

Universidad de las Ciencias Informáticas

Facultad 5



Título: Diseño e implementación del Módulo de Manejo de Archivos para SIPP v2.0.

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

Autora:

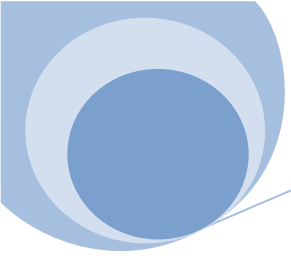
Yanai Rios Zamora

Tutores:

Ing. Omar Martínez Díaz

Cotutor:

Ing. Yordan Gallardo Avilés



DECLARACIÓN DE AUTORÍA

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

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

Firma del Autor

Firma del Autor

Firma del Tutor

Firma del Tutor

DATOS DE CONTACTO

Tutor: Ing. Omar Martínez Díaz

Universidad de las Ciencias Informáticas. La Habana. Cuba

Email: omarmd@uci.cu

Co-tutor: Ing. Yordan Gallardo Avilés

Universidad de las Ciencias Informáticas. La Habana. Cuba

Email: ygaviles@uci.cu

AGRADECIMIENTOS

Agradecemos a todos los que de una forma u otra tuvieron que ver con el resultado de esta tesis. A todos los que estuvieron durante estos meses, dándome apoyo y ayudándome en todo.

A Lannie, Aliander, Prevot que no fueron pocas las veces que los consulté, a cualquier hora, para aclarar dudas, gracias por su disposición y atención.

A mis amistades por resistirme en estos momentos tan complicados, por darme la mano cuando sientes que el mundo se te cae encima, por estar en las buenas y en las malas, gracias los quiero mucho

Agradecer a mi hermana, a Salet, mi niña linda, a mis padres por haberme guiado estos 22 años, por alentarme en cada momento, por su confianza, su apoyo, sus consejos aunque ellos crean que no les hago caso, hoy es el gran día, ya terminamos y aquí tienen el regalo que tanto esperaron.

Quiero dar un agradecimiento especial a mi tutor Omar Martínez Días y a Yordan Gallardo, por toda la ayuda que me brindaron. Por permitir ocupar su tiempo, y siempre estar dispuestos a colaborar conmigo, muchas gracias.

DEDICATORIA

A mi hermana, por estar siempre con nosotros, por preocuparse por mí, aunque es un poco reservada y por darme esa sobrina tan linda, para ella también es esta tesis.

A mi mamá, mi mejor amiga, por siempre estar a mi lado, por darme todo el apoyo del mundo y entenderme en todo momento. Sé que algún día haré lo mismo por tí.

A mi padre, que siempre me ha comprendido y ha estado a mi lado en todas las cosas buenas y malas, apoyándome y dándome fuerzas para seguir.

RESUMEN

Las empresas petroleras han automatizado su trabajo con el uso de software que facilite el aseguramiento de los datos en la perforación de pozos de petróleo. Proceso en el cual se maneja mucha información con un bajo control de almacenamiento. Esto repercute negativamente en la correcta realización de los procesos de la entidad, haciendo lento y difícil el acceso para la toma de decisiones. Como parte de la solución informática propuesta para eliminar estos problemas se encuentra el desarrollo del Módulo de Manejo de Archivos del Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP) versión 2.0, el cual constituye el objetivo de este trabajo.

En el estado del arte se realiza un análisis de algunos sistemas que en la actualidad implementan subsistemas similares al que propone esta investigación. Además, se abordan todas las herramientas y tecnologías utilizadas para la implementación del producto obtenido. Como resultado, el Módulo permite exportar todos los reportes en diferentes formatos (xls, pdf, csv y html) e integra todos los reportes que no se tenían en cuenta en la versión anterior del Módulo. Otras de las facilidades que brinda son las opciones de configuración a través de ficheros .yml, posibilitando que si existiera algún cambio en las plantillas de los reportes, no hay necesidad de realizar modificaciones en el código.

Palabras clave: exportar archivo, módulo de manejo de ficheros, reporte.

ÍNDICE

INTRODUCCIÓN	10
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	15
1.1 Introducción.....	15
1.2 Conceptos asociados al dominio del problema.....	15
1.3 Proceso de manejo de archivos en sistemas para la perforación de pozos de petróleo.....	16
1.4 Análisis de soluciones existentes.....	18
1.5 Comparación de soluciones existentes.....	20
1.6 Caracterización de los nuevos registros de pozos en perforación.....	21
1.7 Metodología de desarrollo.....	22
1.8 Herramientas y Tecnologías.....	24
1.8.1 Aplicación Web.....	25
1.8.2 Lenguaje de Programación PHP 5.....	25
1.8.3 Framework de desarrollo.....	26
1.8.4 Entorno Integrado de Desarrollo NetBeans 7.....	27
1.8.5 Sistema Gestor de Base de Datos.....	27
1.8.5.1 PostgreSQL 8.4.....	28
1.8.6 Servidor Web. Apache 2.1.....	28
1.8.7 Bibliotecas para el manejo de archivos.....	28
1.9 Conclusiones Parciales.....	30
CAPÍTULO 2: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA	31
2.1 Introducción.....	31
2.2 Propuesta del sistema.....	31
2.3 Reglas del Negocio.....	31
2.4 Modelo de Dominio.....	31
2.5 Requisitos.....	32
2.6 Modelo de caso de uso.....	34
2.6.1 Diagrama de caso de uso del sistema.....	35
2.6.2 Actores del Sistema.....	35

2.6.3	Descripción de los Casos de Uso.....	36
2.7	Diseño	39
2.8	Diagrama de clases del diseño.....	39
2.9	Arquitectura	42
2.10	Patrones de Diseño.....	46
2.11	Conclusiones Parciales.....	51
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA		52
3.1	Introducción.....	52
3.2	Implementación	52
3.2.1	Diagrama de componentes.....	52
3.2.2	Diagrama de despliegue	53
3.3	Pruebas de software	54
3.3.1	Modelo de prueba. Funcionalidad	54
3.4	Conclusiones parciales.....	58
CONCLUSIONES.....		60
RECOMENDACIONES.....		61
ANEXOS		64
GLOSARIO DE TÉRMINOS.....		69

ÍNDICE DE TABLAS

Tabla 1	Descripción de los Actores del Sistema.....	36
Tabla 2	Caso de Uso Exportar Archivo	39
Tabla 3	Pruebas de funcionalidad al sistema	56
Tabla 4	Pruebas de rendimiento al sistema.....	58

ÍNDICE DE FIGURAS

Figura 1	Modelo de Dominio.....	32
Figura 2	Diagrama de Caso de Uso del Sistema.....	35
Figura 3	Diagrama de clases del diseño exportar archivo	40
Figura 4	Arquitectura MVC	43

Figura 5 Ejemplo de datos representados de forma cíclica.....	44
Figura 6 Ejemplo de archivo de configuración para datos en forma cíclica.....	45
Figura 7 Ejemplo de datos representados de forma no cíclica.....	45
Figura 8 Ejemplo de archivo de configuración para datos en forma no cíclica.....	46
Figura 6 Patrón Creador en el Sistema.....	47
Figura 7 Patrón de Alta Cohesión en el sistema.....	48
Figura 8 Patrón de Bajo Acoplamiento en el sistema.....	49
Figura 9 Patrón Experto en el sistema.....	50
Figura 10 Patrón Controlador en el sistema.....	50
Figura 11 Diagrama de Componentes. Caso de uso Exportar.....	53
Figura 12 Diagrama de Despliegue.....	54

INTRODUCCIÓN

En el contexto cultural, económico y político en que está insertada la sociedad mundial, las tecnologías de la información y las comunicaciones (TIC´s) están en constante auge. Producto de su desarrollo, la sociedad hace uso de sus servicios para estar informados de todos los acontecimientos ocurridos. Además, las TIC´s constituyen una herramienta esencial para que las empresas mejoren su eficiencia y la calidad de sus productos y servicios, así como para fomentar su inserción en el mercado cada día más complejo y competitivo. La expansión de las TIC, en todos los ámbitos de nuestra sociedad, se produce a gran velocidad, siendo un proceso continuo por la aparición de nuevos elementos tecnológicos.

Debido al avance de las TICs, se ha mejorado la interoperabilidad entre software en el área petrolera. La interoperabilidad es la capacidad de los sistemas de tecnologías de la información y las comunicaciones, y de los procesos empresariales a los que apoyan, de intercambiar datos y posibilitar la puesta en común de información y conocimientos[1]. En las empresas petroleras, la utilización de las tecnologías de la información, constituye una excelente estrategia que permite potenciar el desarrollo de sus capacidades competitivas a nivel internacional. En Cuba, la Unión Cuba Petróleo, conocida por su siglas CUPET, es la entidad a cargo de todas las operaciones relacionadas con la producción, refinación y comercialización del petróleo y sus derivados[2].

Muchos software desarrollados para la industria petrolera, se comunican con otras aplicaciones dentro del entorno a través de ficheros, que no son más que un conjunto de información clasificada y almacenada de diversas formas para su conservación y fácil acceso en cualquier momento[3]. Estos ficheros que permiten almacenar información adyacente al pozo, facilitando la interpretación u obtención de los diversos parámetros físicos y geológicos comerciales, son denominados registros de pozos[4], y tienen gran importancia tanto para la toma de decisiones, como para la creación de futuros pozos productores.

Las empresas petroleras, han automatizado su trabajo con el uso de software que simplifique la interpretación de los datos recogidos en los pozos. De esta forma se logra la interoperabilidad con otros sistemas y se facilita el intercambio de información en la industria petrolera. Dicha automatización en algunos casos, responde a dificultades asociadas con el aseguramiento de los datos relacionados con la

perforación de pozos de petróleo, proceso en el cual existe mucha información con la que hoy día se opera manualmente, por lo que es necesario hacer subsistemas que tengan como objetivo gestionar el manejo de archivos. De esta forma, queda asegurada toda la información que se maneja en la perforación.

La Universidad de las Ciencias Informáticas (UCI) como centro productivo de software, desarrolla un gran número de productos con el objetivo de informatizar la sociedad. En la facultad 5 se encuentra el Centro de Desarrollo Informático Industrial (CEDIN), en el cual se desarrolla el Sistema Integral de Perforación de Pozos de Petróleo y Gas (SIPP). Este proyecto tiene como objetivo proveer a CUPET, de un sistema capaz de gestionar el proceso de perforación, así como, la información resultante del mismo.

Debido al segundo ciclo de desarrollo del sistema SIPP se hace necesario replantearse elementos que componen dicho sistema, como el Módulo de Manejo de Archivos, que se encarga del procesamiento de la información de distintos reportes manejados en el negocio de la perforación de pozos. Actualmente el diseño de clases que presenta el Módulo, no permite una posterior escalabilidad del sistema, en caso de que ello fuera necesario. En la versión 1.0 del Módulo se manipula solamente el formato xls, por lo que sería importante introducir nuevos formatos como pdf, csv (valores separados por coma) y html teniendo en cuenta que estos formatos son utilizados por algunas compañías extranjeras que intervienen en el proceso de perforación, para exportar la información. Además, se necesita que el módulo logre una interoperabilidad con estos sistemas.

En la versión 1.0 se decantaron algunos reportes para ser exportados a los formatos que posibilita la aplicación como:

- Record de Barrenas.
- Reporte de Productos Químicos.
- Reporte de Combustible.
- Cronograma Real de Perforación.

La ausencia de los reportes mencionados, trae consigo que se dificulte la toma de decisiones por parte de los directivos, ya que no se abarcan de manera íntegra todos los procesos en el área de la perforación de pozos. La biblioteca que actualmente utiliza el sistema es PHPEXcel en su versión 1.4.5, cuando el volumen de datos es considerable, se sobrecarga la caché ocasionando la caída momentánea de la aplicación. Ello afecta el rendimiento y la estabilidad de la misma, atentando contra los requisitos no funcionales establecidos por el cliente. Igualmente, el código del subsistema posee mala documentación, por lo que reconocer las características, ventajas, así como las funcionalidades que implica el desarrollo del proyecto, se hace muy engorroso. Esto dificulta el mantenimiento de la aplicación en posteriores versiones.

Una vez analizada la situación actual en el Módulo de Manejo de Archivos del sistema SIPP, surge el siguiente **problema a resolver**: ¿Cómo actualizar el procesamiento de la información del Módulo de Manejo de Archivos versión 1.0 de SIPP para que exista mayor rendimiento, lograr la completitud de los reportes y ayudar a la toma de decisiones?

A partir del problema a resolver se puede definir como **objeto de estudio**: El proceso de manejo de archivos en sistemas informáticos industriales.

Para darle solución al problema identificado se plantea como **objetivo general**: Diseñar e implementar el Módulo de Manejo de Archivos para SIPP, versión 2.0.

Enmarcado en el **campo de acción**: El proceso de manejo de archivos en sistemas para la perforación de pozos petroleros.

Para darle cumplimiento al objetivo general se han trazado las siguientes **Tareas de investigación**:

1. Elaboración del marco teórico de la investigación a partir del estado del arte que existe actualmente sobre el tema.
2. Selección de las herramientas, tecnologías y bibliotecas a utilizar para darle solución al problema planteado anteriormente.
3. Diseño e implementación del Módulo de Manejo de Archivos, versión 2.0.

4. Validación del resultado.

Por todo lo antes mencionado se **defiende la idea** que con el diseño e implementación del Módulo de Manejo de Archivos versión 2.0 elaborado, existirá mayor rendimiento, se logrará la completitud de los reportes y se facilitará el proceso de toma de decisiones.

Para el desarrollo de la investigación se hizo necesario aplicar algunos métodos científicos teóricos y empíricos. De esta forma se facilita la recopilación de la información necesaria para la realización de un subsistema de exportación de registros de pozos.

