje C, abarca tres paradigmas de programación: la programación estructurada, la programación genérica y la programación orientada a objetos por lo que se considera un lenguaje híbrido multiparadigma⁸ (18).

C++ es un lenguaje de programación maduro y de gran velocidad de compilación y presenta las siguientes características:

- ✓ Lenguaje versátil, potente y general.
- ✓ Lenguaje rico en operadores y expresiones.
- ✓ Flexible, conciso y eficiente.
- ✓ Presenta programación modular.
- ✓ Introduce nuevas palabras claves y operadores para manejo de clases (18).

C++ está considerado por muchos como el lenguaje más potente, debido a que permite trabajar tanto a alto como a bajo nivel. Además, se trata de un lenguaje de programación estandarizado (ISO/IEC 14882:1998), ampliamente difundido, y con una biblioteca estándar C++ que lo ha convertido en un lenguaje universal, de propósito general, y ampliamente utilizado tanto en el ámbito profesional como en el educativo. Posee una serie de propiedades difíciles de encontrar en otros lenguajes de alto nivel:

- ✓ Posibilidad de redefinir los operadores (sobrecarga de operadores).
- ✓ Identificación de tipos en tiempo de ejecución (RTTI) (18).

⁸ Un **Lenguaje de programación multiparadigma** es el cual soporta más de un paradigma de programación. Permiten crear programas usando más de un estilo de programación.

C++ fue el lenguaje seleccionado porque además de que lo utiliza el framework QT seleccionado, posee características que lo realzan de los demás lenguajes. Existen varias bibliotecas con algoritmos desarrollados en C++, simplemente es tomarlos y adaptarlos a tu solución. También se seleccionó por las ventajas que brinda la programación orientada a objetos y por el conocimiento y la experiencia adquiridos con este lenguaje.

1.7.6. Paradigma de Programación: Programación Orientada a Objeto (POO)

Un paradigma de programación es un estilo fundamental de programación definido por la forma de dar soluciones a problemas. Proporciona y determina la visión que el programador tiene acerca de la ejecución de un programa. En conclusión, facilita expresar el concepto de computación de diversas maneras. La programación orientada a objeto se ha convertido en un nuevo paradigma junto con la programación imperativa, la funcional, la lógica, entre otras (10).

Los programas que utilizan la Programación Orientada a Objeto (POO) tienen una colección de objetos que interactúan intercambiando mensajes que transforman estados. La orientación es, obviamente, hacia los objetos. Los elementos de programación: modelado de objetos, clases, herencia, encapsulamiento. Soporta diferentes lenguajes de programación como: C++, Java, C#, Eiffel, PHP.

La orientación a objeto (OO) ha tomado por asalto y en forma legítima al mundo del software. Como medio para la generación de programas, tiene varias ventajas. Fomenta una metodología basada en componentes para el desarrollo de software, de manera que primero se genera un sistema mediante un conjunto de objetos, luego podrá ampliar el sistema agregándole funcionalidad a los componentes que ya había generado o agregándole nuevos componentes, y finalmente podrá volver a utilizar los objetos que generó para el sistema cuando cree uno nuevo, con lo cual reducirá sustancialmente el tiempo de desarrollo de un sistema (10).

El paradigma orientado a objeto depende de ciertos principios fundamentales: Abstracción, encapsulamiento, modularidad, herencia, jerarquía y polimorfismo. La programación orientada a objetos considera a un programa como una colección de agentes ampliamente autónomos, llamados objetos. Tiene características muy definidas como paradigma de programación:

- ✓ Cada objeto es responsable de tareas específicas. Es mediante la iteración de los objetos que avanza el cómputo. Por lo tanto, en cierto sentido la programación es, ni más ni menos, la simulación de un universo modelo.

- ✓ El comportamiento de cada objeto queda determinado por la clase que pertenece, cada objeto es un ejemplar de alguna clase, donde todos los objetos de la misma clase se comportarán de una forma similar antes de una solicitud similar.
- ✓ Un objeto exhibirá su comportamiento ante la invocación de un método como respuesta a un mensaje. La interpretación de un mensaje es decidido por el objeto y puede diferir de una clase de objetos a otra.
- ✓ Las clases pueden organizarse en un árbol de herencia jerárquico. Las clases que se encuentran en el nivel más bajo tienen acceso, pueden usar datos y comportamientos asociados con clases más altas del árbol.
- ✓ Al reducir la independencia entre los componentes de software, la programación orientada a objetos permite el desarrollo de sistemas de software reutilizables. Tales componentes pueden crearse y probarse como unidades independientes, aisladas de otras partes de una aplicación de software (10).

Seguindo las características de este paradigmas es que se plantea que la siguiente investigación está dirigida por la programación orientada a objetos, siendo esta una de las más similares al pensamiento real, definiendo elementos como objetos, clases, mensajes y métodos; y situaciones como son: la herencia entre clases y la asignación de responsabilidades a los objetos.

1.7.7. Entorno de Desarrollo: Eclipse

Eclipse es un Entorno de Desarrollo Integrado libre y de código abierto desarrollado por la fundación Eclipse e impulsado por la IBM. Se caracteriza por ser muy flexible y fácilmente extensible mediante plugins⁹, lo que permite que cada desarrollador pueda adaptarlo a sus propias necesidades y características. Brinda soporte para tecnologías centradas en el manejo de datos. Aunque se encuentra enfocado fundamentalmente para el desarrollo sobre Java soporta, mediante plugins, el desarrollo sobre otros lenguajes como C#, C++, PHP, etc. Es multiplataforma por lo que se puede utilizar lo mismo sobre el sistema operativo Linux o Windows (14).

El entorno de desarrollo que se utilizará para la implementación del componente es Eclipse por sus grandes potencialidades y su posibilidad de extensión mediante el uso de plugins, además por presentar funcionalidades más convincentes y realmente necesarias para llevar a cabo el componente

⁹ **Plugin:** Programa que puede anexarse a otro para aumentar sus funcionalidades.

eficiente. También corre en diferentes sistemas operativos permitiendo poder programar en cualquiera de ellos. La principal razón es que se está familiarizado con el mismo, permitiendo mayor rapidez de trabajo porque este constituye uno de los elementos que influye en el tiempo de terminación y desarrollo del componente.

1.7.8. Gestor de Base de Datos: PostgreSQL

PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Fue concebido para un mejor mantenimiento y ajuste menor que los productos de los proveedores comerciales, conservando todas las características, estabilidad, rendimiento y nunca ha presentado caídas en varios años de operación de alta actividad. Está disponible en casi cualquier sistema operativo (Unix, Windows (13)).

Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una cantidad de peticiones simultáneas de manera correcta. Ofrece control de concurrencia multi-versión, soportando casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, y tipos y funciones definidas por el usuario). Tiene la capacidad de comprobar la integridad de referencia, así como también de almacenar procedimientos en la propia base de datos.

Postgre SQL fue seleccionado por ser un gestor de base de datos libre y por estar disponible en cualquier sistema operativo, permitiendo que el componente pueda acceder a la base de datos. Además por ajustarse y soportar la cantidad de memoria que posee el componente, así como por permitir realizar consultas simultáneas.

1.8. Conclusiones Parciales

El análisis de los conceptos asociados al dominio del problema y el estudio a nivel nacional e internacional de las empresas que utilizan software para el análisis del proceso petrofísico de pozos petroleros, permitió conocer el estado actual de la problemática existente, demostrándose la necesidad de implementar el componente de interpretación y análisis de registros de pozos.

La descripción de los métodos de evaluación a utilizar permitió seleccionar los Convencionales como los más adecuados para realizar la interpretación. Además el análisis de las herramientas y tecnologías propuestas por la arquitectura del sistema AnPer, facilita un mejor desarrollo del componente de interpretación.

Capítulo 2: Descripción y Análisis de la solución.

2.1.Introducción

En este capítulo se realiza una descripción detallada de los casos de usos y actores del sistema, así como sus diagramas y relaciones correspondientes. Se abordan las capacidades del componente para analizar e interpretar los registros, cumpliendo los requisitos funcionales definidos. Además se establece la arquitectura definida facilitando la comprensión del componente y los patrones de diseño utilizados en la implementación.

2.2.Características de la solución propuesta

Después de analizar la situación actual que presenta AnPer, se concluye que el componente a construir se implementará utilizando tecnología libre según las políticas del proyecto AnPer y del país. El componente debe caracterizarse por estar acoplado a la forma y estilo de trabajo en la empresa. Su principal objetivo será, de conjunto con la integración a la plataforma AnPer, garantizar menos gastos a la economía cubana, menor dependencia de empresas propietarias y presentar una estructura flexible para posteriores actualizaciones; en vista a lograr mejores resultados sobre la base de un ambiente cómodo y amigable para todos sus usuarios. En resumen, el resultado debe ser un componente sumamente flexible en la construcción de los modelos de interpretación, que ofrecerá un conjunto de opciones y le permitirá al usuario, petrofísico, una mejor interpretación de los resultados obtenidos y le facilitará la toma de decisiones.

2.3.Especificación de los requerimientos del componente

Los requerimientos de un sistema definen las cualidades, funcionalidades y características que debe poseer un software (14).

Requerimientos Funcionales

Después de realizar un análisis detallado del Negocio se logró identificar como Requisitos Funcionales del sistema los siguientes:

RF1. El sistema debe permitir gestionar un modelo en el escenario

RF1.1 El sistema debe permitir en el escenario crear un modelo.

RF1.2 El sistema debe permitir en el escenario eliminar un modelo.

RF1.3 El sistema debe permitir en el escenario cambiar el nombre del modelo.

RF2. El sistema debe permitir Conformar Modelo.

RF2.1 El sistema debe permitir definir volúmenes.

RF2.1.1 El sistema debe permitir mostrar el listado de volúmenes que desea utilizar en la construcción del modelo.

RF2.1.2 El sistema debe dar la posibilidad de seleccionar a partir del listado dado anteriormente los componentes de formación (Rocas, Minerales y Fluidos) a los cuales se le podrá hallar la cantidad que representan en una curva.

RF2.2 El sistema debe permitir definir ecuaciones.

RF2.1.1 El sistema debe mostrar el listado de las ecuaciones que desea utilizar en la construcción del modelo.

RF2.1.2 El sistema debe permitir seleccionar las ecuaciones que se desean utilizar para construir su modelo.

RF3. El sistema debe permitir inicializar los parámetros que van a conformar el modelo.

RF3.1 El sistema debe dar la posibilidad de inicializar y cambiar los parámetros que no están definidos en la base de datos.

RF4. El sistema debe permitir Ejecutar Modelo.

RF4.1 El sistema debe dar la posibilidad de comprobar que el volumen no sobrepase la cantidad de ecuaciones conformadas.

RF4.2 El sistema debe conformar sistemas de ecuaciones lineales.

RF4.3 El sistema debe permitir resolver sistemas de ecuaciones lineales.

