

Universidad de las Ciencias Informáticas

Facultad 5



Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informática.



**Subsistema de Manejo de Archivos del Sistema para el Manejo Integral de la Perforación de
Pozos de Petróleo y Gas.**

AUTOR: Roselí Lemes Acosta.

TUTOR: Ing. David Tavares Cuevas.

Co-TUTOR: Ing. Lisset Salazar Gómez.

Ciudad de la Habana, Junio 2011.

“Año 53 de la Revolución”

Declaración de Autoría

Declaro que soy el único autor de este trabajo, autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Roselí Lemes Acosta

David Tavares Cuevas

Lisset Salazar Gómez

Resumen

En Cuba la Dirección de Intervención y Perforación de Pozos (DIPP), es la entidad encargada de dirigir el proceso de perforación de pozos en tierra. Esta unidad empresarial realiza el monitoreo y control del proceso a través de reportes y partes que se elaboran en los pozos de petróleo en perforación. Existen errores y redundancias en los partes y reportes emitidos; además de un bajo control en el almacenamiento de los mismos. 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 Subsistema de Manejo de Archivos del Sistema para el Manejo Integral de Perforación de Pozos de Petróleo y Gas (SIPP), el cual constituye el objetivo de este trabajo.

Como resultado, este subsistema permite importar y exportar, archivos en formatos estándar (.txt y .xls). A través de estos ficheros, los usuarios pueden manipular la información fuera del sistema, así como, utilizar estos datos para otros procesos dentro de la entidad. Utilizando estos archivos como fuente de intercambio de datos, el SIPP debe garantizar la comunicación y coexistencia con otros sistemas en el entorno de trabajo en la DIPP. En este trabajo también se abordan las herramientas y tecnologías utilizadas para la implementación del resultado. También se realiza un análisis profundo de los sistemas que en la actualidad implementan subsistemas similares, con un objetivo común al de este trabajo.

Palabras Clave: control, desarrollo, gestión, industria y petróleo.

Índice de Contenido

Introducción	1
CAPÍTULO 1: Fundamentación teórica	7
1.1 Introducción.....	7
1.2 Conceptos asociados al dominio del problema.....	7
1.3 Proceso de elaboración del Informe final del pozo.	8
1.3.1 Descripción General.	8
1.3.2 Descripción actual del dominio del problema.....	9
1.4 Análisis de soluciones existentes.....	10
1.4.1 Strater.....	10
1.4.2 MasterLog.	12
1.4.3 Wellflo.....	12
1.4.4 Análisis de registros de pozos petroleros.....	13
1.4.5 WellSight.....	14
1.4.6 Target.....	14
1.4.7 WellView.....	15
1.5 Comparación de soluciones existentes.....	16
1.6 Caracterización de los registros de pozos en perforación.....	17
1.7 Comunicación entre sistemas.....	18
1.7.1 Tipos y niveles de Comunicación.....	18
1.7.2 Ficheros estándar de comunicación en la Industria petrolera.	19
1.7.3 Como se manipulan los ficheros para la comunicación entre sistemas.....	20
1.8 Conclusiones Parciales.....	20
CAPÍTULO 2: Herramientas y Tecnologías	21
2.1. Introducción.....	21

2.2.	Sistemas Web.....	21
2.2.1.	Aplicación Web.....	22
2.2.2.	Framework de desarrollo.....	23
2.2.3.	Estilos y estándares de codificación.....	25
2.3.	Paradigma de Programación.....	27
2.3.1.	Programación Orientada a Objeto.....	27
2.4.	Lenguaje de Programación.....	30
2.4.1.	PHP.....	30
2.5.	Entorno Integrado de Desarrollo.....	31
2.5.1.	Zend Studio.....	31
2.6.	Sistema Gestor de Base de Datos.....	32
2.6.1.	Postgres SQL.....	32
2.7.	Servidor Web.....	34
2.7.1.	Apache.....	34
2.8.	Técnicas de Aseguramiento de la Calidad.....	35
2.9.	Conclusiones Parciales.....	36
Capítulo 3:	Enfoque y Validación de la solución propuesta.....	37
3.1	Introducción.....	37
3.2	Requisitos.....	37
3.2.1	Requisitos Funcionales.....	38
3.2.2	Requisitos No Funcionales.....	38
3.3	Diagrama de Caso de Uso del Sistema.....	39
3.3.1	Diagrama de Caso de Uso del Sistema.....	40
3.3.2	Los actores del sistema.....	40
3.3.3	Descripción de los casos de uso.....	42
3.4	Modelo de Implementación.....	43

3.4.1	Vista de implementación.....	44
3.4.2	Diagrama de Componente.....	45
3.5	Manejo de Archivo.	49
3.6	Elementos de la arquitectura.	52
3.6.1	Patrón Arquitectónico.....	52
3.6.2	Patrones de Diseño.	53
3.7	Pruebas del Subsistema Manejo de Archivo.....	54
3.7.1	Prueba de Caja Negra.	55
3.7.2	Prueba de Aceptación.....	55
3.7.3	Descripción de los Casos de Pruebas.	56
3.8	Conclusiones Parciales.....	62
Conclusiones Generales		63
Recomendaciones		64
Referencias Bibliográficas		65
Bibliografía Consultada		¡Error! Marcador no definido.
Anexos		69
Glosario de Términos		¡Error! Marcador no definido.