Métodos Teóricos

Histórico-Lógico: El método facilitó la realización de la primera parte de la investigación, permitiendo hacer un análisis de cuáles son las características de los sistemas que exportan registros de pozos petroleros, en el ámbito nacional e internacional, lo cual permite conocer el estado actual y la evolución del fenómeno. Además posibilitó la realización de un estudio del Módulo de Manejo de Archivos que actualmente existe en el proyecto SIPP, para de esta forma entender las deficiencias que presenta el módulo y buscar soluciones eficaces para el mismo.

Analítico-Sintético: Se empleó con el objetivo de analizar las teorías y documentos de los sistemas de manejo de archivos existentes, permitiendo describir los elementos más importantes relacionados con el proceso de exportar registros de pozos, para de esta forma establecer una comparación entre los mismos y seleccionar la mejor opción para resolver el problema planteado.

Métodos Empíricos

Observación: Empleado para obtener un nivel de realidad y familiaridad con el Módulo de Manejo de Archivos existente, para dominar cuáles son las dificultades que este presenta y en qué consiste exactamente, y no correr el riesgo de no entender a cabalidad los procesos desarrollados en el mismo.

Estructura del contenido

Capítulo 1 “Fundamentación Teórica”: En el capítulo se realiza una descripción de la fundamentación teórica, además se abordan los conceptos fundamentales asociados al dominio del problema y se caracterizan las principales soluciones existentes alrededor del objeto de estudio. Además se describen las herramientas y tecnologías a utilizar para el desarrollo de la aplicación. Para ello se analizan sus características y ventajas con el fin de dar cumplimiento al objetivo general de la investigación.

Capítulo 2 “Descripción de la solución propuesta”: En el capítulo se exponen los requisitos funcionales y no funcionales del subsistema, así como, el modelo de dominio, el diagrama de casos de uso y la descripción textual de los mismos. Se realiza el diagrama de clases del diseño con la pertinente descripción de los paquetes que lo conforman. La arquitectura y los patrones de diseño seleccionados también quedan expuestos en este capítulo, además de evidenciar cómo se adaptan al diseño del sistema.

Capítulo 3 “Implementación y Prueba”: En el capítulo se describe la implementación del Módulo de Manejo de Archivos a partir del diagrama de componentes y la distribución física del sistema a través del diagrama de despliegue. Se especifican las pruebas a realizar en el sistema para verificar su integridad y si se ajusta a los requerimientos planteados por el cliente.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realiza un estudio de los procesos de gestión de la información en la industria petrolera, para lo cual se hace necesario conocer algunos conceptos relacionados con el tema. Se abarca el proceso de obtención de información de los registros de pozos y la actualización de soluciones existentes. Además se investigan las herramientas, tecnologías, metodología, biblioteca y lenguajes a utilizar con el fin de solucionar el problema planteado.

1.2 Conceptos asociados al dominio del problema.

Para darle solución al problema planteado en este trabajo, es necesario poseer experiencia en el tema o al menos conocer del objeto de estudio de la investigación para poder conceptualizar y entender los términos que se manejan. A continuación, se explican una serie de conceptos, para un mejor entendimiento de los acápites que se abordan en el presente y posteriores capítulos.

El petróleo es una sustancia oleosa de color muy oscuro compuesta de hidrógeno y carbono, y se le llama hidrocarburo. Puede hallarse en estado líquido o gaseoso. En estado líquido es llamado aceite "crudo", y en estado gaseoso, gas natural. Es una mezcla líquida, compleja y de composición variable. Es un recurso natural no renovable y actualmente también es la principal fuente de energía en los países desarrollados[5]. La industria petrolera es la que se encarga de los procesos de exploración, extracción, refino, transportación y comercialización del petróleo y sus derivados. Está enfocada a ser más competitiva a través de mejorar la planeación de la exploración y explotación del petróleo eficientemente[4]. En Cuba la Unión CUPET es la entidad a cargo de todas las operaciones relacionadas con la producción, refinación y comercialización del petróleo y sus derivados[2]. Específicamente CEINPET, instituto perteneciente a la Unión CUPET (Cuba Petróleo) del Ministerio de la Industria Básica, se dedica a la investigación aplicada en la Industria del Petróleo Cubana, y a desarrollar programas y proyectos de investigación y desarrollo[6]. Este instituto es el encargado de procesar y evaluar todos los

datos referentes a los informes recibidos en el registro de pozos, técnicas geofísicas que se utilizan en las operaciones petroleras para obtener una mayor información de los parámetros físicos y geológicos del pozo[4]. Estos registros brindan los datos necesarios y suficientes para detectar de antemano la ocurrencia de alguna complejidad y qué soluciones aplicar acorde a sus causas y características.

1.3 Proceso de manejo de archivos en sistemas para la perforación de pozos de petróleo.

Descripción General

El proceso de manejo de archivos para la perforación de pozos de petróleo brinda las funcionalidades de exportar e importar toda la información recopilada en forma de reportes y en diferentes formatos. Estos reportes son informes organizados que recogen toda la información necesaria de los pozos en un formato determinado y fácil de interpretar por los usuarios, estos pueden ser:

- Diario de geología.
- Diario de Perforación.
- Inclínometría Planificada.
- Inclínometría Real.
- Parte Operativo de Perforación.

La información contenida en cada uno de estos reportes, es exportada por:

- El supervisor: es el jefe del pozo, este es el rol más importante en el subsistema pozo, es el encargado de controlar o supervisar todos los procesos que se desarrollan en un pozo en perforación. Entre las responsabilidades principales está registrar en el sistema las herramientas

usadas, las necesidades y las actividades diarias en el pozo. Además de controlar el cambio de intervalos, los metrajes diarios y el sidetrack¹ en perforación.

- El geólogo: es el responsable de llevar el control de las propiedades del suelo. Entre sus funciones está registrar los topes y las bases en el sistema, además de la descripción litológica.
- La secretaria de despacho: se ocupa en el subsistema de la Dirección de Intervención y Perforación de Pozos (DIPP), de la generación de reportes y partes para los directivos.
- Los directivos: son los encargados de la toma de decisiones en el negocio según los datos que se recojan de los pozos. Entre sus responsabilidades en el sistema está obtener información a través de los reportes y las gráficas generadas por la aplicación.

En el negocio se genera un reporte diario de perforación, que es el más importante de los reportes ya que contiene la información más relevante del avance de la perforación del pozo. Además tiene en cuenta las actividades diarias y las necesidades que se generan durante el proceso.

1.3.1 Descripción actual del dominio del problema.

La Dirección de Intervención y Perforación de Pozos (DIPP), es la responsable de llevar un control diario de la perforación de pozos petroleros, documentando toda la información referente a los mismos, y las complicaciones que puedan surgir durante este proceso. Con el aumento de las perforaciones en Cuba, se han incrementado el número de complejidades presentes durante este proceso. En conjunto con la DIPP, el centro de investigaciones del petróleo (CEINPET) son los encargados de procesar y evaluar todo el flujo de información referente a los informes recibidos, los cuales brindan los datos necesarios y suficientes para detectar de antemano la ocurrencia de alguna complejidad y qué soluciones aplicar acorde a sus causas y características[6].

¹ Operación que se realiza en un pozo de perforación, cuando se presenta una complejidad técnica y/o geológica que imposibilita seguir avanzando por el hoyo perforado.

Actualmente la DIPP cuenta con un subsistema de manejo de archivos, que se encarga de exportar e importar reportes. Posee un software que le permite la elaboración de los reportes de manera automatizada, utilizando la versión 1.4.5 de la biblioteca PHPEXcel, la cual posee una licencia libre. La biblioteca no permite exportar en pdf, csv y html así como tampoco, trabajar con grandes volúmenes de datos. Por tal razón, se hace necesario buscar una nueva biblioteca que además de ejecutar estas funciones, evite dificultades cuando se exporte algún reporte.

El sistema cuenta con una vista destinada a la generación de archivos y en la misma se puede seleccionar el tipo de reporte que se desea visualizar en una fecha que establece el usuario con privilegios para realizar esa acción. Algunos reportes, como el reporte de combustible, productos químicos, record de barrenas, entre otros no se tuvieron en cuenta para ser exportados a los formatos que posibilita la aplicación. Agregar estos reportes facilitaría la toma de decisiones por parte de los directivos de la empresa. El código de la aplicación debe ser organizado y reutilizable, tener una documentación legible y fácil de entender por el lector. De esta forma, si fuese necesario hacer un mantenimiento a la aplicación se entendería a cabalidad todo el proceso de implementación del Módulo.

1.4 Análisis de soluciones existentes.

La existencia de compañías como Peloton e InfoOil, que desarrollan software para las empresas petroleras, ha posibilitado que se exporten archivos relacionados con el proceso de perforación de pozos en diferentes formatos. A continuación se presentan algunos productos en donde se evidencia este proceso:

WellView

WellView es un archivo corporativo del pozo. Desde la solicitud de perforar un pozo hasta el abandono, éste registra todos los cambios y operaciones a través del ciclo de vida del pozo. La base de datos del mencionado archivo recoge todos los datos relacionados con el pozo y sus operaciones así como del agujero y equipo de superficie. Es un sistema para manejar archivos electrónicos de pozos con un completo modelo de datos y anexos. Archivos tipo Excel, PDF, Word, imágenes, pueden ser anexados al

archivo del pozo, quedando almacenados dentro de la base de datos, dando la posibilidad de ser exportados e importados. Esto proporciona a las empresas una forma fácil de manejar todos los documentos incluyendo reportes diarios existentes en archivos Excel o documentos en papel digitalizados o escaneados. Es un software propietario y se sustenta en la plataforma Windows[7].

Target

Target es un software para la exploración, es propietario y se sustenta en la plataforma Windows. Ofrece una solución integral para proyectos de sondeo, desde los datos en bruto hasta la exploración minera. Target le proporciona una capacidad de levantamiento de planos y procesamiento de datos espaciales para trabajar con enormes volúmenes de datos que se utilizan para la exploración minera. Una herramienta de proyección de coordenadas asegura el soporte para una proyección total inmediata de todos los conjuntos de datos, permitiéndole combinar conjuntos de datos múltiples, con diferentes sistemas de coordenadas. Permite importar los datos de sus pozos de sondeo desde un amplio rango de fuentes de datos. [8].

MasterLog.

Software que permite generar registros de formación en blanco y negro o a todo color con un mínimo de instrucciones. El sistema emite un reporte final del pozo que contiene toda la información requerida por el cliente. Esta información será presentada en archivos digitales (Word, Excel, ASCII, PDF) y una copia en papel para su revisión y aprobación. Es un software propietario y se sustenta en la plataforma Windows. Todos los archivos gráficos están generados mediante el programa Adobe Acrobat. Todos los archivos generados con este programa tienen extensión *.PDF[9].

Strater.

STRATER es una herramienta para el registro de pozos y el trazado de perforaciones. Strater proporciona varias funcionalidades para simplificar la tarea de importar datos, crear el diseño de la perforación exacta que el usuario requiere y obtener la salida en el formato necesario, ya sea impreso o exportado a formato electrónico para incluir en un informe o presentación. Su base de datos soporta importación de datos de

muchas fuentes diferentes, incluyendo ficheros de texto ASCII, ficheros XLS y prácticamente cualquier fichero de base de datos. El usuario puede importar datos referentes a múltiples perforaciones en una tabla de datos e importar múltiples tablas de datos en un proyecto. Es un software propietario y se sustenta en la plataforma Windows. También puede crear múltiples vistas de perforaciones en un proyecto[10].

Wellflo.

El software de análisis de sistemas WellFlo es una aplicación para diseñar, modelar, optimizar e identificar problemas de pozos individuales de crudo y gas, ya sean naturalmente fluyentes o levantados artificialmente. Las herramientas de exportación de datos de WellFlo generan datos de tabla de flujo vertical en formato DOS o UNIX para su uso en simuladores de Eclipse y otros. Los archivos de reporte pueden ser generados en formato separado por tabulaciones para permitir una fácil exportación de los datos hacia paquetes de procesamiento de palabra y hoja de cálculo[11].

1.5 Comparación de soluciones existentes.

Durante la investigación realizada se pudo apreciar que las aplicaciones WellView, Target, MasterLog, Strater y WellFlo tienen gran competitividad a nivel mundial y han sido probadas por empresas con experticia, experiencia y prestigio en los procesos concernientes a la industria petrolera; mientras que el sistema desarrollado en la UCI tiene algunos problemas de rendimiento en el manejo de archivos, además de tener limitaciones en los formatos que este soporta al ser solamente Excel (.xls).

Estos sistemas que exportan e importan registros de pozos están protegidos bajo licencias propietarias, por lo que se dificulta el acceso al funcionamiento interno del software al analizar y reutilizar las funcionalidades que brindan. Además de obstaculizar la adaptación del software a las necesidades del subsistema que se va a desarrollar, por lo que implantar una de estas soluciones implicaría para el país una gran inversión de dinero en lo que concierne a la adquisición de sus licencias para mantenerlos actualizados. De igual modo habría que pagar las modificaciones que se le deberían hacer a los reportes para que se ajusten a las características del proceso y los informes en Cuba. Sin embargo el subsistema

de Manejo de Archivos desarrollado en la UCI es libre y se sustenta bajo las plataformas Windows y Linux. Por ende, es perceptible la necesidad de implementar un subsistema para exportar archivos en .xls, html, csv y pdf que se ajuste a los archivos que se manejan actualmente en el informe final de los pozos perforados en tierra. Además de exportar los reportes y partes anteriores e incorporar los nuevos que se realizan en la DIPP y los pozos durante la perforación. ¡Error! La autoreferencia al marcador no es válida.**Anexo 1:**

1.6 Caracterización de los nuevos registros de pozos en perforación.

En la perforación de pozos petroleros existen diferentes tipos de reportes donde se maneja información referente a los datos recolectados durante el proceso de perforación. En el Módulo de Manejo de Archivos de SIPP v1.0 se manejan diferentes tipos de reportes como el reporte diario de perforación, geología, operativo oficial e inclinometría tanto planificada, como real del pozo. Cada uno de estos reportes maneja información específica del proceso para conformar el informe final del pozo en perforación.