RF5 El sistema debe permitir visualizar los resultados obtenidos en gráficas de curvas.

Requerimientos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido y confiable. Son los aspectos del sistema visibles para el usuario, que no están relacionados de

forma directa con el comportamiento funcional del sistema. Los requerimientos no funcionales forman una parte significativa de la especificación. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto, esto puede marcar la diferencia entre un producto bien aceptado y uno con poca aceptación (4).

RNF Usabilidad

- ✓ El componente debe ser usable por cualquier tipo de usuario con experiencia, básica, media o avanzada en los procesos de interpretación y análisis petrofísicos.
- ✓ El sistema seguirá estándares de interfaz de sistemas internacionales de esta área de negocio.
- ✓ El sistema orienta su diseño a partir de sistemas internacionales que manejan el negocio de la interpretación petrofísica, manteniendo colores frescos y de fácil ajuste a la visión del usuario final, midiendo su realización a partir de la aceptación final del cliente.

RNF Apariencia o interfaz externa.

- ✓ El sistema tendrá un menú de herramientas para que el usuario pueda disponer de él en la medida de sus necesidades, visible todo el tiempo para facilitar el fácil acceso a las funcionalidades del software.
- ✓ El sistema tendrá un ambiente agradable y sencillo, combinado con colores y una estructura amigable que permita que el usuario se adapte con facilidad.
- ✓ El sistema permitirá claridad en las funcionalidades que brinda, con términos asociados a los procesos de interpretación y análisis petrofísicos, mejorando la comprensión del sistema.
- ✓ El sistema tendrá una buena navegabilidad, propiciando varias opciones para acceder a cada servicio brindado.

RNF Software

- ✓ Se utilizará como lenguaje de programación C++.
- ✓ Se utilizará como Sistema Operativo, Windows 2000/XP/Vista, Linux, Unix, Mac OSX.
- ✓ Se utilizará con framework de desarrollo Qt 4.2.

RNF Reusabilidad

- ✓ Las funcionalidades del sistema se desarrollarán en forma de componentes con el objetivo de la reutilización de estos en futuras versiones del sistema y en el desarrollo de nuevos sistemas y aplicaciones.

- ✓ El sistema debe permitir posteriores modificaciones y actualizaciones, así como modificar elementos ya existentes.
- ✓ El sistema debe permitir la integración de nuevas funcionalidades.

RNF Portabilidad

- ✓ El sistema debe ser multiplataforma, siendo compatible con cualquier tipo de Sistema Operativo (GNU/Linux, Microsoft Windows, MAC).

2.4. Modelo de Casos de Uso del Sistema

A continuación se darán a conocer los actores que interactuarán con el componente para visualizar curvas de registros de pozos y los casos de uso a implementar (4).

Descripción de los actores del Sistema

Los **actores del sistema**: Cada trabajador del negocio que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema (29). La descripción de los actores del sistema se encuentra a continuación en la Tabla 1.

Tabla 1 Descripción del actor del sistema.

Actores	Descripción
Petrofísico	Es la persona encargada de realizar las operaciones para los procesos de análisis petrofísicos y además de obtener como resultado de la entrada del archivo .LAS en el sistema, la Evaluación de las Formaciones utilizando el componente de Interpretación Mineral Solve. Además son las personas encargadas de elaborar Informes Valorativos sobre los resultados obtenidos durante las diferentes etapas de desarrollo de las investigaciones.

Casos de Uso del Sistema

Los **casos de uso del sistema** son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las capacidades o condiciones (requisitos funcionales) que debe cumplir el sistema. (4) En los anexos se pueden encontrar las descripciones de los casos de uso que aparecen en la tabla 2. (Ver anexo del 1 al 4)

Tabla 2 Casos de Usos del sistema.

Caso de uso del sistema	Prioridad
Gestionar Modelo	Crítico
Editar Modelo	Crítico
Inicializar Parámetros	Crítico
Ejecutar Modelo	Crítico
Visualizar Parámetros en gráficas de curvas.	Crítico

Diagrama de Casos de Uso del Sistema

Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo, por ejemplo la especialización y la generalización son relaciones. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo (30).

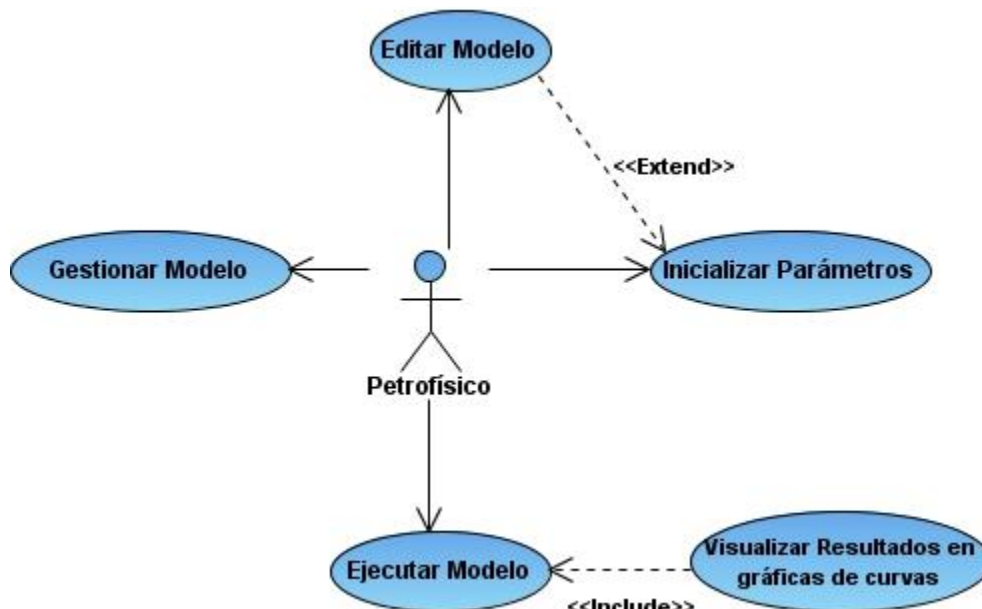


Figura 2 Casos de uso del Negocio.

2.5. Modelo de Diseño

Para lograr un buen diseño, se debe tener en cuenta algunos principios para considerarse teóricamente correcto. A diferencia de muchos tipos de metodologías que dedican una parte importante al diseño completo de la aplicación, las metodologías ágiles definen que el diseño debe ser lo más sencillo posible. Las metodologías clásicas intentan predecir mediante el diseño todo el futuro desarrollo de la aplicación, sin embargo esto no siempre funciona. Lo primero es que en ocasiones un diseño puede parecer muy bueno en papeles pero muy fallido al momento de implementar. Lo segundo es que estas formas de trabajo son muy recias al cambio, cualquier cambio en los requisitos o en el entorno puede significar largas horas de trabajo rectificando el diseño anterior.

Diagramas de Clases del Diseño

Los diagramas de clases son diagramas de estructura estática que muestran las clases del sistema y sus interrelaciones. Son el pilar básico del modelado con UML, siendo utilizados tanto para mostrar lo que el sistema puede hacer (análisis) y como puede ser construido (diseño) (31).

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. Principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa.

A continuación se muestra el diagrama de clases del componente de interpretación en la figura 3 y su descripción están en los anexos (Ver anexos del 6 al 15).

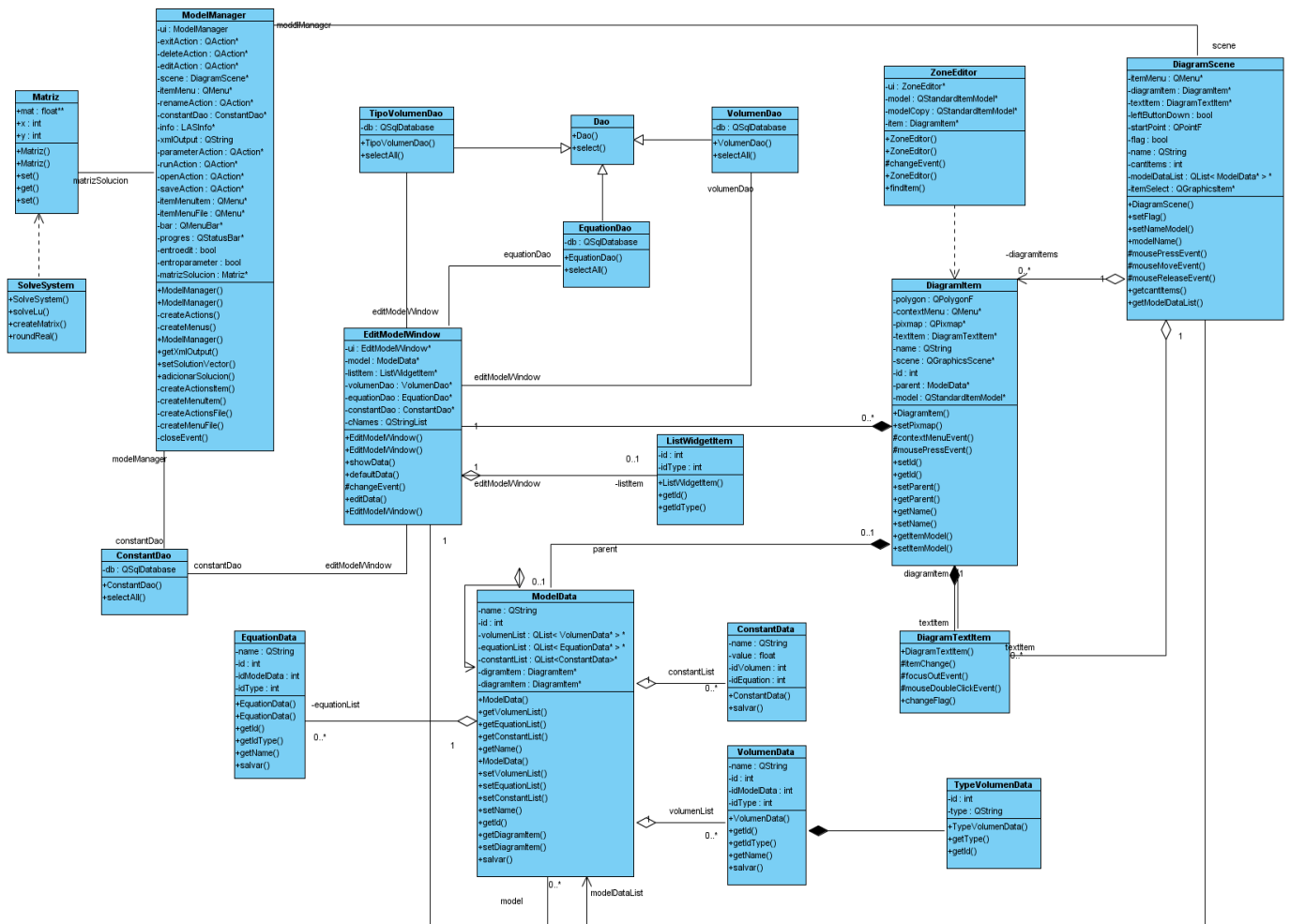


Figura 3 Diagrama de Clases del Diseño.