Índice de Tablas

Tabla 1: Comparación de sistemas que exportan e importan registros de pozos.....	16
Tabla 2: Descripción de los actores del sistema	42
Tabla 3: Caso de Prueba exportar archivo desde el módulo de Pozo.....	59
Tabla 4: Caso de Prueba Exportar Archivo desde el módulo de DIPP con la opción Reporte.....	61
Tabla 5: Caso de Prueba Exportar Archivo desde el módulo de DIPP con la opción Partes.	62

Índice de Figuras

Diagrama 1: Caso Uso del Sistema	40
Diagrama 2: Vista de Implementación.....	44
Diagrama 3: Exportar Archivo	47
Diagrama 4: Importar Archivo	48

Introducción

Hoy día el uso de las Tecnologías de la Información y las Comunicaciones (TIC) es de vital importancia para el desarrollo de nuestra sociedad. Gracias al desarrollo y el buen uso de las mismas se pueden facilitar las comunicaciones, eliminar las barreras de tiempo y espacio, favorecer la cooperación entre distintas entidades y aumentar la producción de bienes y servicios. En Cuba desde el triunfo de la Revolución uno de los principales objetivos de nuestro estado socialista, ha sido desarrollar un camino que pudiera satisfacer las necesidades de nuestro pueblo. El desarrollo de las TIC se ha ido incrementando, tanto así que ha servido de apoyo para el estudio de la ciencia, la tecnología y la innovación. De allí que su uso, se considera como estrategia para el progreso empresarial [1].

Debido al avance de las TIC se ha mejorado la interoperabilidad entre software en el área del petróleo, la cual no es más que la interconexión y funcionamiento conjunto de manera compatible entre los sistemas [2]. Muchos de los 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 [3]. Estos ficheros que almacenan toda la información referente a los parámetros físicos y geológicos del pozo dentro de la industria son denominados registros de pozos [4].

La industria del petróleo no está exenta de este proceso; la Unión Cuba-Petróleo (CUPET) es el conjunto de empresas responsables del desarrollo de la industria del petróleo junto con compañías extranjeras, que operan y prestan servicios a esta importante y vital rama de la economía. CUPET con sus distintas entidades distribuidas en todo el país intervienen en los diferentes procesos por los que debe pasar el crudo. La industria del petróleo se divide en tres grandes procesos:

1. Exploración-Producción.
2. Refinación.
3. Comercialización.

Las entidades especializadas en Exploración-Producción se encargan de la exploración y desarrollo de los campos, la perforación de nuevos pozos, así como la reparación de los que ya se encuentran en producción (Intervención de Pozos) [1].

Compañías transnacionales vinculadas directamente al desarrollo de software para la industria del petróleo, en la rama de la exploración – producción, como Peloton e InfoOil, tienen un amplio desarrollo para el área de la perforación. Peloton desarrolla la plataforma MasterView, la cual contiene la solución Wellview que permite crear un archivo de pozo corporativo completo desde la petición de perforar un pozo hasta la terminación del mismo. Contiene funcionalidades para el pronóstico geológico, de planificación de la perforación, de gestión de las operaciones de perforación, geológicas y de intervención. MasterView agrupa, además, los software RigView y SiteView, enmarcándose en la construcción del cronograma de perforación y la gestión de los proyectos de pozo, y en las operaciones asociadas al control de la locación, montaje y permiso respectivamente [5].

Otras empresas líderes en servicios a la perforación como Schlumberguer, Halliburton, Chevron, entre otras, han incursionado en el desarrollo de sistemas, que soporten los servicios que brindan a la industria. Drilling Office, Petrel, entre otros son soportados por Schlumberguer, compañía que funciona como puente para la comercialización y uso de esos sistemas. Drilling Office es una suite de herramientas las cuales se integran para brindar una solución capaz de soportar el trabajo de la perforación en subprocesos como, proyecciones, cálculos de trayectorias, correlaciones y análisis mecánicos, etc. Este sistema permite importar e importar ficheros de extensión .txt, .xls y ASCII [6].