Actualmente no se manejan otros reportes en el Módulo de Archivos como son: el reporte de récord de barrenas, de combustible, de productos químicos y el cronograma real de perforación del pozo. A continuación se caracterizan cada uno de los reportes en cuestión:

Reporte de Récord de Barrenas: Informe en formato Excel que se emite con toda la información referida a la utilización de las barrenas durante la perforación del pozo. De cada una de las barrenas que se manejan por los intervalos de perforación, se recoge el tipo, nombre del fabricante, número de serie, los metros perforados, la rotación de la composición de herramientas, la presión a la que estuvo sometida la barrena, y en general el código de desgaste que tuvo durante el tiempo de utilización. El reporte tiene como objetivo final recoger todas las barrenas utilizadas en la perforación de pozos y los datos más significativos de las mismas en los intervalos de perforación.

Reporte de Combustible: Informe en formato Excel que contiene toda la información detallada de la utilización del combustible tanto en el campamento como en el equipo de perforación.

Reporte de Productos Químicos: Informe en formato Excel que abarca toda la información de los productos químicos usados en el pozo, donde se recogen los datos del producto, el precio, la cantidad usada, el costo diario y el costo total de productos.

Cronograma de la Perforación: Informe en formato Excel que contiene toda la información tanto del proyecto planificado como del proyecto real de la perforación. Recoge los días de perforación, los metros proyectados para perforar diariamente y la profundidad proyectada para cada día. También recoge los días reales de la perforación, los metros reales y la profundidad real del pozo de cada día.

Para comenzar un estudio más detallado del problema a resolver se hace necesario realizar un análisis de algunas de las herramientas, metodologías y tecnologías a utilizar tanto en el diseño como en la implementación del módulo. A continuación se hace un resumen de la investigación realizada, haciendo énfasis en los detalles más importantes de cada herramienta.

1.7 Metodología de desarrollo

Las metodologías de desarrollo son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de software. Las metodologías van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla[12]. Se pueden clasificar en dos grupos:

Metodologías Ágiles: Se encargan de valorar al individuo y las iteraciones del equipo más que a las herramientas o los procesos utilizados. Es más importante crear un producto de software que funcione que escribir mucha documentación. El cliente está en todo momento colaborando en el proyecto. Es más importante la capacidad de respuesta ante un cambio realizado que el seguimiento estricto de un plan.

- **Programación Extrema o XP (Extreme Programming)**

Es una metodología para el desarrollo de software y consiste básicamente en ajustarse estrictamente a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo. La Programación Extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software. Promueve el trabajo en equipo, preocupándose en todo momento del aprendizaje de los desarrolladores y estableciendo un buen clima de trabajo. Este tipo de método se basa en una realimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas y coraje para los múltiples cambios. Este tipo de programación es la adecuada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo[13].

- **SCRUM**

Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos: El desarrollo de software se realiza mediante iteraciones, denominadas sprints², con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración[13].

Metodologías Pesadas: Son las más tradicionales, se centran en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requiere una extensa documentación, ya que pretende prever todo de antemano. Este tipo de metodologías son más eficaces y necesarias cuanto mayor es el proyecto que se pretende realizar respecto a tiempo y recursos que son necesarios emplear, donde una gran organización es requerida[13].

² Período de tiempo durante el que se desarrolla un incremento de funcionalidad. Constituye el núcleo de **Scrum**, que divide de esta forma el desarrollo de un proyecto en un conjunto de pequeñas "carreras"

- **RUP (Rational Unified Process)**

Una de las metodologías pesadas más conocidas y utilizadas es la Metodología RUP que divide el desarrollo en 4 fases que definen su ciclo de vida:

- Inicio: El objetivo es determinar la visión del proyecto y definir lo que se desea realizar.
- Elaboración: Etapa en la que se determina la arquitectura óptima del proyecto.
- Construcción: Se obtiene la capacidad operacional inicial.
- Transmisión: Obtener el producto acabado y definido.

La metodología RUP tiene 6 principios clave:

- Adaptación del proceso: El proceso debe adaptarse a las características de la organización para la que se está desarrollando el software.
- Balancear prioridades: Debe encontrarse un balance que satisfaga a todos los inversores del proyecto.
- Colaboración entre equipos: Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados, etc.
- Demostrar valor iterativamente: Los proyectos se entregan, aunque sea de una forma interna, en etapas iteradas. En cada iteración se evaluará la calidad y estabilidad del producto y analizará la opinión y sugerencias de los inversores.
- Elevar el nivel de abstracción: Motivar el uso de conceptos reutilizables.
- Enfocarse en la calidad: La calidad del producto debe verificarse en cada aspecto de la producción[13].

Luego del estudio realizado se escoge como metodología a utilizar RUP. La misma dentro de sus fases, nos proporciona disciplinas en las cuales se encuentran guías para poder documentar e implementar de manera fácil y eficiente el desarrollo del software. Además es la metodología que actualmente utiliza el proyecto SIPP.

1.8 Herramientas y Tecnologías.

Las herramientas y tecnologías seleccionadas para el desarrollo de un proyecto deben ser escogidas cautelosamente. Pueden suponer el fracaso del proyecto o pueden aumentar su complejidad, para lo cual es necesario conocer cuáles son las distintas alternativas y las necesidades de nuestro proyecto.

1.8.1 Aplicación Web.

Una aplicación web es una aplicación informática distribuida cuya interfaz de usuario es accesible desde un cliente web, normalmente un navegador. Las aplicaciones web tienen grandes ventajas las cuales pueden ser aplicadas al máximo para darle solución a los problemas existentes en las Empresas de Petróleo, entre estas se encuentra la compatibilidad multiplataforma. Además están siempre actualizadas y son de fácil acceso. Tienen menos requerimientos de memoria y los datos son más seguros[14].

1.8.2 Lenguaje de Programación PHP 5

PHP es un lenguaje de secuencia de comandos para la creación dinámica de documentos HTML y de código abierto. Propicia una curva de aprendizaje fácil de recorrer, permitiendo el desarrollo rápido de aplicaciones web. Es un lenguaje de programación de páginas web del lado del servidor cuyas características principales son la independencia de plataforma y su gratuidad[15].

Características

PHP es un lenguaje multiplataforma. Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos. El código fuente escrito en PHP es invisible al navegador y al cliente, esto hace que la programación en PHP sea segura y confiable. Tiene capacidad de conexión con la mayoría de los gestores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos[15].

Se selecciona el lenguaje PHP, ya que ofrece interfaces para el acceso a la mayoría de las bases de datos comerciales y a todas las bases de datos posibles en sistemas Microsoft, a partir de las cuales podremos editar el contenido de nuestro sitio con absoluta sencillez. Es multiplataforma y permite aplicar

técnicas de programación orientada a objetos. Además en la versión 1.0 del módulo de manejo de archivos de SIPP, el lenguaje de programación utilizado es PHP, por lo que los desarrolladores tienen experiencia con el uso del mismo.

1.8.3 Framework de desarrollo

Un framework de desarrollo es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Es una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Entre los objetivos principales de un framework tenemos: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones[16].

1.8.3.1 Symfony 1.2.8

Symfony es un framework diseñado para optimizar el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. Symfony es compatible con la mayoría de los gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows[17].

Características de Symfony

Symfony es un framework independiente del sistema gestor de bases de datos y sigue la mayoría de las mejores prácticas y patrones de diseño para la web. Está preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa. Es un framework estable para desarrollar aplicaciones a largo plazo. Además el código es fácil de leer y de extender, lo que permite su integración con bibliotecas desarrolladas por terceros.

Symfony ofrece ventajas para el desarrollo de aplicaciones web ágiles entre estas se destaca que simplifica el desarrollo de las aplicaciones, haciéndolo más organizado y entendible. La estructura del código fuente es más legible. Encapsula operaciones complejas en instrucciones sencillas. Separa la lógica del negocio, la lógica del servidor y la presentación de la aplicación[17]. Por todo lo antes mencionado se selecciona Symfony como framework de desarrollo a utilizar.

1.8.4 Entorno Integrado de Desarrollo NetBeans 7

NetBeans es un IDE por sus siglas en inglés (*Integrated Development Environment*) multilenguaje completo y modular. Es gratuito, y de código abierto para desarrolladores de software. Es una plataforma para construir aplicaciones completas para el cliente. El IDE ha sido desarrollado para distintas plataformas como Linux, MacOS X, Solaris y también Windows[18].

NetBeans es el IDE seleccionado para el desarrollo de la aplicación, ya que está integrado con el framework Symfony y la integración con esta herramienta hace que se agilice más el desarrollo de las aplicaciones. Además NetBeans tiene mejoras en el editor de código. La instalación y actualización del IDE es más simple. Tiene integración con subversión para el control de versiones del código, características visuales para desarrollo web y soporte para PHP y JavaScript.

1.8.5 Sistema Gestor de Base de Datos.

Un Sistema Gestor de Base de Datos (SGBD): es un sistema de software que permite la definición de bases de datos; así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación. Un SGBD relacional es un modelo de datos que facilita a los usuarios describir los datos que serán almacenados en la base de datos junto con un grupo de operaciones para manejar los datos.

Los SGBD relacionales son una herramienta efectiva que permite a varios usuarios acceder a los datos al mismo tiempo. Brindan facilidades eficientes y un grupo de funciones con el objetivo de garantizar la confidencialidad, calidad, seguridad e integridad de los datos que contienen, así como un acceso fácil y eficiente a los mismos[19].

1.8.5.1 PostgreSQL 8.4

PostgreSQL es un Sistema Gestor de Bases de Datos Relacionales Orientadas a Objetos, derivado de Postgres. Es un gestor de bases de datos de código abierto, brinda un control de concurrencia multi-versión que permite trabajar con grandes volúmenes de datos; soporta parte de la sintaxis SQL y cuenta con un grupo de enlaces con lenguajes de programación[19].

PostgreSQL soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes y cadenas de bits. También permite la creación de tipos propios. Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos[20].

Se selecciona PostgreSQL como gestor de base de datos porque es un sistema estable y de alto rendimiento. Debido a la liberación de la licencia, PostgreSQL se puede usar, modificar y distribuir de forma gratuita para cualquier fin, ya sea privado, comercial o académico. Tiene gran flexibilidad ya que funciona en la mayoría de los sistemas Unix y se puede integrar al ambiente de Windows, permitiendo a los desarrolladores generar nuevas aplicaciones o mantener las existentes.

1.8.6 Servidor Web. Apache 2.1

Un servidor web puede ser tanto un ordenador de grandes dimensiones y capacidad como un programa informático que utiliza el protocolo de comunicaciones HTTP para recibir las peticiones de información de un programa cliente (navegador) en el ordenador del usuario. Apache es un servidor Web potente y flexible que funciona en distintas plataformas y entornos. Es una tecnología gratuita de código abierto. Trabaja con lenguajes como Perl y PHP. Es modular, puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, por lo que se han desarrollado diversas extensiones entre las que se destaca PHP, un lenguaje de programación del lado del servidor[21].

1.8.7 Bibliotecas para el manejo de archivos

PHPExcel 1.4.5:

Biblioteca que facilita el trabajo con el formato Excel. Permite agregar más hojas de cálculo (pestañas), insertar valores y fórmulas a celdas individuales, combinar celdas y proteger el rango de celdas con contraseña. Tiene opciones de salida en distintos formatos (Excel 2007, BIFF8 (Excel 97 y superior). Opciones de lectura en diferentes formatos (Excel 2007, BIFF5 (Excel 5.0 / Excel 95) y BIFF8 (Excel 97 o superior))[4].

PHP ExcelReader:

Biblioteca que permite la carga de ficheros de texto plano y hojas de cálculos, recupera información de ficheros .xls. Es una biblioteca que lee ficheros Excel (extensión xls) y permite acceder a la información y modificarla. Con esta biblioteca se puede modificar datos de celdas, filas y darle estilos[4].

PHPExcel 1.7.6

Esta versión de PHPExcel además de poseer las mismas características que la versión utilizada en la aplicación anterior, presenta mejoradas opciones de trabajo que facilitan el funcionamiento de la misma. Agrega la opción de escribir en formato pdf, brindando varias opciones para el trabajo con el mismo. PHPExcel implementa un cargador automático o "gestor de perezoso", lo que significa que no es necesario incluir todos los archivos dentro de PHPExcel, sólo los archivos que son requeridos por la secuencia de comandos se pueden cargar en la memoria de PHP. El principal beneficio de esto es que se reduce la huella de memoria de PHPExcel, para que utilice menos memoria de PHP. Excel2007 es el formato de archivo principal de PHPExcel. Permite la salida de la hoja de cálculo en la memoria en un archivo .xlsx. PHPExcel admite leer y escribir en archivos CSV (valores separados por comas). También permite escribir una hoja de cálculo en formato HTML, para la representación rápida de los datos en ella a cualquier persona que no tiene una hoja de cálculo en su computadora. Los lectores y los escritores son las herramientas que le permiten generar archivos de Excel a partir de plantillas. Esto requiere menos esfuerzo de codificación de generar el archivo de Excel a partir de cero, sobre todo si la plantilla tiene muchos estilos, las propiedades de configuración de página, encabezados, etc[22].

Requisitos del software:

El siguiente software es necesario para desarrollar el uso de PHPEXcel:

- Versión de PHP 5.2.0 o posterior.
- La extensión PHP `php_zip` activado.
- La extensión PHP `php_xml` habilitado.
- La extensión PHP `php_gd2` habilitado (si no está compilado)[22].

Tras el posterior análisis de las bibliotecas mencionadas para el trabajo con ficheros estándar, se propone PHPEXcel en su versión 1.7.6. Esta versión, reúne todas las especificaciones necesarias para que el Módulo de Manejo de Archivos sea implementado correctamente y cumpliendo con todos los requisitos establecidos por el cliente.