2.6. Arquitectura de Software

El término Arquitectura de Software (AS) lleva más de una década por lo que existen varias definiciones y todavía no se ha llegado a una definición común entre la mayoría de los estudiosos del tema. La IEEE-1471-200 define: “La Arquitectura de Software es la organización fundamental de un sistema enmarcada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución” (3).

Paul Clements define: “La arquitectura de software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para

alcanzar la misión del sistema. La vista arquitectónica es una vista abstracta, aportando el más alto nivel de comprensión y la supresión o diferimiento del detalle inherente a la mayor parte de las abstracciones.”

“La Arquitectura de Software de un sistema de programa o sistema de computación es la estructura o las estructuras del sistema, la cual comprende los componentes del software, las propiedades de esos componentes visibles externamente, y las relaciones entre ellos.”

Philippe Kruchten define: “La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, como la confiabilidad, escalabilidad, portabilidad, y disponibilidad”.

Se puede decir que la **AS** es una visión abstracta de alto nivel, que propone como van a estar relacionados los componentes de software, además tiene una relación muy estrecha con los requisitos no funcionales del sistema: que expresan las propiedades que debe tener el sistema, define estilos o combinación de estos para una solución y es esencial para el éxito o el fracaso de un proyecto (3).

Estilos Arquitectónicos

Un estilo arquitectónico define una codificación particular de elementos de diseño. El estilo limita los tipos de elementos de diseño y su acuerdo formal. Un estilo arquitectónico restringe tanto los elementos de diseño como las relaciones formales entre ellos (4).

Los estilos arquitectónicos tradicionalmente se reconocen como las cuatro “C” de la **AS**:

- ✓ Componentes.
- ✓ Conectores.
- ✓ Configuraciones.
- ✓ Restricciones.

Arquitectura basada en componentes

Tomando los resultados de la línea base de la arquitectura para (AnPer), la arquitectura que se va a utilizar es basada en componentes. Esto permitirá describir un acercamiento al diseño del sistema como un conjunto de componentes que exponen interfaces bien definidas y que colaboran entre sí (19).

Se puede decir que el desarrollo de software basado en componentes es la evolución natural de la ingeniería software para mejorar la calidad, disminuir los tiempos de desarrollo y gestionar la creciente complejidad de los sistemas. Sin embargo, disponer de componentes no es suficiente tampoco, a menos que seamos capaces de reutilizarlos, y reutilizar un componente no significa usarlo más de una vez, sino que implica la capacidad del mismo de ser utilizado en contextos distintos a aquellos para los que fue diseñado (4).

Arquitectura en Capas

Se llegó a la conclusión de que el estilo arquitectónico que más se adapta a la solución de arquitectura es la Arquitectura en Capas, la cual se basa en una distribución jerárquica de los roles y las responsabilidades para proporcionar una división efectiva de los problemas a resolver. Este estilo asegura que los componentes se estructuren en niveles o capas donde cada nivel invoca sólo al nivel inferior y las interfaces entre capas están claramente definidas. Es definido como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior. Su objetivo principal es separar la lógica del negocio de la del diseño (21)

Entre las principales ventajas de la arquitectura en capas se encuentran:

- ✓ Soporta un diseño basado en niveles de abstracción crecientes, lo cual a su vez permite la partición de un problema complejo en una secuencia de pasos incrementales.
- ✓ Admite optimizaciones y refinamientos.
- ✓ Proporciona amplia reutilización.

Se utilizará para la solución arquitectónica del componente la arquitectura en tres capas, que divide la aplicación en tres capas lógicas distintas:

- ✓ Capa de Presentación: esta capa se encarga de gestionar todos aquellos aspectos relacionados con la lógica de presentación de la aplicación como son las validaciones de la

información y la gestión de usuarios, pues es la capa que interactúa con el mismo. Se comunica con la capa de Negocio.

- ✓ Capa de Lógica del Negocio o Dominio: Se encarga de recibir las peticiones de los usuarios y contiene todo el código que define las reglas de negocio para dar respuestas a las mismas luego de su proceso. Se puede definir como el conjunto de reglas de negocio que abstraen el problema real a tratar.

- ✓ Capa de acceso a datos: Contiene los datos necesarios para la aplicación y también puede ofrecer servicios relacionados con la persistencia o recuperación de información, en este caso del componente se encargará de toda la gestión de curvas, volúmenes y parámetros que cada volumen tiene respecto a una ecuación.

2.7. Visión general de la Arquitectura

El diagrama ilustra la organización del modelo de diseño en capas lógicas e incluye la descripción de cada una de ellas. Esta vista general fue aplicada al componente de interpretación desarrollado.

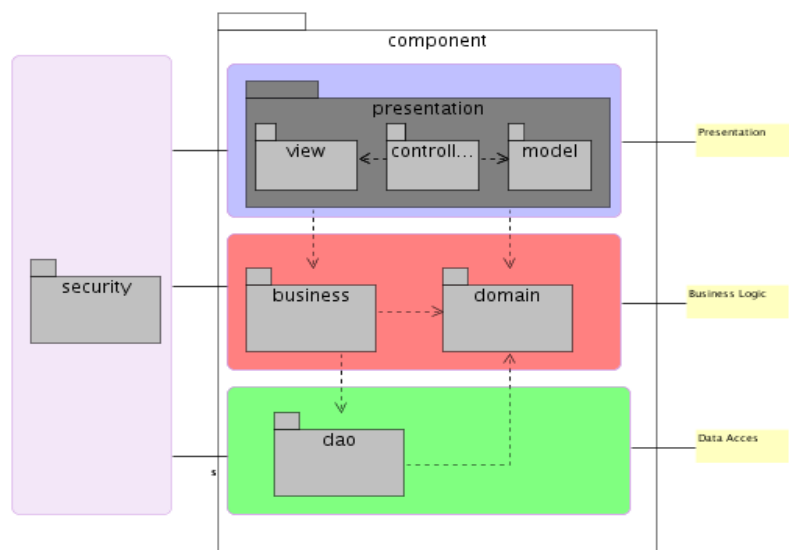


Figura 4 Vista general de la arquitectura.

Descripción de las Capas y sus componentes

Componentes de Interfaz de Usuario (view): Proveen el mecanismo que permite la interacción del usuario con la aplicación.

Componentes de Lógica de Interfaz de Usuario (controller): Ayuda a sincronizar las interacciones del usuario.

Componentes de Lógica de Interfaz de Usuario (model): Interpreta las acciones del mouse y el teclado, informándole al modelo y/o a la vista que cambien según resulte apropiado.

Componentes de Negocio (business): Implementan la lógica de negocio de la aplicación. Contienen las clases que implementan las reglas del negocio.

Componentes Entidades de Negocio (domain): Las entidades de negocio son usadas para pasar datos entre componentes. Las entidades representan los datos reales del negocio, como pueden ser Rocas y Fluidos. Las entidades de negocio son implementadas como clases orientadas a objetos que representan entidades del mundo real en la aplicación.

Componentes de Acceso a datos (dao): Los objetos de acceso a datos abstraen a la capa de negocio de la forma en que se accede a la capa de datos (19).

2.8. Patrones de Diseño utilizados

En el diseño del componente se utilizan los siguientes patrones para dar solución a las diferentes problemáticas (4).

Patrones Grasp¹⁰

Creador: Guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que debemos conectar con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento (4).

¹⁰ Son patrones generales de software para asignación de responsabilidades.

El patrón *creador* se percibe en las clases `VolumenData` y `EquationData` ya que son las que poseen los datos necesarios para crear una estructura de tipo `ModelData`. Esta última estructura contendrá los datos necesarios para realizar la interpretación.

Experto: Patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Con este no se pretende designar una idea oscura ni extraña; expresa simplemente la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen. Indica que el objeto con la información necesaria para desempeñar la responsabilidad debería encargarse de ella (4).

Este patrón se pone de manifiesto en las clases `VolumenData` y `EquationData` que son expertas y solo manejan información relacionadas con los datos de los volúmenes y las ecuaciones respectivamente.

Bajo Acoplamiento: Es un principio que debemos recordar durante las decisiones de diseño: es la meta principal que es preciso tener presente siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño. Soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que acrecientan la oportunidad de una mayor productividad (4).

Alta cohesión: Una clase tiene responsabilidades moderadas en un área funcional y colabora con las otras para llevar a cabo las tareas (4).

Patrones GOF

Fachada (Facade): Propone la selección de un objeto que represente un sistema u organización global para que sea controlador (4).

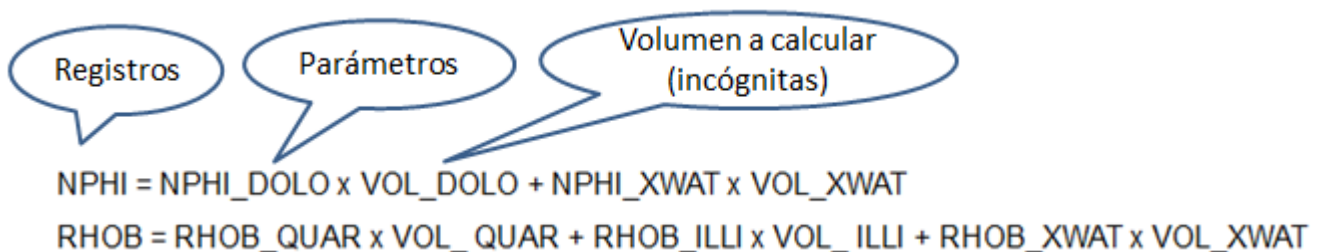
Este patrón es aplicable dentro del framework Qt en la interfaz visual de la aplicación.

Singleton: Nos permite asegurar que de una clase habrá solo una instancia, y proporciona un punto de acceso a ella global a todo el código (4).

Este patrón es aplicable en varios lugares dentro de la solución propuesta uno de ellos es el mantenimiento de una única instancia de conexión a bases de datos, lo cual agiliza el proceso de consulta y obtención de información de este medio de almacenamiento.

2.9. Algoritmo a utilizar: Descomposición (LU)

Se decide utilizar este algoritmo porque al seleccionar las diferentes ecuaciones y los componentes de formación (minerales, rocas o fluidos) se forma un sistema de ecuaciones lineales. Cada volumen tiene parámetros asociados a cada registro (curvas). Se formaría un sistema de ecuaciones de la siguiente forma: si el especialista selecciona como ecuaciones NPHI y RHOB, y como volúmenes DOLO y XWAT se tiene:



A este sistema de ecuaciones se le adiciona la **suma de los volúmenes** (componentes de la formación). La suma de todos los componentes de la formación (volúmenes) debe ser igual a 1. Esta condición se adiciona siempre al resto de las ecuaciones del modelo.