WELLSIGHT (Baroid Fluid System¹) está diseñado para brindar al ingeniero de campo acceso completo a historiales de pozos cercanos al sitio de perforación, obtener datos y generar informes en formato electrónico. Ofrece además 5 opciones para obtener información de sus base de datos en formatos creados para la industria o para clientes específicos: aplicaciones de manejo de datos de Landmark (DIMS® y OpenWells®), DISWIN de Chevron,

¹ Baroid Fluid System empresa de servicios de lodo de la corporación transnacional Halliburton.

sistema de base de datos de DBR de Statoil y software de modelado hidráulico DGF [7]. LandMark, es una empresa que fueron absorbidas por Halliburton; sus sistemas DIMS y OpenWells han estandarizado sus formatos para que puedan ser leídos por otros sistemas de la misma corporación, como es el caso de WELLSIGTH y garantizando la comunicación y el intercambio de datos de manera más rápida y segura [7].

Estos sistemas tienen las características de ser privativos y desarrollados únicamente para la plataforma Windows. Algunos de ellos pueden ser visualizados en la web, pero principalmente son desarrollados para entornos de escritorio. Todos provén funciones al usuario que le permita manipular los datos fuera de ellos y así poder permitir la interoperabilidad con otros sistemas en el entorno de trabajo. A pesar de su fiabilidad, confiabilidad y experiencia de uso por los usuarios, debido a su alto costo o imposibilidad de adquirirlo; como es el caso del WELLSIGTH, por ser procedente de Estados Unidos de América (EUA) y las restricciones comerciales que impone este país a Cuba por medio del bloqueo económico.

En territorio cubano la entidad responsable de dirigir el proceso de perforación en tierra es la Dirección de Intervención y Perforación de Pozos (DIPP), la cual forma parte de la Empresa de Producción y Extracción de Petróleo del Centro (EPEPC). Actualmente la DIPP no cuenta con un software para la gestión y control de la información del proceso de perforación que permita dar solución de forma automatizada a las actividades que allí se desarrollan, entre las que se encuentran la elaboración de registros con los datos de los pozos durante la perforación. En la Universidad de las Ciencias Informáticas (UCI) se desarrollan sistemas para la automatización de los procesos de la industria del petróleo. En la facultad 5 se encuentra el Centro de Desarrollo Informático Industrial (CEDIN), en el cual se desarrolla el Sistema para el Manejo Integral de la 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.

Durante el levantamiento de negocio, se plantea la necesidad de que se pudieran seguir manipulando la información del proceso, en los formatos preestablecidos por la entidad en

forma de reportes; estos reportes se emiten casi en su totalidad en formato .xls, el otro formato utilizado es .txt. De esta manera los partes y reportes emitidos en .xls ya no son necesarios elaborarlos nuevamente, si no exportarlos del sistema y poder enviar la información diariamente a las instancias superiores. Un elemento identificado fue la posibilidad de que el sistema pudiera importar los ficheros .txt y/o .xls que exportan las compañías direccionales². Este trabajo se hacía manualmente calculando nuevamente los valores que daban las compañías usando Microsoft Excel 2003. Esto trae como consecuencia desfases en los cálculos de dirección del pozo entre las compañías y la empresa, dado que estos se hacen basados en aproximaciones y métodos diferentes.

Además, la entidad utiliza un software llamado Drilling Office el cual se utiliza para hacer los cálculos de trayectoria de los proyectos de pozos; utilizando una funcionalidad análoga se pudiera brindar la posibilidad de tener ambos registros en el sistema con simplemente importarlos. Este sistema exporta ficheros en .xls, aunque con un formato predefinido por la aplicación, por lo cual el costo de la funcionalidad sería crear un algoritmo capaz de leer el fichero y guardarlo en la base de datos de SIPP. Siguiendo esta misma línea el SIPP pudiera interoperar con este u otros sistemas que tienen características similares, a través de ficheros estandarizados en extensiones .txt o .xls.

Partiendo de lo antes planteado se ha definido como **problema a resolver** ¿Cómo obtener información de los registros de pozos a través de ficheros estándar del Sistema para el Manejo integral de la Perforación de Pozos de Petróleo y Gas? Para el desarrollo de este trabajo es necesario analizar el proceso de elaboración del informe final de los pozos de petróleo perforados en tierra por CUPET, lo que constituye el **objeto de estudio**. Para darle solución al problema planteado se hace necesario Implementar un subsistema de Manejo de Archivos del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP) lo cual representa el **objetivo general** de la investigación, el cual está enmarcado en la interoperabilidad del sistema a través de ficheros estándar como las hojas de cálculos (.xls)

² Los pozos en la actualidad se perforan utilizando la técnica de perforación direccional, la cual consiste en perforar haciendo una parábola moviéndose tanto por el eje x, como por el de las y para encontrar el objetivo propuesto. Las compañías que brindan este tipo de servicio se les denomina compañías de servicio direccional.

y los procesadores de texto plano (.txt) lo que representa el **campo de acción**. Por todo lo antes mencionado se **defiende la idea** que implementando el subsistema de Manejo de Archivos, se logrará obtener información de los registros de los pozos perforados en tierra por CUPET.