1.9 Conclusiones Parciales

En este capítulo se estableció que la aplicación va a ser desarrollada manteniendo como lenguaje de programación PHP. El entorno integrado de desarrollo que se propone es Netbeans 7 integrado con el framework Symfony en su versión 1.2.8, pues estas herramientas agilizan el desarrollo de aplicaciones web. La utilización de la biblioteca PHPEXcel en su versión 1.7.6, ayudará a solucionar los problemas presentes en el módulo y los requisitos exigidos por el cliente.

CAPÍTULO 2: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

2.1 Introducción

En el presente capítulo como paso previo al análisis y diseño se realiza el Modelo de Dominio, para capturar y enunciar la visión del proyecto. Se especifican los requisitos tanto funcionales como no funcionales de la aplicación. Los requisitos funcionales quedan agrupados en el diagrama de casos de uso del sistema. Se establece la arquitectura de la aplicación y los patrones de diseño utilizados en la codificación del proyecto. Las clases, con sus métodos y atributos, así como sus relaciones, se reflejan en el diagrama de clases del diseño.

2.2 Propuesta del sistema

Para el manejo eficaz de archivos en las empresas petroleras, se propone el desarrollo de un Módulo de Manejo de Archivos al cual los usuarios autorizados, previamente autenticados en el sistema, podrán acceder fácil e intuitivamente, mediante una interfaz sencilla. El mismo, posibilitará el exportado de los reportes del pozo y en diferentes formatos, ya sea xls, pdf, csv y html. El especialista al poseer información referente al pozo en perforación, actualizada y confiable, le será más sencillo la toma de decisiones. Se garantizará el acceso rápido y en el formato establecido por la empresa, de la información que se solicite.

2.3 Reglas del Negocio

Una Regla de Negocio define o limita un aspecto del negocio con el objetivo de establecer una estructura o un grado de influencia que condiciona el comportamiento de los actores del negocio. Por lo tanto, para exportar un reporte, este solamente puede ser exportado en los formatos, xls, pdf, csv y html.

2.4 Modelo de Dominio

Para desarrollar cualquier sistema, independientemente de su complejidad, es necesario dividirlo en secciones, estas se pueden representar mediante modelos que permitan abstraer sus características

esenciales. Como paso previo al análisis y diseño del sistema es preciso realizar un Modelo de Negocio o Modelo de Dominio para capturar y enunciar la visión del proceso, encaminado a satisfacer las necesidades durante estas fases. (Pressman, 2001)

Cuando no existen procesos definidos, RUP propone realizar un modelo de dominio, que es un subconjunto del modelo de negocio. Un modelo de dominio recoge los tipos de objetos más importantes que existen en el sistema, y tiene por objetivo ayudar a comprender mejor los conceptos que utilizan los usuarios, con los que se trabaja. En el siguiente Modelo de Dominio se especifican las relaciones que existen entre los principales conceptos que interactúan en el sistema:

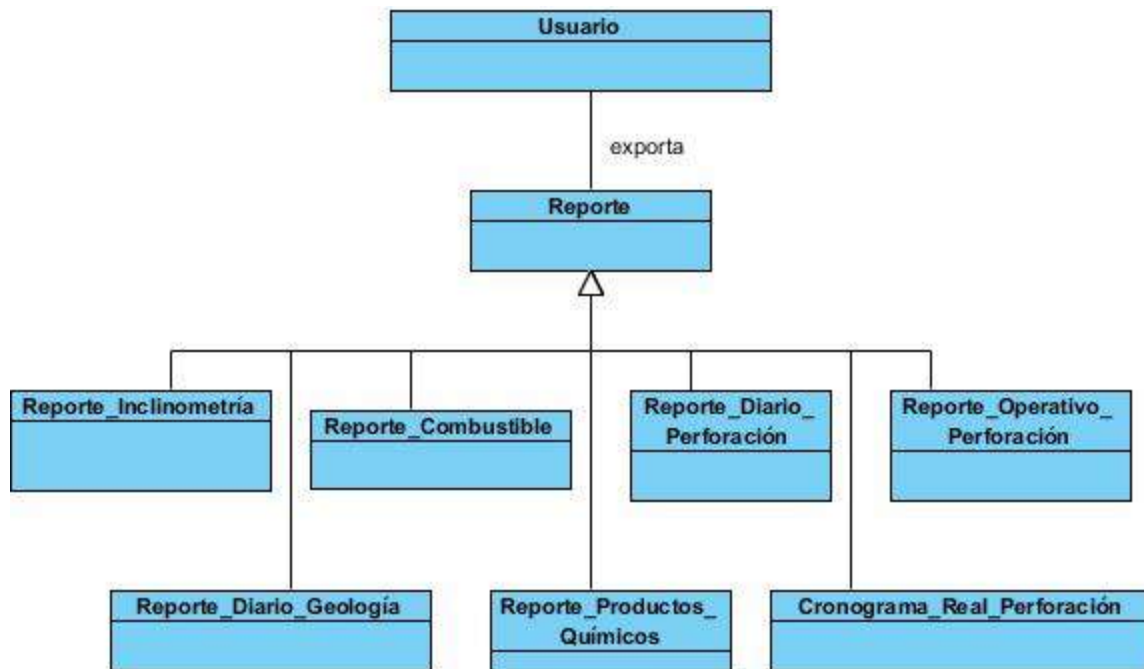


Figura 1 Modelo de Dominio

2.5 Requisitos

La Ingeniería de Requisitos ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del

software sobre el negocio, qué es lo que el cliente quiere y cómo interactúan los usuarios finales con el software. (Pressman, 2006: 155)

2.5.1 Requisitos Funcionales

Los requisitos funcionales son las capacidades o condiciones que el sistema debe cumplir. A continuación se establecen los requisitos funcionales para el Módulo de Manejo de Archivos:

RF1: Exportar ficheros en formato CSV, HTML, Excel y PDF.

RF2: Seleccionar ubicación para exportar el fichero.

RF3: Seleccionar el tipo de formato.

RF4: Seleccionar el tipo de reporte.

2.5.2 Requisitos No Funcionales

Los requisitos no funcionales son las cualidades que el producto debe tener, con el objetivo de lograr que este sea fiable y seguro. A continuación se relacionan un conjunto de cualidades establecidas por el cliente:

RNF: Seguridad

- **Confidencialidad:** La información manejada por el sistema está protegida de acceso no autorizado y divulgación.
- **Disponibilidad:** Los usuarios autorizados (según su rol) se les garantizará el acceso a la información, los dispositivos o mecanismos utilizados para lograr la seguridad, no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.

RNF: Rendimiento

- El sistema debe responder en un tiempo relativamente rápido a las peticiones del usuario (6 segundos).

RNF: Interfaz

- Interfaces amigables, fáciles de interactuar con ellas.

RNF: Hardware (Servidos Web)

- Memoria RAM ³_{>=} 4 GB.

RNF: Hardware (Cliente Web)

- Procesador Pentium 3 (o superior).
- Memoria RAM mínima de 256 MB.

RNF: Portabilidad.

Será un sistema multiplataforma, se podrá disponer del mismo tanto en el sistema operativo Linux como Windows.

2.6 Modelo de caso de uso

El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema. Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. Un caso de uso es una unidad de trabajo significativo, puede 'incluir' la funcionalidad de otro caso de uso o puede 'extender' otro caso de uso con su propio comportamiento. Cada caso de uso tiene una descripción que especifica la funcionalidad que se incorporará al sistema propuesto. Los casos de uso típicamente se

³ Random Access Memory o Memoria de Acceso Aleatoria

relacionan con 'actores'. Un actor es un humano o una máquina que interactúa con el sistema para realizar un trabajo significativo[23].

2.6.1 Diagrama de caso de uso del sistema

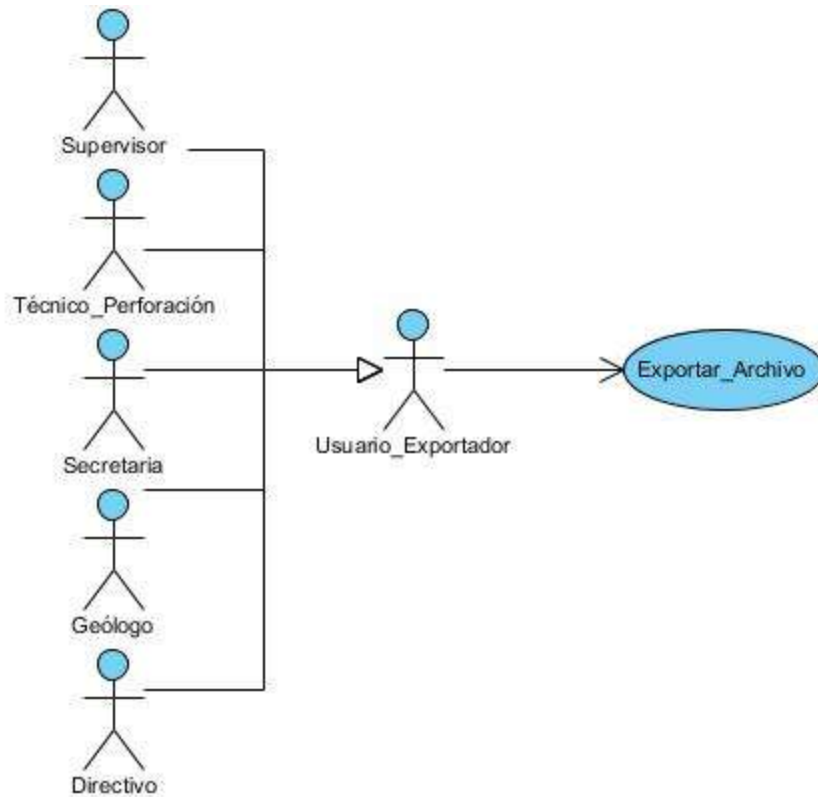


Figura 2 Diagrama de Caso de Uso del Sistema

2.6.2 Actores del Sistema

Un actor es un usuario del sistema. Esto incluye usuarios humanos y otros sistemas computacionales. Un actor usa un caso de uso para ejecutar una porción de trabajo de valor para el negocio. El conjunto de casos de uso al que un actor tiene acceso define el rol en el sistema y el alcance de su acción[23].

Capítulo 2: Descripción de la solución propuesta

2.6.2.1 Descripción de los Actores

Actor	Descripción
Geólogo del CEINPET	Encargado de elaborar el Reporte Diario de Geología, a partir de la información enviada por los geólogos del pozo. Este reporte es enviado posteriormente a la DIPP de perforación y a CUPET.
Secretaria de Despacho	Encargada de la elaboración de los reportes en la oficina de Despacho de la DIPP, a partir de la información enviada de los pozos de petróleo, por lo cual se convierte en una figura clave en este proceso.
Supervisor del Pozo	Este actor proviene del trabajador del negocio del mismo nombre. Interactúa principalmente con los casos de Uso del Módulo Supervisor. Este Módulo pertenece al Subsistema Pozo.
Directivo	Este actor representa a los directivos de la DIPP, consultando toda la información referente a los pozos así como iniciar la perforación de un Pozo.
Técnico en Perforación	Este actor es el encargado de dar inicio a los pozos (Gestionar Pozo), cambiarle el estado, así como consultar información de los pozos en perforación.
Supervisor Jefe del Pozo	Este actor aglutina las funciones de los actores Supervisor del Pozo y Administrador del Subsistema Pozo.

Tabla 1 Descripción de los Actores del Sistema

2.6.3 Descripción de los Casos de Uso

Exportar Archivos

Caso de Uso:	Exportar Archivo
Actores:	Usuario
Resumen:	El caso de uso se inicia cuando el actor necesita exportar cualquier información persistente referente a los pozos, ya sea de los diferentes reportes y cronogramas que se generan.
Precondiciones:	Usuarios del Sistema ya Autenticados.
Referencias	RF1, RF2, RF3, RF4
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Buscar Información Específica”	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la Opción “Exportar Archivo”.	1.1 El sistema muestra una interfaz con un formulario para seleccionar el nombre del archivo y la extensión con la que será salvado.
2. El usuario selecciona la extensión y el tipo de reporte.	2.1 El sistema muestra un formulario para insertar la fecha, en caso de que el reporte seleccionado sea alguno de los que a continuación se mencionan:
3. El usuario selecciona la fecha.	<ul style="list-style-type: none"> • Reporte Diario de Geología. • Reporte Diario de Perforación. • Reporte Operativo Oficial de Perforación. • Reporte Operativo de Perforación de las 4 pm. • Reporte de Productos Químicos.

Capítulo 2: Descripción de la solución propuesta

4. El usuario selecciona la opción de exportar.	4.1 El sistema muestra un cuadro de diálogo para seleccionar la ubicación para exportar el archivo.
5. El usuario selecciona la ubicación para guardar el archivo y acepta.	5.1 El sistema exporta el archivo y muestra un mensaje del éxito de la operación.
Flujos Alternos	
Acción del Actor	Acción del Sistema
3a.1 El usuario selecciona la opción de exportar.	3a.1 El sistema muestra un cuadro de diálogo para seleccionar la ubicación para exportar el archivo.
4a.1 El usuario selecciona la opción para abrir el archivo y acepta.	4a.2 El sistema muestra el reporte en el formato seleccionado.
7a.1 El usuario selecciona la opción cancelar.	7a.2 El sistema cierra el cuadro de diálogo exportar.
Prototipo de Interfaz	



The screenshot displays the MaximusDrillPro software interface. At the top, there is a green header with the logo and the text 'MaximusDrillPro'. Below the header is a navigation menu with the following items: Inicio, Inventarios, Combustible, Herramientas, Procesos Diarios, Fluido, Otros Procesos, Reportes, Gráficas, Archivo, and Módulos. A red button labeled 'Exportar a Archivos' is prominently displayed. Below this button is a dialog box titled 'Selecciones:' with the following fields: 'Formato: Excel2007', 'Reporte: Reporte Diario de Perforación', and a date field '2012-04-06'. An 'Exportar' button is located at the bottom of the dialog box. Below the screenshot, a table provides the conditions for the export function.