Como hay que resolver el sistema de ecuaciones donde solamente cambia el número de los registros, es decir cuando se trata de resolver un conjunto de sistemas de ecuaciones con la misma matriz, conviene utilizar un método de factorización, ya que sólo es necesario realizarla una vez. Además de que el tiempo de respuesta con respecto a otros métodos, como el Gauss-Seidel, es mucho más rápido según prueba internas desarrolladas durante la implementación.

El nombre de este algoritmo se deriva de las palabras inglesas "Lower" y "Upper", que en español se traducen como "Inferior" y "Superior". Estudiando el proceso que se sigue en la descomposición LU es posible comprender el porqué de este nombre, analizando cómo una matriz original se descompone en dos matrices triangulares, una superior y otra inferior (20).

La descomposición LU involucra solo operaciones sobre los coeficientes de la matriz [A], proporcionando un medio eficiente para calcular la matriz inversa o resolver sistemas de álgebra lineal.

Primeramente se debe obtener la matriz [L] y la matriz [U].

[L] es una matriz diagonal inferior con números 1 sobre la diagonal. [U] es una matriz diagonal superior en la que sobre la diagonal no necesariamente tiene que haber números 1.

El primer paso es descomponer o transformar [A] en [L] y [U], es decir obtener la matriz triangular inferior [L] y la matriz triangular superior [U].

PASOS PARA ENCONTRAR LA MATRIZ TRIANGULAR SUPERIOR (MATRIZ [U])

1. Hacer cero todos los valores abajo del pivote sin convertir este en 1.
2. Para lograr lo anterior se requiere obtener un factor el cual es necesario para convertir a cero los valores abajo del pivote.
3. Dicho factor es igual al número que se desea convertir en cero entre el número pivote.
4. Este factor multiplicado por -1 se multiplica luego por el pivote y a ese resultado se le suma el valor que se encuentra en la posición a cambiar (el valor en la posición que se convertirá en cero). Esto es:

$$\text{- factor * pivote + posición a cambiar}$$

PASOS PARA ENCONTRAR LA MATRIZ TRIANGULAR INFERIOR (MATRIZ [L])

Para encontrar la matriz triangular inferior se busca hacer ceros los valores de arriba de cada pivote, así como también convertir en 1 cada pivote. Se utiliza el mismo concepto de “factor” explicado anteriormente y se ubican todos los “factores” debajo de la diagonal según corresponda en cada uno.

Esquemáticamente se busca lo siguiente:

$$[L] = \begin{pmatrix} 1 & 0 & 0 \\ l_{12} & 1 & 0 \\ l_{13} & l_{23} & 1 \end{pmatrix} \quad [U] = \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}$$

Originalmente se tenía:

$$[A] = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Debido a que $[A] = [L][U]$, al encontrar $[L]$ y $[U]$ a partir de $[A]$ no se altera en nada la ecuación y se tiene lo siguiente:

$$[A] = [L][U]$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ l_{12} & 1 & 0 \\ l_{13} & l_{23} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}$$

Por lo tanto, si $Ax = b$, entonces $LUx = b$, de manera que $Ax = LUx = b$.

PASOS PARA RESOLVER UN SISTEMA DE ECUACIONES POR EL MÉTODO DE DESCOMPOSICIÓN LU

1. Obtener la matriz triangular inferior L y la matriz triangular superior U .
2. Resolver $Ly = b$ (para encontrar y).
3. El resultado del paso anterior se guarda en una matriz nueva de nombre “ y ”.
4. Realizar $Ux = y$ (para encontrar x).
5. El resultado del paso anterior se almacena en una matriz nueva llamada “ x ”, la cual brinda los valores correspondientes a las incógnitas de la ecuación (20).

2.10. Conclusiones Parciales

Con el desarrollo de este capítulo se puede arribar a las siguientes conclusiones:

- Con el estudio de los estilos arquitectónicos definidos por el proyecto Anper se logró realizar un modelo de diseño que se adapta a las características del componente a desarrollar.
- El análisis de las características del método de descomposición LU permitió seleccionarlo como el más idóneo para resolver los sistemas de ecuaciones lineales que se originen durante el desarrollo de la interpretación.

Capítulo 3: Implementación y Validación de la solución.

3.1. Introducción

En el presente capítulo se describen los principales artefactos desarrollados durante la fase de elaboración. Durante esta fase los flujos de trabajo, según la metodología seleccionada (RUP), de mayor importancia son los de Implementación y Pruebas por lo que se detallan los diagramas de componentes y diagramas de despliegue, herramientas fundamentales elaboradas durante el flujo de implementación, y se describen algunas de las pruebas realizadas a la aplicación así como los resultados.

3.2. Modelo de Implementación

El modelo de implementación representa la composición física de la implementación en términos de subsistemas de implementación y elementos de implementación (carpetas y ficheros, incluyendo código fuente, datos y ejecutables). Un modelo de implementación más detallado puede incluir un código fuente de bajo nivel y ficheros derivados, y sus relaciones con el modelo de diseño. Tales detalles son recomendados solo si se tiene una sincronización automatizada entre el modelo y los ficheros, lo cual no es el caso (32).

3.2.1. Vista de Implementación

La vista de desarrollo o de implementación se centra en la organización real de los módulos de software y describe la descomposición del software en subsistemas de implementación. En este diagrama se tiene en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software y la reutilización.

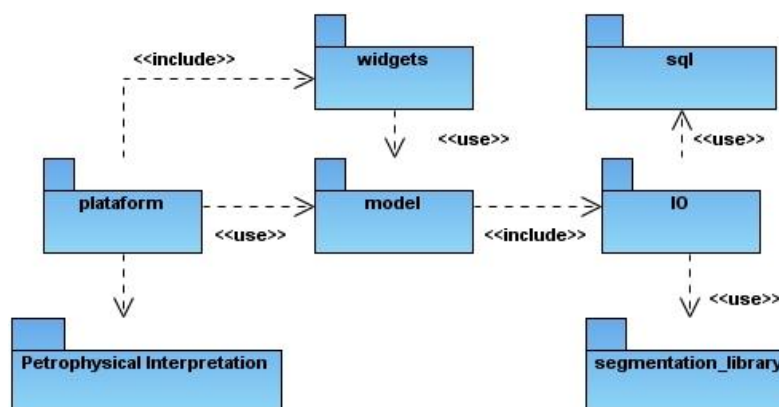


Figura 5 Vista de Implementación.

A continuación se detallan los componentes que integran el diagrama de implementación de esta solución:

- ✓ **petrophysical interpretation:** componente que permite la interpretación de los registros de pozos.
- ✓ **plataform:** módulo que contendrá los paquetes de clases o componentes generales de la plataforma.
- ✓ **widgets:** contiene todas las ventanas (form) y componentes (botones, menús, barras de herramientas, etc.) que serán visualizadas en el sistema a partir de los datos que les provea el **model**.
- ✓ **model:** contiene toda la lógica de negocio del sistema, es decir, algoritmos necesarios y reglas del negocio.
- ✓ **io:** contiene los elementos necesarios para la manipulación de entrada/salida de datos y la comunicación con fuentes de datos.
- ✓ **segmentacion_library:** contiene todos los componentes para la carga y salva de ficheros, ya sean ficheros .las, .lis y .txt.
- ✓ **sql:** contiene los elementos de acceso a los datos, dígame, consultas y configuraciones.

3.3. Diagrama de Componentes

Un componente es la parte modular de un sistema, desplegable y reemplazable que encapsula implementación, un conjunto de interfaces y proporciona la realización de los mismos. Típicamente contiene clases y puede ser implementado por uno o más artefactos (ficheros ejecutables, binarios, etc.). Son las piezas reutilizables de alto nivel a partir de las cuales se pueden construir los sistemas.

Algunos estereotipos estándar de componentes son:

- ✓ **Ejecutable:** Es un programa que se puede ejecutar en un nodo.
- ✓ **Biblioteca:** Es una biblioteca de objetos estática o dinámica.
- ✓ **Tabla:** Es una tabla de una Base de Datos.
- ✓ **Archivo:** Es un fichero que contiene código fuente o datos.
- ✓ **Documento:** Es un documento.
- ✓ **Página Web:** Es una página que se obtiene de la ejecución del sistema.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre sus elementos. Los componentes representados pueden ser datos, archivos, ejecutables, código fuente y directorios. A continuación se muestra el correspondiente al sistema desarrollado (4).

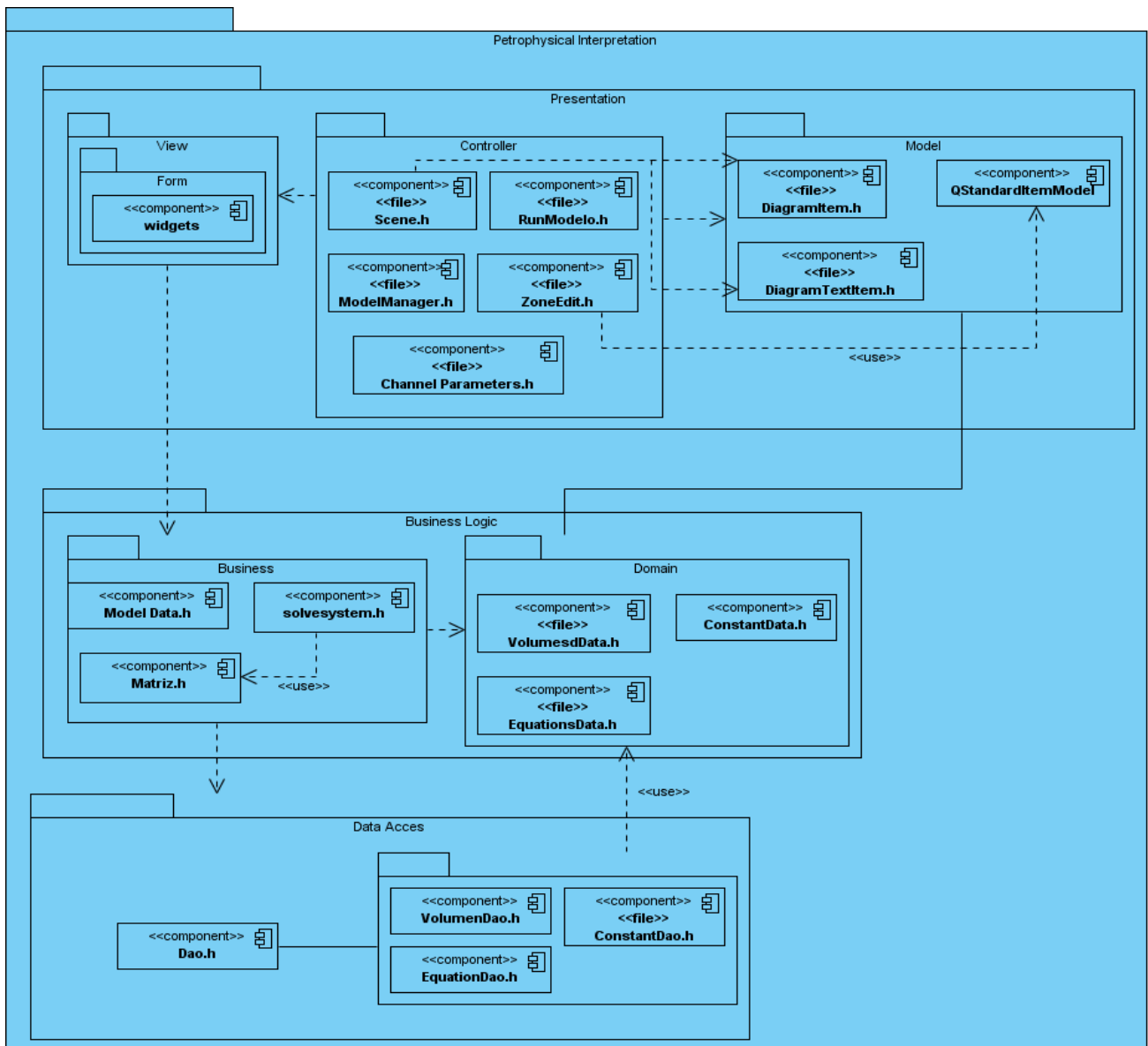


Figura 6 Diagrama de Componentes.