Pos condiciones	Se exporta la información solicitada.
------------------------	---------------------------------------

Tabla 2 Caso de Uso Exportar Archivo

2.7 Diseño

El diseño del software es un proceso de muchos pasos pero que se clasifican dentro de uno mismo. En general, la actividad del diseño se refiere al establecimiento de las estructuras de datos, la arquitectura general del software, representaciones de interfaz y algoritmos. El proceso de diseño traduce requisitos en una representación de software. El objetivo del diseño es producir un modelo o representación de una entidad que se va a construir posteriormente. (Pressman 98)

2.8 Diagrama de clases del diseño

Durante el diseño, el Diagrama de Clases se elabora para tener en cuenta los detalles concretos de la implementación del sistema. Este describe gráficamente las especificaciones de las clases del software y las interfaces en una aplicación, conteniendo definiciones de las entidades del software en vez de

Capítulo 2: Descripción de la solución propuesta

conceptos del mundo real. Su uso expresa la definición de las clases como componentes del software. (Pressman, 2001)

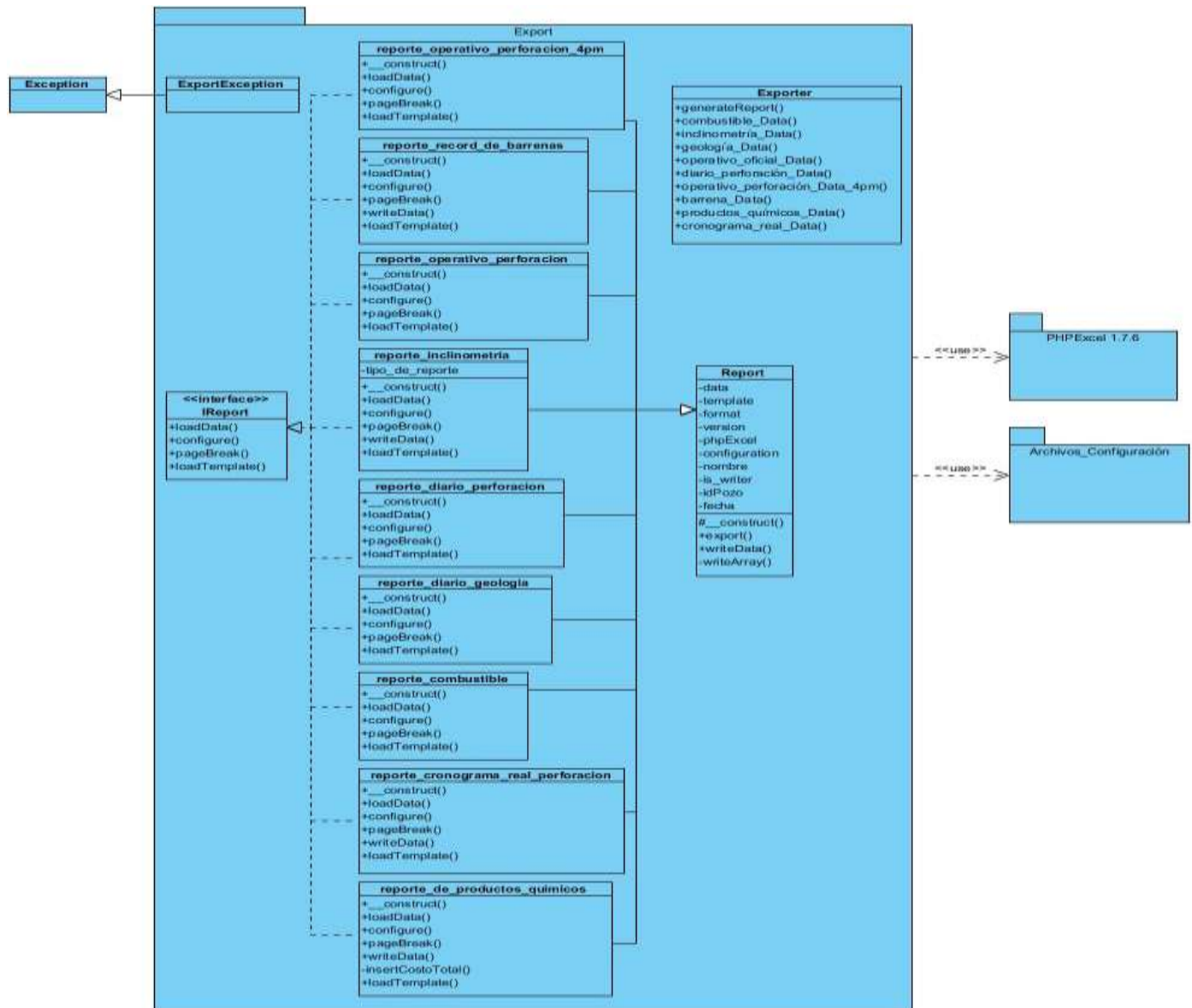


Figura 3 Diagrama de clases del diseño exportar archivo

2.8.1 Descripción de las clases del diseño

IReport: funciona como interfaz, son declaraciones de funcionalidades implementadas en las clases de cada reporte y define los métodos que tienen en común para estandarizar el exportado de los mismos.

Report: funciona como la clase padre de cada reporte que se maneja en el módulo. Esta se encarga de exportar cada reporte y maneja las excepciones en caso de algún error.

Exporter: La clase **Exporter** es la encargada de retornar el objeto que manipula el reporte seleccionado y posee métodos responsables de retornar los datos para cada reporte en el formato esperado por cada clase.

PHPExcel: Biblioteca de PHP que brinda API⁴'s para el trabajo con ficheros estándar como Excel, PDF, CSV y HTML.

Archivos de Configuración: Paquete de archivos de configuración con extensión .yml que contiene la configuración de los reportes.

ExportException: La clase **ExportException** hereda de la clase **Exception** del núcleo de PHP. Su principal función es ser lanzada cuando una condición anormal ocurre en el proceso.

ReporteDiarioPerforación: La clase **ReporteDiarioPerforación** es la encargada de obtener los datos del reporte diario de perforación y personalizarlos según el modelo de la plantilla para poder ser exportados.

ReporteDiarioGeología: La clase **ReporteDiarioGeología** es la encargada de obtener los datos del reporte diario de geología y personalizarlos según el modelo de la plantilla para poder ser exportados.

⁴ Application Programming Interface o Interfaz de programación de aplicaciones, son el conjunto de funciones y procedimientos que ofrece una biblioteca para ser como una capa de abstracción.

ReporteRecorddeBarrenas: La clase **ReporteRecorddeBarrenas** es la encargada de obtener los datos del reporte de record de barrenas y personalizarlos según el modelo de la plantilla para poder ser exportados.

ReporteInclinometría: La clase **ReporteInclinometría** es la encargada de obtener los datos del reporte de inclinometría, seleccionar el tipo de inclinometría y personalizarlos según el modelo de la plantilla para poder ser exportados.

ReporteProductosQuímicos: La clase **ReporteProductosQuímicos** es la encargada de obtener los datos del reporte de productos químicos y personalizarlos según el modelo de la plantilla para poder ser exportados.

2.9 Arquitectura

La Arquitectura es el esqueleto o base de una aplicación, en esta se analiza la aplicación desde varios puntos de vista. En RUP, esta se compone de 4+1 vistas: vista lógica, vista de procesos, vista de implementación, vista de despliegue y estas cuatro regidas por la vista de casos de uso[24]. De acuerdo con Pressman, en su forma más simple, la arquitectura es la estructura jerárquica de los módulos del programa, la manera de interactuar de estos componentes, y la estructura de los datos usados por estos módulos. (Pressman 98)

Existen muchos patrones arquitectónicos, específicamente para darle solución al problema existente en la investigación utilizamos el MVC que son las siglas de modelo-vista-controlador (*o en inglés, model-view-controller*). El desarrollo de aplicaciones web con symfony está basado en el patrón arquitectónico MVC y este a su vez está formado por 3 niveles. El modelo representa la información con la que trabaja la aplicación, es decir su lógica de negocio, las reglas, restricciones y condiciones definidas para la operación de la aplicación. La vista transforma el modelo en una página web que permite al usuario interactuar con ella. Por último, el controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista[25].

Capítulo 2: Descripción de la solución propuesta

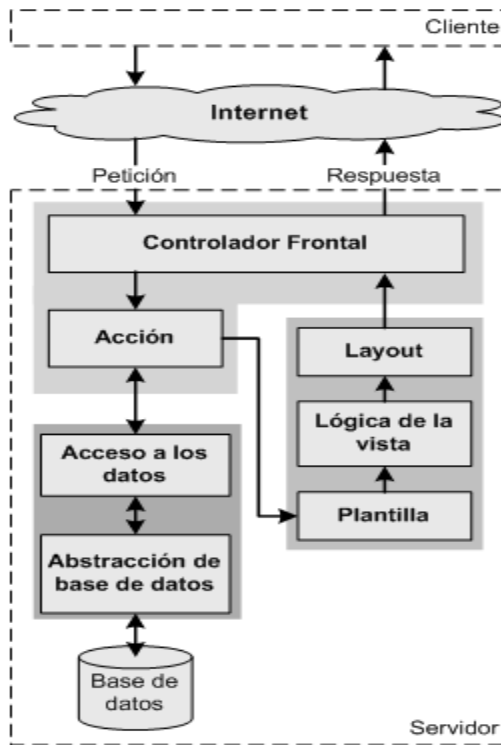


Figura 4 Arquitectura MVC

Symfony toma lo mejor de la arquitectura MVC y lo implementa de forma tal que la creación de aplicaciones sea lo más sencilla posible. El framework Symfony crea un controlador frontal y un layout comunes para cada aplicación. El controlador frontal es el único punto de entrada de la aplicación, es el encargado de cargar la configuración y determina la acción que debe ejecutarse[26]. Se crea un controlador para cada acción. El mismo es el encargado de establecer el orden lógico de las operaciones, y modificar el modelo o la vista en caso que fuera necesario. Las vistas de la aplicación estarán agrupadas en la carpeta templates de cada módulo. Para hacer reusable y más fácil de entender el código, Symfony cuenta con técnicas, como los slot y componentes, los cuales mejoran el rendimiento de la aplicación. Las clases necesarias para manipular los datos y encapsular la lógica de la aplicación son generadas por la biblioteca Propel. Estas contienen un conjunto de métodos comunes para el trabajo con la base de datos.

Capítulo 2: Descripción de la solución propuesta

Las bibliotecas usadas en cualquier proyecto symfony se encuentran en las carpetas *lib*. Symfony es capaz de reconocer y cargar automáticamente todas las clases, permitiendo su posterior uso. El módulo para el Manejo de Archivos del proyecto SIPP, se encontrará ubicado en las *lib* de la aplicación. El mismo contará con un núcleo, en el cual se encuentran las clases imprescindibles para su funcionamiento, **Report**, **IReport**, **ExportException**. El resto de las clases heredan e implementan, de las clases del núcleo, las funciones necesarias para realizar el reporte. El módulo también cuenta con las plantillas Excel de cada reporte establecida por el cliente en la carpeta **template**, así como los archivos de configuración en formato **YAML**⁵ en la carpeta **config**, para cada uno de los reportes.

La clase **sfYaml**, del núcleo de Symfony, es la encargada de manipular los archivos de configuración. Estos definen los datos que va a tener cada celda. De esta manera se facilita la configuración y escalabilidad del módulo. La ventaja fundamental que posibilitan estos archivos de configuración, es que cambios sencillos en el formato de las plantillas de cada reporte establecido por el cliente, no implica modificaciones en el código fuente.

Los datos, extraídos del modelo, son representados en las plantillas de dos formas: pueden ser **cíclicos** y **no cíclicos**. La primera se establece cuando el área destinada para representar los datos está conformada por varias filas y columnas, las cuales se repiten **n** veces.

El diagrama muestra una tabla con un encabezado de dos celdas: 'Última muestra descrita a la profundidad:' y 'm'. La primera fila de datos tiene seis columnas con encabezados: 'Litotipo', '%', 'Litotipo', '%', 'Litotipo' y '%'. Una flecha roja horizontal indica la repetición de la primera fila. Una flecha roja vertical indica la repetición de las primeras dos columnas. Tres corchetes rojos debajo de la tabla agrupan las columnas en tres bloques de dos columnas cada uno, correspondientes a los encabezados de 'Litotipo' y '%'.

Última muestra descrita a la profundidad:				m	
Litotipo	%	Litotipo	%	Litotipo	%

Figura 5 Ejemplo de datos representados de forma cíclica

⁵Yet Another Markup Language u Otro Lenguaje de Mercado. Dado que XML (**eXtensible Markup Language**) se usa frecuentemente para serializar datos, YAML puede ser considerado como un lenguaje de marcado sencillo y fácil de interpretar.


```
break_page_column: H1
nombre_pozo: A2
begin_row: 5
dias_proyecto: A
m_proyecto: B
h_proyecto: C
dias_real: D
m_real: E
h_real: F
fecha: G
```

Figura 8 Ejemplo de archivo de configuración para datos en forma no cíclica.

2.10 Patrones de Diseño

Un Patrón de Diseño (*design pattern*) es una solución repetible a un problema recurrente en el diseño de software. Esta solución no es un diseño terminado que puede traducirse directamente a código, sino más bien una descripción sobre cómo resolver el problema, la cual puede ser utilizada en diversas situaciones. Los patrones de diseño reflejan todo el rediseño y re-modificación que los desarrolladores han ido haciendo a medida que intentan conseguir mayor reutilización y flexibilidad en el software[27].

Creador: Indica quién debe ser el responsable de la creación de una nueva instancia de alguna clase. Su uso en el desarrollo del módulo brinda la posibilidad de mantenimiento, mayor claridad y reutilización del código.