3.4. Diagrama de Despliegue

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos (21). El diagrama de despliegue propone la distribución física de los componentes hardware y software en el sistema. Es un conjunto de nodos unidos por conexiones de comunicación. Se satisfacen además parte de los requisitos no funcionales de hardware y software.

La intención del modelo de despliegue no es describir la infraestructura, sino el camino en el cual los componentes específicos deben corresponder a una aplicación que despliega a través de él (21). El diagrama de despliegue de la aplicación está representado por dos nodos. Uno de ellos es una PC Cliente, esta a su vez establece comunicación con el servidor de Base de Datos en el cual corre la Base de Datos. En la siguiente figura se muestra el Diagrama de Despliegue.

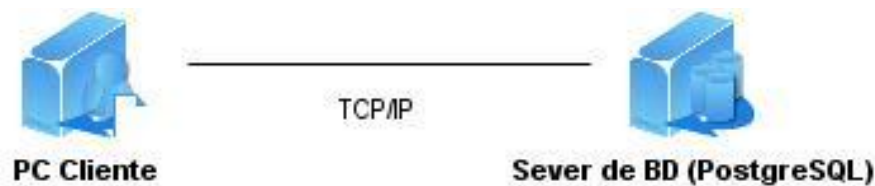


Figura 7 Diagrama de Despliegue.

3.5. Estructura de la Base de Datos

El diseño de bases de datos es el proceso por el que se determina la organización de una base de datos, incluido su estructura, contenido y las aplicaciones que se han de desarrollar. El rendimiento adecuado del sistema se deberá a una correcta selección de las tablas que formarán parte de la base de datos, así como de la información que contendrá cada una de ellas, las relaciones que se establezcan entre las mismas y del correcto uso de formas de normalización. Las clases del diseño y las realizaciones de los casos de usos son artefactos de entradas para realizar el artefacto principal que se genera en esta actividad: **el modelo de datos**, el cual describe la representación lógica y física de los datos persistentes. El resultado de la fase del diseño conceptual, que no es más que el diagrama entidad relación, constituyó la entrada para la realización del modelo entidad relación de la base de datos propuesta, el mismo se representa a continuación:

hecha de algún aspecto del sistema o componente. Las pruebas son técnicas de validación predominantes o procesos de ejecución de un programa con la intención de descubrir errores de una aplicación que antes no se habían descubierto (4).

3.7.1. Pruebas de Caja Negra

La prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. Los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene (4).

3.7.1.1 Casos de Pruebas

El propósito de los casos de pruebas es especificar una forma de probar el sistema, incluyendo las entradas con las que se prueba, los resultados esperados y las condiciones bajo las que ha de probarse. Los casos de prueba son un producto de desarrollo de software, que ayudan a validar y verificar las expectativas de los stakeholders.

Normalmente, un Caso de Prueba se deriva de un caso de uso en el modelo de casos de uso. Con estos Casos de Prueba se validan los requerimientos funcionales del sistema.

Los siguientes son Casos de Prueba comunes:

- Un Caso de Prueba que especifica cómo probar un caso de uso o un escenario específico de un caso de uso, incluye la verificación del resultado de la interacción entre los actores y el sistema. Además verifica que se satisfacen las precondiciones y pos condiciones especificadas por el caso de uso y que se sigue la secuencia de acciones especificadas por el caso de uso.
- Un Caso de Prueba que especifica cómo probar una realización de caso de uso-diseño o un escenario específico de la realización, puede incluir la verificación de la interacción entre los componentes que implementan dicho caso de uso.

Para verificar las funcionalidades del componente se realizaron Casos de Pruebas para cada escenario de los diferentes casos de uso. Los escenarios de un caso de uso se identifican describiendo los distintos caminos del flujo básico y flujo alternativo del caso de uso.

3.7.1.1.1 Caso de prueba para el caso de uso del sistema “Gestionar Modelo”.

Descripción General: El caso de uso es uno de los más importantes del software, pues a partir de él es que se van a realizar todas las funcionalidades.

Condiciones de Ejecución: Acceder a la barra de herramientas “Crear Modelo” o a al menú “Archivo”.

Tabla 3 Caso de Prueba de la sección Crear Modelo del caso de uso “Gestionar Modelo”.

Sección Crear Modelo.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
EC 1.1 Crear un nuevo modelo.	El sistema permite crear un nuevo modelo ventana en el área de dibujo.	El sistema crea una el modelo en el área de dibujo.	El usuario escoge la opción crear modelo. El sistema muestra el modelo creado en el área de dibujo automáticamente con un nombre.

Tabla 4 Caso de Prueba de la sección Cambiar Nombre del caso de uso “Gestionar Modelo”.

Sección Cambiar Nombre.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
EC 2.1: Cambiar nombre del modelo seleccionado correctamente.	El sistema permite cambiarle el nombre a un modelo existente.	El sistema cambia el nombre del modelo existente.	El usuario selecciona la opción cambiar nombre. El sistema muestra un cuadro de diálogo en el área de dibujo con el nombre anterior del modelo. El usuario introduce el nombre y da click en el botón “Aceptar”. El sistema comprueba que los datos

			<p>sean correctos.</p> <p>Si son correctos el sistema cambia el nombre del modelo.</p>
<p>EC 2.2:</p> <p>Cambiar nombre con datos incorrectos.</p>	<p>Cambia el nombre del modelo existente por un nombre con datos incorrectos.</p>	<p>El sistema realiza el cambio del nombre, el cual contiene caracteres incorrectos, constituyendo esto una no conformidad.</p>	<p>El usuario selecciona la opción cambiar nombre.</p> <p>El sistema muestra un cuadro de diálogo en el área de dibujo con el nombre anterior del modelo.</p> <p>El usuario introduce el nombre con caracteres incorrectos y da clic en el botón "Aceptar".</p> <p>El sistema cambia el nombre con datos incorrectos.</p>
<p>EC 2.3:</p> <p>Cambiar nombre del modelo dejando el campo vacío</p>	<p>No se cambia el nombre del modelo existente.</p>	<p>El sistema muestra un mensaje de error informando que el modelo tiene que tener un nombre</p>	<p>El sistema comprueba los datos, el campo de nombre no puede estar vacío y por tanto lanza un mensaje notificando la falta de información necesaria para realizar el cambio.</p>

Tabla 5 Caso de Prueba de la sección Eliminar Modelo del caso de uso "Gestionar Modelo".

Sección Eliminar Modelo.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
<p>EC 4.1: Eliminar un modelo.</p>	<p>El sistema permite eliminar un modelo existente.</p>	<p>Elimina satisfactoriamente un modelo seleccionado.</p>	<p>El usuario escoge la opción eliminar proyecto dando clic derecho sobre el modelo.</p> <p>El sistema elimina el modelo seleccionado.</p>

3.7.1.1.2 Caso de prueba para el caso de uso del sistema “Editar Modelo”.

Descripción General: El caso de uso se inicia cuando el usuario desea editar el modelo seleccionado, este caso de uso permite seleccionar los componentes de formación y las ecuaciones que van a conformar el modelo.

Condiciones de Ejecución: Dar clic derecho sobre el modelo seleccionado o acceder en la barra de herramientas al botón “Editar Modelo”.

Tabla 6 Caso de Prueba de la sección Definir Volumen del caso de uso “Editar Modelo”.

Sección Definir Volumen.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
EC 1.1: Definir Volumen.	El sistema permite seleccionar los volúmenes que van a formar parte del modelo.	Muestra una ventana donde se ve una lista de los volúmenes seleccionados.	<p>El usuario da click derecho sobre el modelo seleccionado y selecciona la opción editar modelo.</p> <p>El sistema muestra una ventana con una lista desplegable que permite seleccionar los volúmenes que van a conformar el modelo.</p> <p>El usuario selecciona los volúmenes y presiona el botón “flecha hacia la derecha”.</p> <p>El sistema muestra la lista de volúmenes seleccionados en el panel de lista de volúmenes y los elimina de la lista original.</p>
EC 1.2: Quitar Volumen.	El sistema permite quitar los volúmenes que no van a formar parte	Muestra una ventana donde se ve una lista de los volúmenes	<p>El sistema muestra la lista de volúmenes seleccionados en el panel de lista de volúmenes.</p> <p>El usuario selecciona los volúmenes que</p>

	del modelo.	seleccionados anteriormente.	<p>no van a formar parte del modelo anteriormente definido y presiona el botón “flecha hacia la izquierda”.</p> <p>El sistema elimina los volúmenes seleccionados del panel de lista de volúmenes y lo adiciona a la lista original.</p>
--	-------------	------------------------------	--

Tabla 7 Caso de Prueba de la sección Definir Ecuación del caso de uso “Editar Modelo”.

Sección Definir Ecuación.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
EC 2.1: Definir Ecuación.	El sistema permite seleccionar las ecuaciones que van a formar parte del modelo.	Muestra una ventana donde se ve una lista de las ecuaciones seleccionados.	<p>El usuario da click derecho sobre el modelo seleccionado y selecciona la opción editar modelo.</p> <p>El sistema muestra una ventana con una lista desplegable que permite seleccionar las ecuaciones que van a conformar el modelo.</p> <p>El usuario selecciona las ecuaciones y presiona el botón “flecha hacia la derecha”.</p> <p>El sistema muestra la lista de ecuaciones seleccionadas en el panel de lista de volúmenes y los elimina de la lista original.</p>

<p>EC 2.2: Quitar Ecuación.</p>	<p>El sistema permite quitar los volúmenes que no van a formar parte del modelo.</p>	<p>Muestra una ventana donde se ve una lista de las ecuaciones seleccionadas anteriormente.</p>	<p>El sistema muestra la lista de ecuaciones seleccionadas en el panel de lista de ecuaciones.</p> <p>El usuario selecciona las ecuaciones que no van a formar parte del modelo anteriormente definido y presiona el botón “flecha hacia la izquierda”.</p> <p>El sistema elimina las ecuaciones seleccionados del panel de lista de ecuaciones y lo adiciona a la lista original.</p>
---------------------------------	--	---	--

3.7.1.1.3 Caso de prueba para el caso de uso del sistema “Inicializar Parámetros”.

Descripción General: Este caso de uso se encarga de inicializar todos los parámetros seleccionados en el Editar Modelo. Se inicia cuando el usuario desea inicializar los valores de los parámetros del modelo seleccionado.