En la aplicación, la clase Exporter es la encargada de crear las instancias de las clases según el tipo de reporte, lo que evidencia el uso de este patrón. En la figura se presenta un ejemplo del patrón Creador en el sistema:

Capítulo 2: Descripción de la solución propuesta

```
class Exporter {
    /**
     * Genera los reportes.
     *
     * @param array $param Array de asociativo array('idPozo', 'fecha', 'tipoReporte', 'formato', 'version', 'tipo')
     * @return Report
     * @throws ExportException, Exception
     */
    public static function generateReport($param = array()){
        $obj = null;

        switch ($param['tipoReporte']){
            case 'geologia':
                $obj = new reporte_diario_geologia($param['idPozo'], $param['fecha'], $param['formato'], $param['
                break;
            case 'diario':
                $obj = new reporte_diario_perforacion($param['idPozo'], $param['fecha'], $param['formato'], $para
                break;
            case 'combustible':
                $obj = new reporte_combustible($param['idPozo'], $param['fecha'], $param['formato'], $param['ver
                break;
            case 'inclinometria':
                $obj = new reporte_inclinometria($param['idPozo'], $param['tipo'], $param['formato'], $param['ver
                break;
            case 'barrenas':
                $obj = new reporte_record_de_barrenas($param['idPozo'], $param['formato'], $param['version']);
                break;
            case 'quimicos':
                $obj = new reporte_de_productos_quimicos($param['idPozo'], $param['fecha'], $param['formato'], $p
                break;
            case 'real_perforacion':
                $obj = new reporte_cronograma_real_perforacion($param['idPozo'], $param['formato'], $param['ver
                break;
        }
    }
}
```

Figura 9 Patrón Creador en el Sistema

Alta Cohesión: La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme[28]. El uso de este patrón beneficia en gran medida el módulo, ya que mejora la claridad y facilidad con que se entiende el diseño.

En el módulo se evidencia este patrón en la clase Report, pues no tiene muchos métodos relacionados funcionalmente, no realiza mucho trabajo y colabora con otras clases. A continuación se muestra un ejemplo del patrón en el sistema:

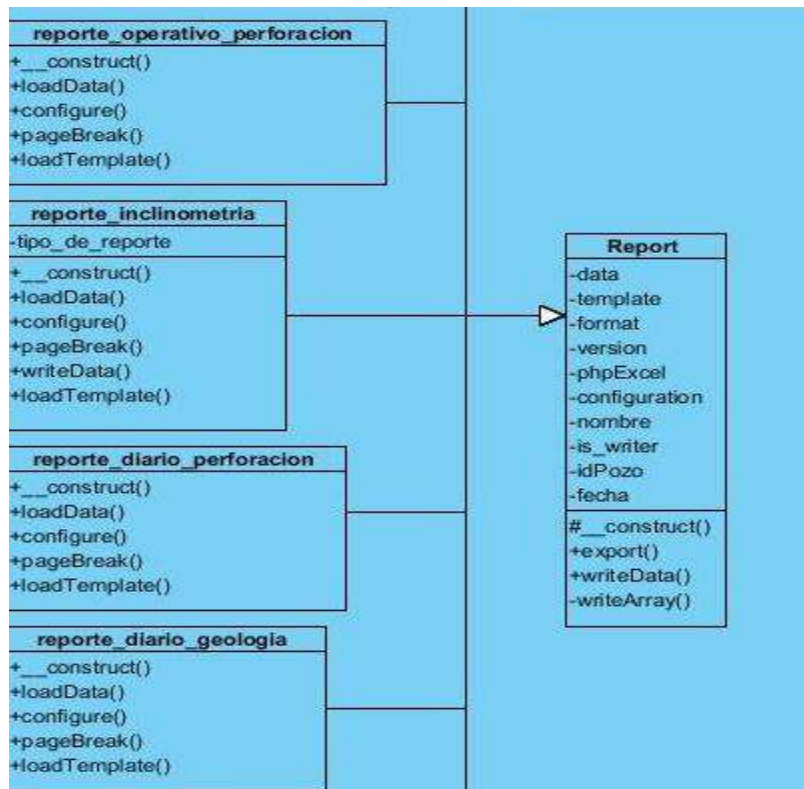


Figura 10 Patrón de Alta Cohesión en el sistema

Bajo Acoplamiento: El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases y con que recurre a ellas. Acoplamiento bajo significa que una clase no depende de muchas clases[28]. El uso de este patrón trae grandes beneficios para el desarrollo de este módulo, pues facilita la reutilización y entendimiento del código.

En el módulo se evidencia este patrón en las clases de cada reporte, ya que estos no están relacionados con el modelo de datos. Cada clase reporte percibe la función que debe hacer sin depender del resto de las clases. A continuación se muestra un ejemplo del patrón Bajo Acoplamiento en el sistema:

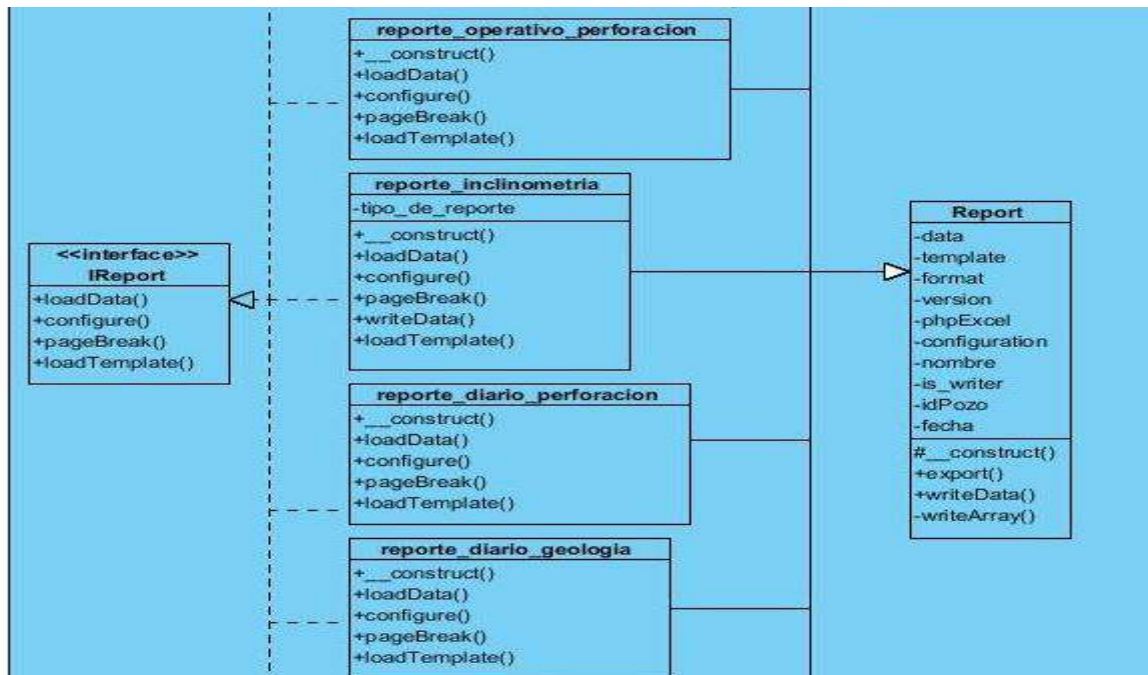


Figura 11 Patrón de Bajo Acoplamiento en el sistema

Experto: En el diseño orientado a objetos el patrón Experto es muy importante, ya que se usa más que cualquier otro, para asignar responsabilidades. La utilización de este patrón en el módulo trae muchos beneficios para la aplicación, pues se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece al hecho de tener sistemas más robustos y de fácil mantenimiento.

En el desarrollo de la aplicación se evidencia este patrón en la clase de cada reporte, pues cada uno sabe cómo manejar sus atributos de manera independiente. A continuación se muestra un ejemplo del patrón Experto en el sistema:

Capítulo 2: Descripción de la solución propuesta

```
39 public function writeData() {
40     $this->phpExcel->setActiveSheetIndex(0);
41     $row = $this->configuration["begin_row"];
42
43     foreach($this->data as $key1 => $value1):
44         if(!is_array($value1) && $key1 && $key1 == 'nombre_pozo'):
45             $this->phpExcel->getActiveSheet()->setCellValue($this->configuration[$key1], $value1);
46             continue;
47         endif;
48         foreach ($value1 as $value):
49             foreach ($value as $key => $v):
50                 $this->phpExcel->getActiveSheet()->setCellValue($this->configuration[$key].$row, $v);
51             endforeach;
52             $row++;
53         endforeach;
54         $row = $this->configuration["begin_row"]; //ponerlo en la posición inicial para que cuando le toque al o
55     endforeach;
56
57     $this->phpExcel->setActiveSheetIndex(0);
58     $this->is_writer = true;
59 }
```

Figura 12 Patrón Experto en el sistema

Controlador: El patrón controlador es el encargado de manejar un evento en el sistema. Este patrón en el módulo se evidencia en la clase ExportarActions, pues se encarga de manejar todos los parámetros que se le pasan a la clase generar reporte, además de tratar los errores, verificar que el usuario está autorizado a ejecutar la acción y establecer el orden lógico en que se van a realizar las acciones. A continuación se muestra un ejemplo del patrón Controlador en el sistema:

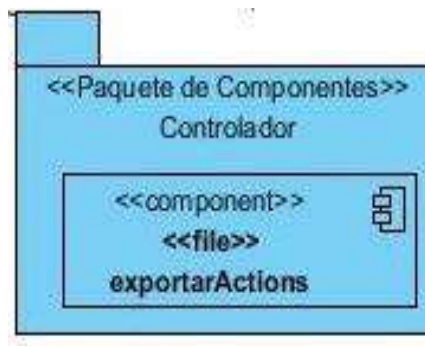


Figura 13 Patrón Controlador en el sistema

2.11 Conclusiones Parciales

En este capítulo quedan definidos los principales artefactos de la disciplina de análisis y diseño, ya que se tienen las primeras propuestas de clases que se encargan de garantizar los distintos servicios respondiendo a los requisitos del cliente. Se obtuvo el diagrama de clases del diseño, teniendo estructurada la arquitectura del sistema. Se aplican los patrones de diseño para obtener una visión detallada de la implementación del módulo.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA

3.1 Introducción

En este capítulo se implementan los elementos del Modelo de Diseño en forma de componentes, asigna los componentes a los nodos y los integra a un sistema ejecutable. Utiliza el Modelo de Implementación, conjuntamente los Diagramas de Componentes para comprender cómo se organizan los Componentes y dependen unos de otros. Se hace uso del diagrama de despliegue para asegurar que el producto está preparado para el cliente. Por último se muestran todas las pruebas realizadas a las funcionalidades necesarias.

3.2 Implementación

Durante la implementación del sistema se define la organización del código. Se implementan los elementos del diseño, se integran los resultados producidos en un sistema ejecutable y se distribuyen físicamente asignando componentes ejecutables a nodos.

3.2.1 Diagrama de componentes

En este diagrama se muestran todos los componentes requeridos para la implementación del sistema. El diagrama muestra todos los componentes del sistema, estos se encuentran encapsulados en paquetes lo que agiliza la navegación a través del software. Describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable.

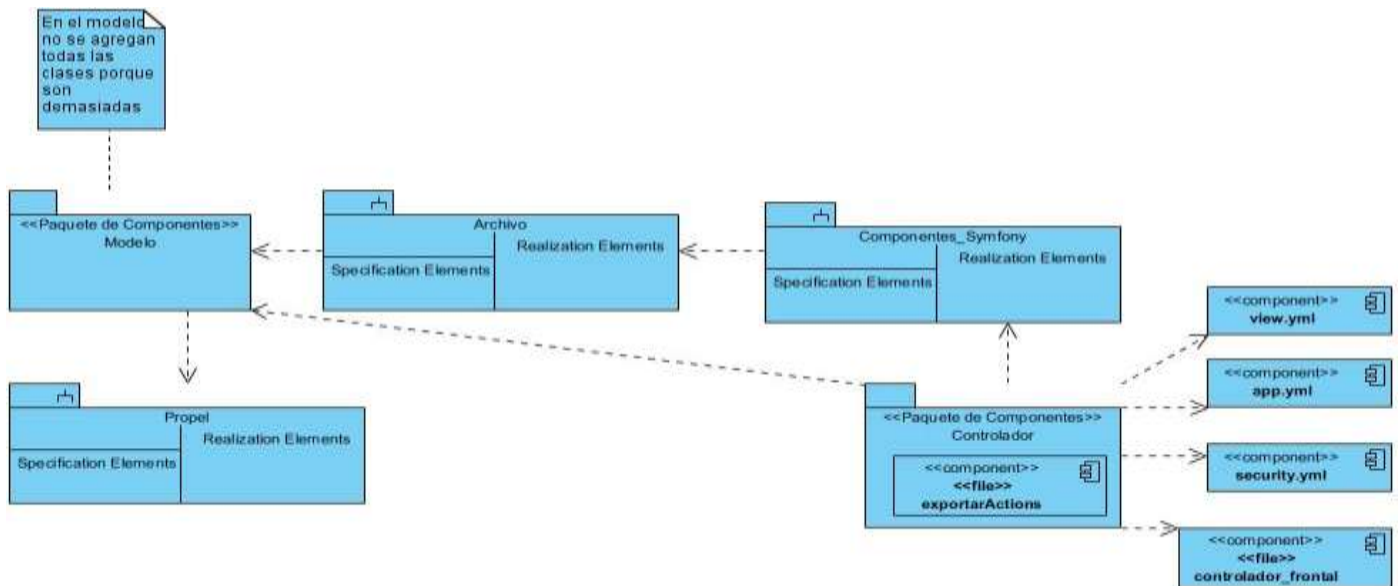


Figura 14 Diagrama de Componentes. Caso de uso Exportar

3.2.2 Diagrama de despliegue

El diagrama de despliegue permite modelar la disposición física o topología de un sistema. Muestra el hardware usado y los componentes instalados en el hardware, además de las conexiones físicas entre este y las relaciones entre componentes. Es utilizado para capturar los elementos de configuración del procesamiento y las conexiones entre esos elementos. También se utiliza para visualizar la distribución de los componentes de software en los nodos físicos. El mismo está compuesto por: nodos, dispositivos y conectores[23].