Condiciones de Ejecución: Dar clic sobre el modelo seleccionado y seleccionar la opción Parámetros o acceder en la barra de herramientas al botón “Inicializar Parámetros”. El modelo debe estar editado.

Tabla 8 Caso de Prueba del caso de uso “Inicializar Parámetros”.

Inicializar Parámetros.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
<p>EC 1.1: Inicializar Parámetros correctamente.</p>	<p>El sistema permite inicializar los parámetros que anteriormente fueron definidos en</p>	<p>Muestra una ventana donde se ve una tabla con los valores iniciales (cargados de la</p>	<p>El usuario da click derecho sobre el modelo seleccionado y selecciona la opción “Parámetros”.</p> <p>El sistema muestra una ventana con una lista de las ecuaciones seleccionadas.</p>

	el “Editar Modelo”.	base de datos o introducidos por el usuario) de los parámetros seleccionados con anterioridad.	<p>El usuario selecciona las ecuaciones que desea ver o cambiar el valor inicial y presiona el botón “Mostrar Valores”.</p> <p>El sistema muestra una tabla de ecuaciones contra volúmenes y el valor que representa cada volumen respecto a cada ecuación.</p> <p>El sistema permite que estos sean seleccionados y modificados.</p> <p>El usuario presiona el botón “Aceptar”.</p> <p>El sistema comprueba que los datos sean correctos, si son correctos el sistema emite un mensaje de éxito de la operación.</p>
EC 1.2: Inicializar Parámetros incorrectos.	El sistema permite inicializar los parámetros que anteriormente fueron definidos en el “Editar Modelo”.	Muestra un mensaje de error informando que los valores son incorrectos.	<p>El sistema muestra una ventana con una lista de las ecuaciones seleccionadas.</p> <p>El usuario selecciona las ecuaciones que desea ver o cambiar el valor inicial y presiona el botón “Mostrar Valores”.</p> <p>El sistema muestra una tabla de ecuaciones contra volúmenes y el valor que representa cada volumen respecto a cada ecuación.</p> <p>El usuario presiona el botón “Aceptar”.</p> <p>El sistema comprueba los datos, al ser incorrectos muestra un mensaje de error informando que los valores son incorrectos.</p>

<p>EC1.3: Inicializar Parámetros con campos vacíos.</p>	<p>El sistema no inicializa los parámetros que anteriormente fueron definidos en el “Editar Modelo”.</p>	<p>No realiza ninguna operación, por lo que no muestra un mensaje de error informando que los campos de los valores no pueden estar vacíos, constituyendo esto una no conformidad.</p>	<p>El sistema muestra una ventana con una lista de las ecuaciones seleccionadas. El usuario selecciona las ecuaciones que desea ver o modificar el valor inicial y presiona el botón “Mostrar Valores”. El sistema muestra una tabla de ecuaciones contra volúmenes y el valor que representa cada volumen respecto a cada ecuación. El usuario presiona el botón “Aceptar”. El sistema no valida que los datos estén vacíos.</p>
---	--	--	---

3.7.1.1.4 Caso de prueba para el caso de uso del sistema “Ejecutar Modelo”.

Descripción General: Este caso de uso se encarga de validar que todos los datos de entrada requeridos (modelo, parámetros y canales de datos) deben estar disponibles y ser válidos. Se inicia cuando el usuario desea ver el resultado del modelo gráficamente.

Condiciones de Ejecución: Dar clic sobre el modelo seleccionado y seleccionar la opción Ejecutar o acceder en la barra de herramientas al botón “Ejecutar Modelo”.

Tabla 9 Caso de Prueba del caso de uso “Ejecutar Modelo”.

Sección Ejecutar Modelo.			
Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Flujo central
<p>EC 1.1 Ejecutar Modelo correctamente.</p>	<p>El sistema permite ejecutar un modelo visualizando los resultados en una gráfica de curvas.</p>	<p>Muestra una nueva ventana mostrando los resultados obtenidos en una gráfica de curvas.</p>	<p>El usuario selecciona el modelo y da click en la barra de herramientas en el botón “Ejecutar Modelo”. El sistema valida que todos los datos (modelo, parámetros y canales de datos) estén correctos.</p>

			Si son correctos el sistema muestra una nueva ventana con los resultados en una gráfica de curvas.
EC1.2: Ejecutar Modelo incorrectamente	El sistema no permite ejecutar un modelo.	Muestra mensajes de errores informando lo que está mal.	<p>El usuario selecciona el modelo y da click en la barra de herramientas en el botón “Ejecutar Modelo”.</p> <p>El sistema valida que todos los datos (modelo, parámetros y canales de datos) estén correctos.</p> <p>Si se detectan errores durante la comprobación de la validación, el programa imprime los mensajes adecuados.</p>

Después de culminar la ejecución de los diseños de Casos de Pruebas al componente de interpretación se obtuvieron los siguientes resultados:

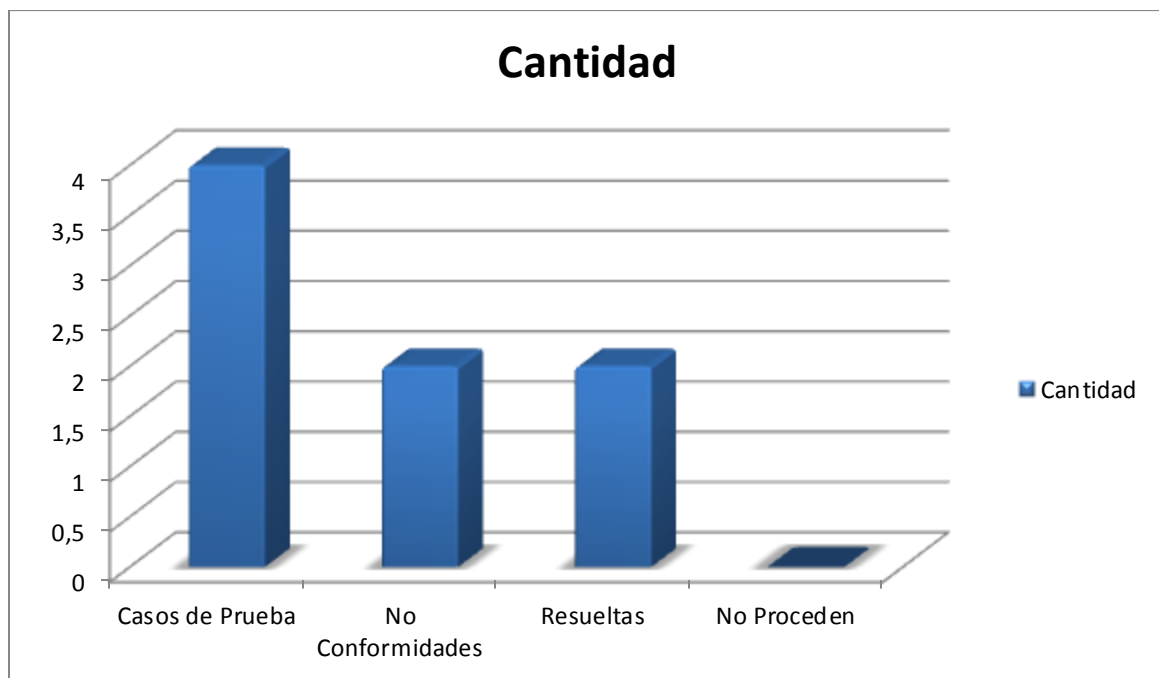
Primer ciclo de Iteración.

Tabla 10 Tabla de no conformidades.

#	No Conformidad	DCP	Escenario	Importancia	Solución	Estado
1	No se cambia el nombre del modelo existente.	Sección cambiar nombre del caso de prueba Gestionar Modelo.	Cambiar nombre del modelo dejando el campo vacío.	Alta	Comprobar que el nombre no esté vacío y mostrar un mensaje notificando informando el error.	Resuelta
2	El sistema no inicializa los	Caso de prueba	Inicializar Parámetros	Alta	Validar que los datos no estén	Resuelta

parámetros que anteriormente fueron definidos en el “Editar Modelo”.	inicializar parámetros.	con campos vacíos.		vacíos.	
--	-------------------------	--------------------	--	---------	--

La realización de un caso de prueba por cada caso de uso arrojaron los resultados que se presentan en el gráfico siguiente:



A partir de las pruebas de caja negra realizadas al componente se pudo constatar que el mismo cumple con los requerimientos funcionales identificados y permite dar solución a la problemática planteada al inicio de la investigación.

3.8. Conclusiones Parciales

Mediante el modelo de implementación y el modelo de despliegue se permitió un mejor entendimiento de la distribución física y lógica del componente. Además el diagrama de componentes, en el cual se describen los componentes a construir, su organización y dependencias, brindó claridad en cuanto a la estructura que posee la implementación del componente. Los estándares de codificación posibilitaron



desarrollar un código fuente de fácil comprensión. Y por último con las pruebas de caja negra realizadas al componente para corregir errores, se logró obtener un software confiable y que pueda ser integrado al sistema AnPer permitiendo facilitar el trabajo con el mismo.



Conclusiones Generales

Una vez finalizado el desarrollo del presente trabajo, se logró desarrollar un componente de software, con la finalidad de interpretar la información contenida en los registros de pozos.

Por otra parte, se arribaron a las siguientes conclusiones:

- ✓ Las herramientas, metodología de desarrollo y lenguajes utilizados, brindaron el soporte necesario para lograr un producto con los requerimientos deseados, además de proporcionarle al mismo una calidad y rendimiento acordes a las exigencias planteadas por el cliente.
- ✓ Se desarrolló un modelo de casos de uso del sistema que conjunto con los requisitos funcionales y no funcionales sentaron las bases para desarrollar todo el trabajo de manera organizada y poder dar cumplimiento a las expectativas del cliente.
- ✓ Se obtuvo un componente, desarrollado en entornos libres y que da respuesta al problema planteado al inicio de la investigación. Además, de ser un sistema capaz de informar al usuario los errores, inmediatamente que son producidos.
- ✓ El componente para interpretar registros de pozos petroleros permite a AnPer incorporarse en el mercado como un producto de análisis e interpretación petrofísica.
- ✓ Permite minimizar en tiempo el proceso de interpretación que antes de existir se realizaba con soluciones propietarias y procedimientos engorrosos.