El diagrama de despliegue propuesto está compuesto por 3 nodos: la PC_Cliente, el Servidor Web Apache y el Servidor de base de datos PostgreSQL. Los protocolos de comunicación entre los nodos serán: HTTP entre la PC_Cliente y el servidor de aplicaciones y el protocolo TCP/IP entre los dos servidores, el de aplicaciones y el servidor de base de datos. A continuación se presenta el diagrama de despliegue del sistema:

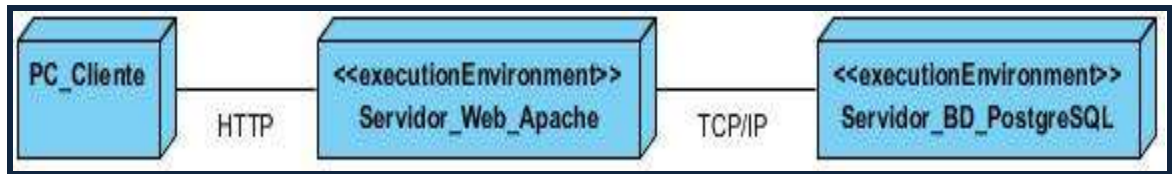


Figura 15 Diagrama de Despliegue

3.3 Pruebas de software

Las pruebas de software son para comprobar el resultado de la implementación de los componentes. Su objetivo fundamental es validar el software, con la intención de encontrar errores, para luego depurarlo y obtener como resultado un software de calidad.

3.3.1 Modelo de prueba. Funcionalidad

Entre las pruebas existentes para comprobar el cumplimiento de los requerimientos, se encuentra la prueba por funcionalidad. Esta pretende evaluar el cumplimiento de los requisitos funcionales, incluye la navegación, entrada de los datos, procesamiento y la obtención de resultados. Se ejecuta en cada funcionalidad de caso de uso, corroborando la implementación adecuada de las reglas del negocio. Su uso permite conocer si mediante la introducción de datos válidos, se obtienen los resultados esperados, además de que sean exhibidos los mensajes apropiados de error y precaución cuando se usan datos inválidos. (pruebasdesoftware, 2011)

Descripción de los casos de prueba

Los casos de pruebas son para probar detalladamente el sistema, todas las posibles entradas que debe tener, en diferentes circunstancias y los resultados esperados. Para comprobar el funcionamiento del Módulo de Manejo de Archivos se precisaron un conjunto de casos de pruebas que agrupan todas las situaciones posibles. Las pruebas han sido realizadas sobre el propio sistema de manejo de archivos, a través de la técnica de partición de equivalencia.

Modelo de Pruebas

Capítulo 3: Implementación y prueba del sistema

Caso de Uso	Variable 1	Variable 2	Variable 3	Ejecutar Acción	Resultado Esperado	Resultado Obtenido
Exportar Archivo	Formato xlsx	Tipo de Reporte Reporte de Inclinometría Real	Fecha (No tiene)	Selecciona la opción Aceptar y busca ubicación para guardar el archivo.	Mensaje de éxito de la operación.	Mensaje de éxito de la operación.
	xlsx	Reporte de Inclinometría Real	(No tiene)	Selecciona la opción Abrir. Anexo 2	Se muestra el archivo en formato xlsx.	Se muestra el archivo en formato xlsx. Anexo 3
	html	Reporte de Inclinometría Real	(No tiene)	Selecciona la opción Cancelar.	Se cierra el cuadro de diálogo exportar.	Se cierra el cuadro de diálogo exportar.
	csv	Reporte de Inclinometría Real	(No tiene)	Selecciona la opción Aceptar y busca ubicación para guardar el archivo.	Mensaje de éxito de la operación.	Mensaje de éxito de la operación.
	pdf	Reporte de Combustible	Selecciona la fecha	Selecciona la opción Abrir. Anexo 4	Se muestra el archivo en formato pdf.	Se muestra el archivo en formato pdf. Anexo 5

	HTML	Reporte Operativo de Perforación	Selecciona la fecha	Selecciona la opción Abrir. Anexo 6	Se muestra el archivo en formato HTML.	Se muestra el archivo en formato HTML. Anexo 7
	CSV	Reporte de Productos Químicos	Selecciona la fecha	Selecciona la opción Abrir. Anexo 8	Se muestra el archivo en formato CSV.	Se muestra el archivo en formato CSV. Anexo 9

Tabla 3 Pruebas de funcionalidad al sistema

3.3.2 Pruebas de rendimiento

Pruebas dirigidas a evaluar la conformidad del sistema con requisitos de desempeño específicos. A continuación se muestra una tabla que muestra el tiempo de ejecución y la memoria usada al exportar un reporte.

Módulo de Manejo de Archivos				
Nombre del Reporte	Versión 1.0		Versión 2.0	
	Tiempo de Ejecución	Memoria Usada	Tiempo de Ejecución	Memoria Usada
Inclinometría Plan	12.9s	38.45 MB	9s	15.23 MB
Inclinometría Real	12.9 s	40.25 MB	10s	17.23 MB
Diario de Geología	10.09 s	48.5 MB	7s	12.23 MB
Diario de	13.9 s	70.25 MB	9s	25.23 MB

Capítulo 3: Implementación y prueba del sistema

Perforación				
Operativo Oficial de Perforación	12.9 s	60.30 MB	8s	20.23 MB
Operativo de Perforación 4 pm	11.9 s	34.25 MB	7s	14.25 MB
Record de Barrenas	No soportado	No soportado	8s	10.34 MB
Productos Químicos	No soportado	No soportado	6s	9.56 MB
Combustible	No soportado	No soportado	3s	4.3 MB
Cronograma Real de Perforación	No soportado	No soportado	7s	11.45MB
Diagrama de análisis del Diseño				
Versión 1.0			Versión 2.0	

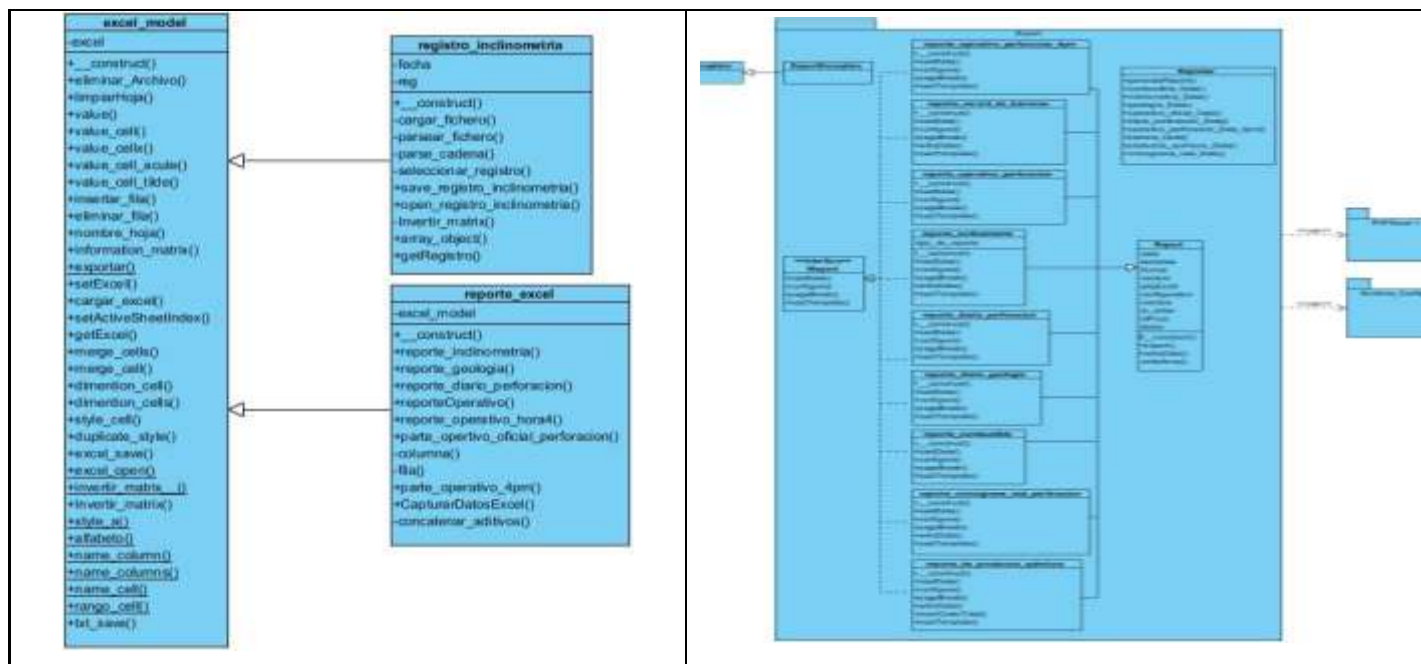
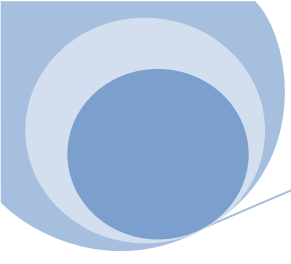


Tabla 4 Pruebas de rendimiento al sistema

Luego de haber realizado las pruebas de funcionalidad y rendimiento al Módulo de Manejo de Archivos en su versión 2.0 se comprueba a través del diseño de casos de prueba la correcta implementación de las interfaces así como las respuestas del mismo referente a cada requisito funcional del sistema. Las clases del diseño del Módulo en la versión 2.0 a diferencia de la versión 1.0, utilizan los patrones de diseño y tiene una arquitectura bien estructurada. Además brinda opciones para la configuración de los datos y el código es legible, fácil de entender y organizado. El tiempo de respuesta y la cantidad de memoria usada de las peticiones al servidor supera en rendimiento al Módulo de Manejo de Archivos versión 1.0. No se produce sobrecarga en la caché al cargar los datos. Por todo lo antes expuesto se valida que el Módulo de Manejo de Archivos versión 2.0 está apto para integrarse al Sistema de Perforación de Pozos de Petróleo.

3.4 Conclusiones parciales

En este capítulo se obtiene el diagrama de componente del sistema que describe los elementos físicos de este y sus relaciones. Se definió la organización del código, se implementaron todas las funcionalidades



Capítulo 3: Implementación y prueba del sistema

definidas para el sistema y se integraron los resultados en un sistema ejecutable. Con el objetivo de comprobar el correcto funcionamiento de los requerimientos del sistema se realizaron las pruebas que determinaron la validez del módulo. El resultado de esto proporciona un módulo completo listo para ser integrado a SIPP.

CONCLUSIONES

El desarrollo del Módulo de Manejo de Archivos versión 2.0 logró cumplir con el objetivo propuesto. A continuación se mencionan los resultados obtenidos a lo largo de la investigación:

- El Módulo de Manejo de Archivos versión 2.0 logra alcanzar la interoperabilidad con otros sistemas que exportan información referente a la perforación de pozos.
- La librería PHPExcel en su versión 1.7.6, permite exportar la información en diferentes formatos como xlsx, pdf, csv y html y mejora el rendimiento de la aplicación.
- El módulo desarrollado facilita el acceso a la información de cada reporte, asegurando la fiabilidad de las decisiones tomadas por los directivos, ya que se abarca de manera íntegra gran parte del área de perforación de pozos.
- El código del módulo fue documentado detalladamente, es fácil de entender, bien estructurado y reutilizable.

RECOMENDACIONES

Teniendo en cuenta el resultado obtenido en la investigación, se proponen las siguientes recomendaciones:

- Se recomienda analizar nuevos registros de pozos para que se valore su implementación en el Módulo de Archivos.
- Se recomienda modificar el diseño a los PDF, para que luzcan mejor a la vista del usuario.

REFERENCIAS BIBLIOGRÁFICAS

1. Olea, I. *Definición de Interoperabilidad*. 2006 [Citado 2011, 18 de Noviembre]; Disponible en: <http://olea.org/diario/archive/2006/dic-21-0.html>.
2. Vaillant, O.R., *El sistema Internacional de Unidades en la Unión CubaPetróleo*. Cuba : Boletín Científico Técnico INIMET, 2008.
3. Victoria. *Definición de fichero*. [Citado 2011, 1 de Diciembre]; Disponible en: <http://www.definicionabc.com/general/fichero.php>.
4. Acosta, R.L., *Subsistema de Manejo de Archivos del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas*. 2011, Universidad de las Ciencias Informáticas: Cuba.
5. *Petróleo*. Ecured [Citado 2011, 12 de Diciembre]; Disponible en: <http://www.ecured.cu/index.php/Petr%C3%B3leo>.
6. Correa, O.M.R., *Solución Informática a pegaduras de tuberías en la perforación de pozos petroleros*. . 2010.
7. *Wellview*. Peloton [Citado 2012, 13 de Enero]; Disponible en: <http://www.peloton.com/es/default.asp?id=13>.
8. *Target*. Geosoft Mayo, 2008 [Citado; Disponible en: www.geosoft.com].
9. *Registros Geológicos*. [Citado 2012, 15 de Enero]; Disponible en: http://petrokemls.com/index.php?option=com_content&view=article&id=87&Itemid=78.
10. *Strater*. AddLink [Citado 2012, 14 de Enero].
11. Pereira, P. *Software, Wellflo*. Modelaje de Pozos [Citado 2012, 12 de Enero]; Disponible en: www.ep-solutions.com/Spanish/Solutions/EPS/WellFlo.htm.
12. *Informática aplicada a la Gestión Pública*. [Citado 2012, 3 de Febrero]; Disponible en: <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>.
13. Pérez, I.C., *Metodología de Desarrollo del Software*. 2008.
14. *Ventajas de las Aplicaciones Web* Solcre Technology Solutions [Citado 2012, 14 de Febrero]; Disponible en: www.solcre.com.
15. *Características de PHP*. Agosto de 2011 [Citado 2012, 13 de Febrero]; Disponible en: <http://www.cav.jovenclub.cu/comunidad/?q=node/>.

16. Saavedra, E., *Frameworks para desarrollo de aplicaciones web*. 2008.
17. *Symfony en pocas palabras*. [Citado 2012, 23 de Febrero]; Disponible en: http://www.librosweb.es/symfony/capitulo1/symfony_en_pocas_palabras.
18. Cerda, F., *NetBeans 6.5*.
19. *Sistema Gestor de Base de Datos*. 9 de Febrero del 2012 [Citado 2012, 27 de Febrero]; Disponible en: http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos
20. *PostgreSQL*.
21. *Características y ventajas del Apache*. Instituto Tecnológico de Mexicali [Citado 2012, 13 de Marzo]; Disponible en: <http://es.scribd.com/doc/52208534/29/CARACTERISTICAS-Y-VENTAJAS-DEL-APACHE>.
22. Balliauw, M., *PHPEXcel Developer Documentation*.
23. Sparks, G., *Introducción al modelado de sistemas de software*.
24. *Flujo de trabajo Análisis y Diseño*. Ecured [Citado 2012, 23 de Mrzo]; Disponible en: http://www.ecured.cu/index.php/Flujo_de_Trabajo_An%C3%A1lisis_y_Dise%C3%B1o
25. *Desarrollo MVC*. [Citado 2012, 25 de Marzo]; Disponible en: <http://iloo.wordpress.com/2010/01/21/desarrollo-mvc-3-capas>.
26. Díaz, O.M., *Ambiente integrado para el modelado y ejecución de procesos de gestión de la información. Herramienta web para la modelación de los procesos*. 2010, Universidad de las Ciencias Informáticas
27. Campo, G.D., *Patrones de diseño, Refactorización y Antipatrones*. 2009.
28. Visconti, M., *Fundamentos de la Ingeniería de Software*.

ANEXOS

¡Error! La autoreferencia al marcador no es válida. **Anexo 1: Comparación de soluciones existentes**

	WellView	Target	MasterLog	Strater	WellFlo
Ficheros que exporta e importa	Excel, Adobe Reader (.pdf), Word (.doc)	ASCII, ODBC, acQuire, CAD, DXF y MapInfo	Word (.doc), Excel (.xls), ASCII, Adobe Reader (.pdf)	ASCII, LAS, Excel (.xls)	Word (.doc), Excel (.xls)
Licencia	Propietario	Propietario	Propietario	Propietario	Propietario
Plataforma	Windows	Windows	Windows	Windows	Windows

Anexo 2: Cuadro de Diálogo Abrir Inclinometría en formato xlsx



Anexo 3: Reporte de Inclinometría en formato xlsx

Reporte de inclinometría [Sólo lectura] - Microsoft Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	Reporte de Inclinometría Real															
	MD	INC	AZI	TVD	VSECT	NS	EW	DLS	EASTING	NORTHING	LATITUDE	LONGITUDE	SUBSEA	FECHA		
4	0	0	325	0	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	-12,47	2012-04-06		
5	30	0	325	30	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	17,53	2012-04-06		
6	60	0	325	60	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	47,53	2012-04-06		
7	90	0	325	90	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	77,53	2012-04-06		
8	120	0	325	120	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	107,53	2012-04-06		
9	150	0	325	150	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	137,53	2012-04-06		
10	180	0	325	180	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	167,53	2012-04-06		
11	210	0	325	210	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	197,53	2012-04-06		
12	240	0	325	240	0	0	0	0	463420	365672	N 23 7 14.260	W 81 21 25.570	227,53	2012-04-06		

Anexo 4: Cuadro de Diálogo Abrir Reporte de Combustible en formato pdf

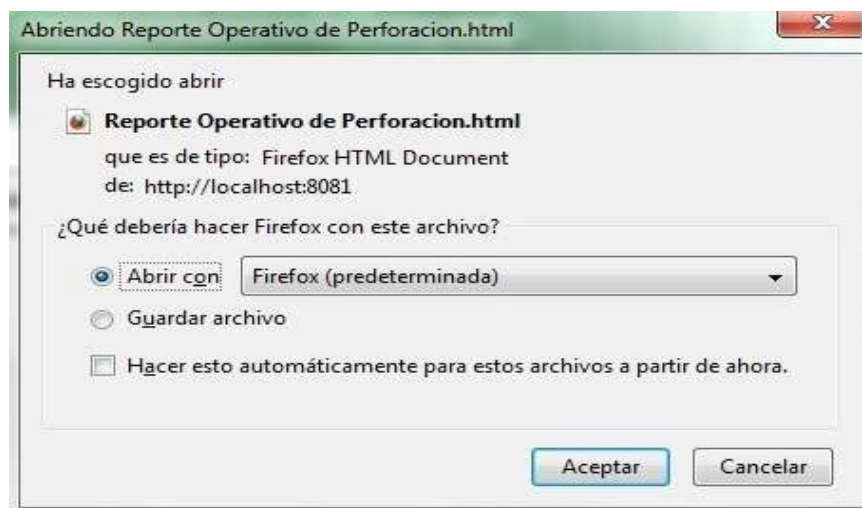


Anexo 5: Reporte de Combustible en formato pdf

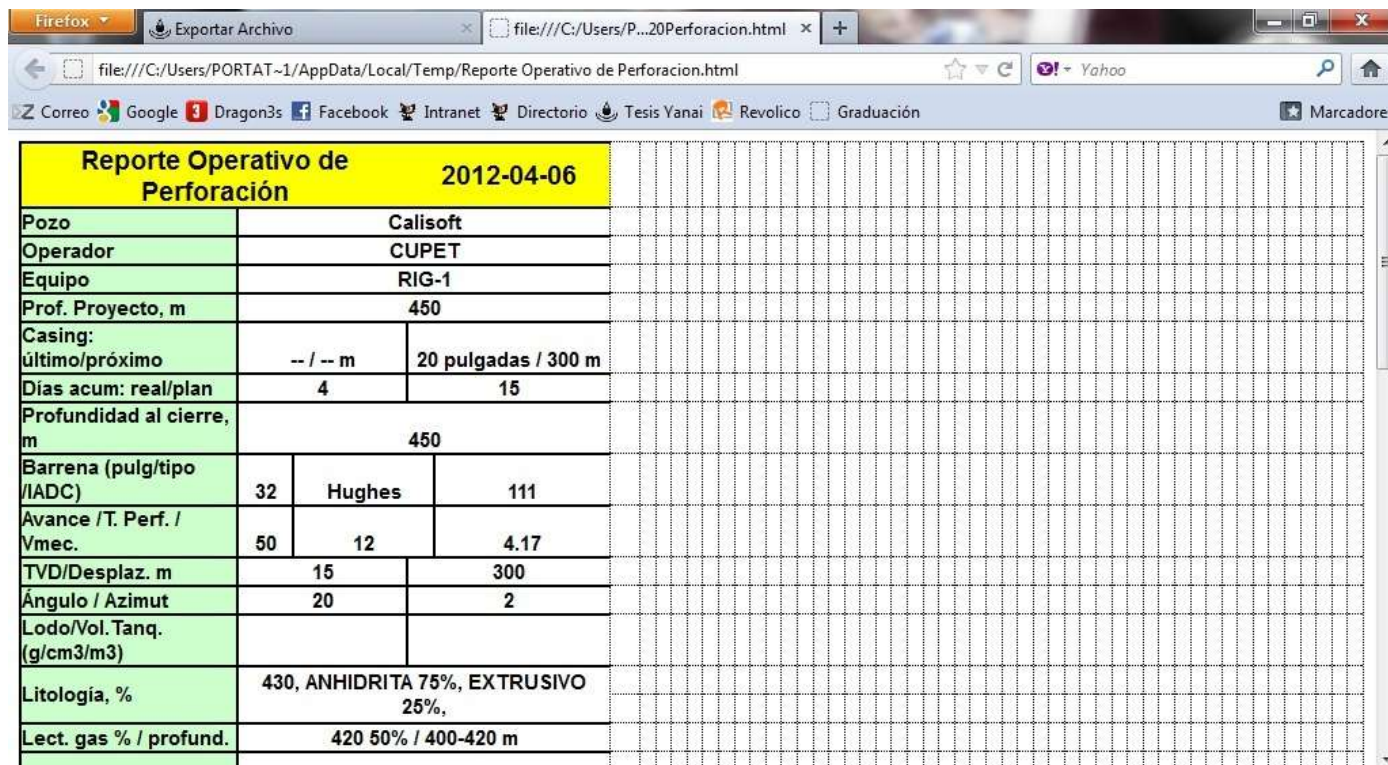
The screenshot shows a PDF document titled 'Reporte de Combustible.pdf' open in Adobe Reader. The document contains a table with the following data:

Reporte de Combustible						
						201
						10
Equip o	Capacidad Total		Consumo Diario		Transferencia	
	Equipo	Campamento	Equipo	Campamento	Equipo	Cam
RIG-1	120000	45000	10	10	678	

Anexo 6: Cuadro de Diálogo Abrir Reporte Operativo de Perforación en formato HTML

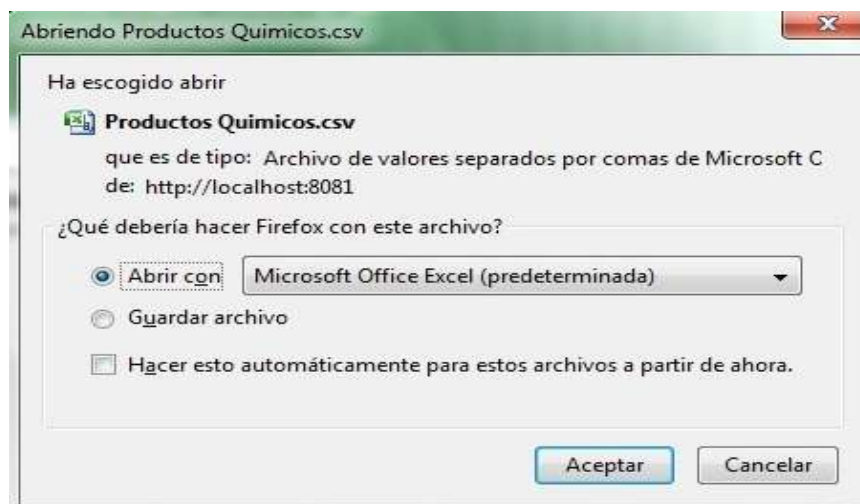


Anexo 7: Reporte Operativo de Perforación en formato HTML



Reporte Operativo de Perforación		2012-04-06	
Pozo	Calisoft		
Operador	CUPET		
Equipo	RIG-1		
Prof. Proyecto, m	450		
Casing: último/próximo	-- / -- m	20 pulgadas / 300 m	
Días acum: real/plan	4	15	
Profundidad al cierre, m	450		
Barrena (pulg/tipo /IADC)	32	Hughes	111
Avance / T. Perf. / Vmec.	50	12	4.17
TVD/Desplaz. m	15	300	
Ángulo / Azimut	20	2	
Lodo/Vol. Tanq. (g/cm3/m3)			
Litología, %	430, ANHIDRITA 75%, EXTRUSIVO 25%,		
Lect. gas % / profund.	420 50% / 400-420 m		

Anexo 8: Cuadro de Diálogo Abrir Reporte de Productos Químicos en CSV



Anexo 9: Reporte de Productos Químicos en CSV

	A	B	C	D	E	F	G	H	I	J	K	L
1	Reporte de Productos Químicos,											
2	,"2012-04-06",											
3	Calisoft,"10:27 am",											
4	Datos Productos,											
5	Producto,"Precio","Cantidad","Costo Diario"											
6	AMICOR,"4","600","2400",											
7	BICARBONATO DE SODIO,"1.3","400","520",											
8	CLORURO DE POTASIO (CUPET),"0","20","0",											
9	CLORURO DE POTASIO (CUPET),"0","80","0",											
10	AMICOR,"4","900","3600",											
11	AMICOR,"4","400","1600",											
12	BICARBONATO DE SODIO,"1.3","600","780",											
13	XGD,"14.8","89","1317.2",											
14	AMICOR,"4","100","400",											
15	Costo Total,"","10617.2",											
16												
17												

GLOSARIO DE TÉRMINOS

TCP/IP: Es un conjunto de protocolos. La sigla TCP/IP significa "**Protocolo de control de transmisión/Protocolo de Internet**" y se pronuncia "T-C-P-I-P". Proviene de los nombres de dos protocolos importantes del conjunto de protocolos, es decir, del protocolo TCP y del protocolo IP.

HTTP: Cada transacción de información realizada en la Web es realizada utilizando el protocolo HTTP, "HyperText Transfer Protocol" por sus siglas en inglés, o **Protocolo de Transferencia de Hipertexto**.

Propel: Es un ORM para PHP que facilita la labor de desarrollo de aplicaciones web, gracias a la capa que transforma el tratamiento de la BD mediante objetos, con la que se puede recuperar, insertar y modificar datos.

HTML: Siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

RUP (*Rational Unified Process*, en español Proceso Unificado de Racional): Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

IDE (*Integrated Development Environment*, en español Entorno de Desarrollo Integrado): Es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (*GUI*). Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Framework: Marco de Trabajo. Es un esquema (un esqueleto, un patrón) para el desarrollo y/o la implementación de una aplicación.

CSV: Archivos separados por coma (Comma Separated Values), es un tipo de archivo en el que se almacena información para exportación/importación desde/hacia una base de datos, donde los valores de cada renglón (registro) de la misma aparecen separados por comas.