Por todo lo antes expuesto se concluye que el objetivo y las tareas propuestos para el presente trabajo de diploma, se han alcanzado satisfactoriamente y que el componente puede ser desplegado en el CEINPET para su uso por los especialistas.

Recomendaciones

Tras haber cumplido con el objetivo trazado en esta investigación se recomienda:

- ✓ Tomar como material de consulta la propuesta de solución presentada para sistemas que tengan características similares.
- ✓ Extender el componente de manera que pueda ser utilizado no solo en el CEINPET, sino en cualquier empresa que requiera de este tipo de servicios.
- ✓ Desarrollar nueva versión mejorando las funcionalidades e incorporando la utilización de ecuaciones no lineales y cuadráticas.
- ✓ Mejorar la visualización del resultado de la interpretación con plantillas propias o definidas por el usuario.



Referencias Bibliográficas

1. **Neira Rodríguez, Yusleidy.** *Análisis y Diseño del Sistema para el análisis petrofísico.* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2009. Tesis de grado.
2. **HIDROCARBUROS, SUBSECRETARÍA DE.** *GLOSARIO DE TÉRMINOS PETROLEROS.*
3. **Vega Leyte Vidal, Francisco.** *DISEÑO DE LA ARQUITECTURA DE SOFTWARE PARA EL SISTEMA DE ANÁLISIS DE REGISTROS DE POZOS PETROLEROS (ANPER).* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2010. Tesis de grado.
4. **Áviles González, Rolando.** *Componente para visualizar curvas de registro de pozos.* Universidad de las Ciencias Informáticas. Ciudad de la Habana : s.n., 2010. Tesis de grado.
5. **Ariza Rojas, Maribel y Molina García, Juan Carlos .** 2.2.Conceptos. *INTRODUCCIÓN Y PRINCIPIOS BÁSICOS DEL DESARROLLO DE SOFTWARE BASADO EN COMPONENTES.* 2004.
6. Espectroscopía: La clave para la obtención de respuestas rápida y confiable. [En línea] [Citado el: 24 de 11 de 2011.] http://www.slb.com/~media/Files/resources/oilfield_review/spanish05/aut05/02_spectroscopy.aspx.
7. **Schlumberger.** Principios/Aplicaciones de la Interpretación de Registros By Schlumberger. [En línea] 8 de mayo de 2010. [Citado el: 6 de 11 de 2011.] <http://oilerhouse.blogspot.com/2010/05/principiosaplicaciones-de-la.html>.
8. **Schlumberger.** ELANPlus Program Theory. Capítulo 2: Interpretation Models.
9. **Castro Castiñeira, Olga .** *PETROFÍSICA: ANÁLISIS DE REGISTROS GEOFÍSICOS DE POZOS.* La Habana : s.n.
10. **González Torres, Miriam Yilieny.** *Subsistema de importación y exportación de registros de pozo.* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2010. Tesis de grado.
11. **Schlumberger.** Schlumberger. [En línea] 2011. [Citado el: 2 de 2 de 2011.] <http://www.slb.com/services/software/geo/geoframe/petrophysics/elanplus.aspx>.



12. Productos de Neuralog. [En línea] 14 de Agosto de 2001. http://www.adams-consulting.net/Productos/Neuralog/Productos_Neuralog/Neuralog_productos.htm.
13. **Mabel del Fresno Justo, Diana y Luarca Guerra, Maria Luisa.** *Aplicación Web para Estudios Farmacocinéticos.* Universidad de las Ciencias Informáticas. 2007. Tesis de grado.
14. **Ferrer Sánchez, Franklin y Romero Columbié, Franklin.** *Aplicación Web para el Alineamiento Múltiple de Estructuras de Proteínas.* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2007. Tesis de grado.
15. **Schmuller, Joseph.** Aprendiendo UML en 24 Horas. 968-444-463-X. [En línea] 2000. [Citado el: 15 de abril de 2011.] <http://bibliodoc.uci.cu/pdf/reg00004.pdf>.
16. **Blanco González, Aislen y Muñoz Morejón , Madelayne.** *Sistema de Ayuda Médica para la Atención de la Dislipoproteinemias.* Ciudad Habana : s.n., 2007. Tesis.
17. **Godoy Lara, Rolando y Castillo Martínez, Cristhian .** *Aplicación de la librería QT en la Elaboración de un Directorio Telefónico.* Universidad de las Ciencias Informáticas. La Habana : s.n., 20. Tesis de grado.
18. **López Estrada, Alain y Rivas Torrente, Ronni .** *Aplicación para la Gestión de la información del Puesto de Mando.* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2008. Tesis de grado.
19. **Pimienta Fernández, Yunier Alexander.** *Arquitectura de software .* Ciudad Habana : s.n., 2010.
20. Wikipedia. [En línea] 22 de Enero de 2007. [Citado el: 26 de Abril de 2011.] http://es.wikipedia.org/wiki/Factorizaci%C3%B3n_LU.
21. **Ortiz Fidalgo, Jackeline y Rios Morales, Fredy Hector .** *Desarrollo de un componente de software para importar reportes desde documentos Excel a Sistemas Gestores de Bases de Datos en el Centro de Investigaciones del Petróleo (CEINPET).* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2010. Tesis de grado.
22. Wikipedia. [En línea] Octubre de 2008. [Citado el: 1 de Mayo de 2011.] http://en.wikipedia.org/wiki/Programming_style .



23. **Hernández Rodríguez, Luz María.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos.* Mexico : Cámara Nacional de la Industria Editorial Mexicana, 1999.
24. **Pimienta Fernández , Yunier Alexander.** Excriba. [En línea] 2011. [Citado el: 6 de 4 de 2011.] <http://excriba.cedin.prod.uci.cu/eXcribaweb/dashboard/index#>.
25. **Mr. Robert Charlebois.** PetroLog - Advanced Log Analysis Software. [En línea] <http://www.petrolog.net/product-detail.asp?iSoftwareID=8>.
26. **UNIVERSIDAD SIMÓN BOLÍVAR.** [En línea] 2004. <http://www.gc.usb.ve/petrolab/Manual%209.9x%20CL.pdf>.
27. **Schlumberger.** *ELANPlus GeoFrame Advanced Petrophysical Interpretation.* Ayuda.
28. **Blümich, Bernhard.** RMN portátil. s.l. : Investigación y Ciencia, 2009, págs. pp. 82-87.
29. **Pressman, Roger S.** *Ingeniería del Software un enfoque practico.* La Habana : s.n., 2005.
30. **Olivares, Freddy Egdamar Paez.** Diseño UML. [En línea] 23 de abril de 2009. [Citado el: 8 de marzo de 2011.] <http://egdamar877.blogspot.com/2009/04/resumen-caso-de-uso-un-caso-de-uso-es.html>.
31. **Pinto, Julieth.** Análisis (UML). [En línea] 1 de marzo de 2009. [Citado el: 20 de marzo de 2011.] <http://diagraflu.blogspot.com/2009/03/diagramas-de-clases.html>.
32. **Acuña, Karennny Brito.** Diseño e implementación del sistema. [En línea] [Citado el: 2 de marzo de 2011.] <http://www.eumed.net/libros/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.
33. **Van Dillewijn, Derkje.** Software Trining from Schlumberger nformation Solutions. [En línea] Schlumberger, 16 de junio de 2009. [Citado el: 13 de junio de 2011.] <http://www.regonline.com/builder/site/Default.aspx?eventid=686277>.
34. **Halliburton.** Halliburton. [En línea] 2011. [Citado el: 13 de junio de 2011.] <http://www.halling>.



35. **Rios Morales, Fran Hector.** *Arquitectura de Software del Sistema para Graficar Columnas Litológicas de Pozos de Petróleo.* Universidad de las Ciencias Informáticas. La Habana : s.n., 2010. pág. 71, Tesis de grado.



ANEXOS

Anexo 1

CU 1. Gestionar Modelo.

Objetivo	<i>Permitir que el petrofísico pueda crear, eliminar un modelo e incluso editar el nombre del mismo en el escenario.</i>	
Actores	<i>Petrofísico</i>	
Resumen	<i>El caso de uso se inicia cuando el petrofísico desea crear o eliminar un modelo existente en el escenario. Además cuando desea editar el nombre de un determinado modelo en el escenario.</i>	
Complejidad	<i>Alta</i>	
Prioridad	<i>Alta</i>	
Precondiciones	<i>Debe estar cargado el fichero de datos con la información de un pozo.</i>	
Postcondiciones	<i>Se crea, elimina y edita el modelo.</i>	
Flujo de eventos		
Flujo básico Gestionar modelo en el escenario.		
	Actor	Sistema
1.	<i>Hace clic en menú Interpretación y escoge la opción Mineral Solve.</i>	<i>Muestra una ventana con las opciones que se puede realizar: crear, cambiar nombre, mover y eliminar modelo.</i>
2.	<i>Escoge una de las opciones definidas para el Modelo de Formación.</i>	<i>Ejecuta una de las opciones definidas en el Modelo de Formación:</i> <i>a. Sesión 1: Crear modelo.</i> <i>b. Sesión 2: Cambiar nombre.</i> <i>c. Sesión 3: Eliminar modelo.</i>
		<i>Termina el Caso de Uso.</i>
Sesión 1: Crear modelo en el escenario.		
	Actor	Sistema
3.	<i>En la barra de herramientas da clic en el botón "Crear".</i>	
4.	<i>El petrofísico da click dentro del escenario donde quiere crear el modelo.</i>	<i>a. Crea el modelo y automáticamente le asigna un nombre.</i>



Sesión 2: Cambiar nombre del modelo.		
	Actor	Sistema
3.	<i>Selecciona el modelo que desea cambiarle el nombre.</i>	<i>Cambia el color del modelo seleccionado.</i>
4.	<i>Da clic derecho encima del modelo y selecciona la opción cambiar nombre.</i>	<i>Muestra una ventana con el nombre que posee el modelo seleccionado.</i>
5.	<i>Escribe el nuevo nombre que va a tener el modelo.</i>	
6.	<i>Hace clic en el botón Aceptar.</i>	<ul style="list-style-type: none"> a. <i>Valida que el campo de nombre no este vacío.</i> b. <i>Valida que no contenga caracteres extraños(\$,#,@,!,%,^,&,*).</i> c. <i>Actualiza el nombre del modelo en el escenario.</i>
		<i>Termina el Caso de Uso.</i>
Flujos alternos		
El campo de nombre está vacío.		
	Actor	Sistema
6a		<i>Muestra un mensaje de alerta "El modelo debe tener nombre".</i>
		<i>Termina el Caso de Uso.</i>
El campo contiene caracteres extraños(\$, #, @, !, %, ^, &,*).		
	Actor	Sistema
6b		<i>Muestra un mensaje de alerta "El nombre del modelo no puede contener caracteres extraño".</i>
		<i>Termina el Caso de Uso.</i>
Presionar el botón de Cancelar.		
	Actor	Sistema
6c		<i>No actualiza el nombre del modelo.</i>
		<i>Termina el Caso de Uso.</i>
Sesión 3: Eliminar modelo		



	Actor	Sistema
3.	<i>Selecciona el modelo que desea eliminar.</i>	<i>Cambia el color del modelo seleccionado.</i>
4.	<i>Da clic derecho encima del modelo seleccionado y selecciona la opción eliminar.</i>	<i>Elimina el modelo seleccionado en el escenario.</i>
		<i>Termina el Caso de Uso.</i>
Relaciones	CU Incluidos	
	CU Extendidos	

Tabla 11 Descripción del Caso de Uso Gestionar Modelo

Anexo 2

CU 2. Editar Modelo.

Objetivo	<i>Editar los componentes que conforman el modelo.</i>	
Actores	<i>Petrofísico</i>	
Resumen	<i>El caso de uso se inicia cuando el petrofísico desea editar los componentes que desea utilizar en la construcción del modelo (volumen y ecuaciones)</i>	
Complejidad	<i>Alta</i>	
Prioridad	<i>Crítico</i>	
Precondiciones	<i>Debe haber creado al menos un modelo.</i>	
Postcondiciones	<i>Quedan editados los componentes para construir el modelo.</i>	
Flujo de eventos		
Flujo básico Editar modelo.		
	Actor	Sistema
1.	<i>Selecciona el modelo que desea editar.</i>	<i>Cambia el color del modelo seleccionado.</i>
2.	<i>Da clic derecho encima del modelo y selecciona la opción "Editar".</i>	<i>Muestra la ventana donde puede seleccionar los componentes a editar.</i> <i>a. Sección1: Definir Volúmenes (minerales, rocas y fluidos).</i> <i>b. Sección2: Definir Ecuaciones (lineales).</i>
		<i>Termina el Caso de Uso.</i>
Sección 1: Definir Volúmenes		



	Actor	Sistema
3.		<i>Muestra un listado con los volúmenes (minerales, rocas o fluidos).</i>
4.	<i>Selecciona los volúmenes que desea (minerales, rocas o fluidos).</i>	
5.	<i>Da clic en el botón flecha hacia la derecha.</i>	<i>Visualiza en el panel de lista de volúmenes los minerales, rocas o fluidos seleccionados y los elimina de la lista de volúmenes inicial.</i>
		<i>Termina el Caso de Uso.</i>

Flujos alternos**Eliminar volúmenes (minerales, rocas, fluidos) que se adicionaron a la lista.**

	Actor	Sistema
5.	<i>Hace clic en el elemento que desea quitar panel de lista de volúmenes.</i>	
6.	<i>Hace clic en la flecha hacia la izquierda.</i>	<i>Elimina el elemento seleccionado del panel de volúmenes y lo adiciona a la lista original.</i>
		<i>Termina el Caso de Uso.</i>

Sección 2: Definir Ecuaciones

	Actor	Sistema
3.	<i>Selecciona las ecuaciones.</i>	<i>Visualiza un listado de ecuaciones lineales.</i>
4.	<i>Selecciona de la lista la ecuación que desea y da clic en el botón flecha hacia la derecha.</i>	<i>Muestra en el panel de lista de ecuaciones las ecuaciones seleccionadas.</i>
		<i>Termina el Caso de Uso.</i>

Flujos alternos**Eliminar ecuaciones que se adicionaron a la lista.**

	Actor	Sistema
	<i>Hace clic en el elemento que desea quitar del panel de lista de ecuaciones.</i>	
	<i>Hace clic en la flecha hacia la izquierda.</i>	<i>Elimina el elemento seleccionado de panel de ecuaciones y lo adiciona a la lista original.</i>
		<i>Termina el Caso de Uso.</i>



Relaciones	CU Incluidos	
	CU Extendidos	Inicializar Parámetros.

Tabla 12 Descripción del Caso de Uso Editar Modelo

Anexo 3

CU 3. Inicializar Parámetros.

Objetivo	<i>Inicializar los valores de las curvas.</i>	
Actores	<i>Petrofísico</i>	
Resumen	<i>El caso de uso se inicia cuando el petrofísico desea inicializar los valores de las curvas para realizar la interpretación.</i>	
Complejidad	<i>Alta</i>	
Prioridad	<i>Crítico</i>	
Precondiciones	El modelo esté editado.	
Postcondiciones	<i>Todas las curvas deben tener un valor inicial.</i>	
Flujo de eventos		
Flujo básico <Nombre del flujo básico>		
	Actor	Sistema
1.	<i>Da clic derecho encima del modelo y selecciona la opción "Parámetros".</i>	<i>Muestra la ventana de Inicializar Parámetros.</i>
2.	<i>Selecciona las ecuaciones que desea inicializar sus parámetros y hace clic en el botón Mostrar Valores.</i>	<i>Muestra una tabla con los valores que tiene un volumen respecto a una curva (se cargan de la base de datos).</i>
3.	<i>Da doble clic en el valor que desea cambiar</i>	
	<i>Introduce un nuevo valor y presiona la tecla "Enter".</i>	<i>Cambia el valor.</i>
4.	<i>Da clic en el botón "OK".</i>	<ul style="list-style-type: none"> a. <i>Comprueba que las celdas de la tabla no estén vacías.</i> b. <i>Comprueba que el valor entrado sea un número.</i>
		<i>Termina el caso de uso</i>
Flujos alternos		
4a. Celdas vacías.		
	Actor	Sistema
4.		<ul style="list-style-type: none"> a. <i>Muestra un mensaje de alerta "Los datos no son válidos, se encuentran campos vacíos."</i>
4b. Valor de entrada no es un número.		
	Actor	Sistema
4.		<ul style="list-style-type: none"> b. <i>Muestra un mensaje de alerta "Los valores de entrada deben ser números".</i>



Relaciones	CU Incluidos	
	CU Extendidos	<i>Ejecutar Modelo.</i>

Tabla 13 Descripción del Caso de Uso Inicializar Parámetros

Anexo 4

CU 4. Ejecutar Modelo.

Tabla 14 Descripción del Caso de Uso Ejecutar Modelo

Objetivo	<i>Validar, construir y resolver el sistema de ecuaciones.</i>	
Actores	<i>Petrofísico</i>	
Resumen	<i>El caso de uso se inicia cuando el petrofísico desea obtener los resultados de su modelo.</i>	
Complejidad	<i>Alta</i>	
Prioridad	<i>Crítico</i>	
Precondiciones	Se deben haber ejecutado los casos de uso Editar Modelo e Inicializar Parámetros.	
Postcondiciones	<i>Se obtienen nuevas curvas como resultado de la interpretación.</i>	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
1.	<i>Selecciona el modelo</i>	
2.	<i>Selecciona en la barra de herramientas el botón Ejecutar Modelo.</i>	<i>a. Conformar el sistema de ecuaciones. b. Resuelve el sistema de ecuaciones. c. Valida que el sistema de ecuaciones tenga solución. d. Valida que el resultado no sea infinito. e. Ejecuta el caso de uso incluido "Visualizar Resultados en gráficas de curvas".</i>
Flujos alternos		
2c Resultado infinito		
	Actor	Sistema
2c		<i>Muestra un mensaje de alerta "El sistema de ecuaciones formado no tiene solución".</i>
El sistema de ecuaciones no tiene solución.		
	Actor	Sistema
2d		<i>Muestra un mensaje de alerta "Los</i>



		<i>resultados obtenidos no son eficientes, vuelva a inicializar sus parámetros”.</i>
		<i>Termina el caso de uso</i>
Relaciones	CU Incluidos	<i>Visualizar Resultados en gráficas de curvas.</i>
	CU Extendidos	

Anexo 5

CU 5. Visualizar Resultados en gráficas de curvas.

Tabla 15 Descripción del Caso de Uso Visualizar Resultados en gráficas de curvas

Objetivo	<i>Visualizar los resultados de las ecuaciones mediante gráficas de curvas.</i>	
Actores	<i>Petrofísico</i>	
Resumen	<i>El caso de uso se inicia cuando el petrofísico desea visualizar los resultados de las ecuaciones mediante gráficas para realizar informes valorativos.</i>	
Complejidad	<i>Alta</i>	
Prioridad	<i>Crítico</i>	
Precondiciones	<i>Se debe haber realizado el caso de uso Ejecutar Modelo.</i>	
Postcondiciones	<i>Se muestran los resultados en gráficas.</i>	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
1.		<i>El sistema muestra las curvas asociadas de cada ecuación seleccionada anteriormente para ver cuál o cuáles de los registros utilizados no se corresponden con los originales.</i>
		<i>Termina el caso de uso.</i>
Relaciones	CU Incluidos	
	CU Extendidos	



GLOSARIO DE TÉRMINOS.

UCI: Universidad de las Ciencias Informáticas.

AnPer: Sistema para el Análisis e Interpretación de los Registros de Pozos.

CEINPET: Centro de Investigaciones del Petróleo.

Ficheros .LAS: (Formato estandarizado de la Asociación Canadiense de Registros). Representa el encabezado del registro de pozo y las curvas ópticas en forma digital. Es reconocido por tener el tamaño justo. Es portátil, accesible y fácil de usar. Su estructura simple se asemeja a las de una hoja de cálculo donde los datos se encuentran ordenados según la profundidad. Se puede abrir el archivo con cualquier editor de texto y extraer información del pozo en forma visual.

FMS (Micro Scanner Formation): obtiene imágenes orientadas, bidimensionales y de alta resolución de las variaciones de microrresistividad alrededor de la pared del pozo. El concepto de la medición es una extensión de la tecnología del registro de echados.

FMI (Fullbore Formation Microlmager): es la última generación de la familia de imágenes eléctricas de formación.

NMR: La **resonancia magnética nuclear (RMN)** es un fenómeno físico basado en las propiedades mecánico-cuánticas de los núcleos atómicos. RMN también se refiere a la familia de métodos científicos que explotan este fenómeno para estudiar las moléculas (espectroscopia de RMN), macromoléculas (RMN biomolecular), así como tejidos y organismos completos (imagen por resonancia magnética). (28)

DSI: Herramienta de generación de Imágenes Sónica.

Licencia QPL: Versión modificada de la GPL destinada a asegurar que debes entregar (nótese que la GPL no obliga a entregar el código fuente, solo dice que debe quedar disponible para todo aquel que obtenga el software) el código fuente de tu aplicación si distribuyes los binarios. Además debes garantizar que cualquiera puede modificar tu software, siempre y cuando te garantice que podrás usar sus modificaciones.



Crossplot: La opción de Crossplot en la mayoría de los programas utilizados para realizar interpretación petrofísica permite desplegar gráficos cruzados bidimensionales de las curvas desplegadas por el programa. Esto permite realizar interpretaciones avanzadas sobre el conjunto de datos que conforman el registro petrofísico del pozo.