El subsistema de manejo de archivos, dotará a la solución SIPP de gran versatilidad y capacidad de comunicación utilizando ficheros. Esta funcionalidad, posibilitará que este se pueda equiparar a sistemas similares, incluso superándolos en el futuro, dado que los sistemas existentes no comparten sus formatos, por lo cual la comunicación de estos es limitada. Brindará las funcionalidades con la especificidad suficiente, para el cambio en el proceso y formula de trabajo de la entidad sea mínima y por tanto el proceso de adaptación al sistema sea menor. Todo esto aparejado a la capacidad de coexistir con otros sistemas en el ambiente de despliegue, sin duda es una capacidad que sistemas de distribuidores diferentes no son capaces de ofrecer.

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

- Caracterización de los sistemas que exportan e importan registros de pozos petroleros para conformar el estado del arte.
- Descripción de los registro de pozo para conformar las plantillas de los reportes.
- Descripción de las herramientas y tecnologías a utilizar.
- Selección de las técnicas de aseguramiento de la calidad del código.
- Implementación de la solución propuesta en la investigación.
- Validación del resultado a través de pruebas funcionales para garantizar la calidad del producto.

Para el desarrollo de la investigación se hizo necesario aplicar algunos métodos científicos dentro de los que se incluyen los métodos teóricos y empíricos, facilitando la recopilación de la información necesaria para la realización de un subsistema de importación y 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 a nivel nacional e internacional de las características de los sistemas que exportan e importan registros de pozos petroleros, lo cual permite conocer el estado actual y la evolución del fenómeno.

Analítico-Sintético: Se define con el objetivo de analizar las teorías, documentos, etc., permitiendo obtener, resumir y describir los elementos más importantes relacionados con los procesos de importación y exportación de registros de pozos.

Métodos Empíricos

Observación: Cumple la función de obtener un nivel de realidad, de roce con el objeto de estudio sin el cual se corre el riesgo de no entender a cabalidad los procesos desarrollados en el mismo.

Estructura del contenido

Capítulo 1: En este capítulo se hará una descripción de la fundamentación teórica, además se abordan los conceptos fundamentales asociados al dominio del problema y se hará un estudio de las principales soluciones existentes alrededor del objeto de estudio.

Capítulo 2: Este capítulo describe 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 con mayor eficiencia y calidad al objetivo general de la investigación.

Capítulo 3: En este capítulo se exponen los requisitos funcionales y no funcionales del subsistema, así como, el diagrama de casos de uso y la descripción textual de los mismos. Se analizan las características generales de la implementación y los casos de pruebas diseñados para validar el correcto funcionamiento de la aplicación.

CAPÍTULO 1: Fundamentación teórica.

1.1 Introducción.

Existen diversos conceptos asociados a los procesos de gestión de la información en la industria petrolera, el conocimiento de los mismos es de vital importancia para el entendimiento del problema planteado. Se abarca específicamente el proceso de obtención de información de los registros de pozos y el análisis de soluciones existentes. Así como la caracterización de los archivos relacionados con los pozos de petróleo en perforación.

1.2 Conceptos asociados al dominio del problema.

Al problema que se le da solución en este trabajo, se requiere de mucha experiencia para poder conceptualizar y entender los términos que se manejan, a continuación se muestran una serie de conceptos identificados a lo largo de la investigación, para un mejor entendimiento de los acápite que se abordan en el presente y posteriores capítulos.

Industria Petrolera: La industria petrolera es la que se encarga de los procesos de exploración, extracción, refino, transportación y comercialización del petróleo y sus derivados [8].

Petróleo: Es una mezcla de hidrocarburos que se extrae de lechos geológicos continentales o marítimos, es un compuesto químico en el que se encuentran juntas las partes sólidas, líquidas y gaseosas. También es conocido como crudo y su color puede variar desde incoloro a negro [9].

Información: Se define como un conjunto de datos organizados, significativos y pertinentes que describen sucesos o entidades. En cuanto al universo de la computadora, la información es un factor fundamental que se representa a través de símbolos, específicamente en forma de datos binarios [10].

Unión CUPET: Es la entidades que se encarga de todas las actividades desde la exploración hasta el refinado del crudo, así como de satisfacer las necesidades en el mercado nacional [11].

Registros de Pozos: Los registros de pozos de petróleo son técnicas geofísicas, que se utilizan en las operaciones petroleras para obtener una mayor información de los parámetros físicos y geológicos del pozo [12].

Subsistema: Un subsistema es un paquete de clases, asociaciones, operaciones, sucesos y restricciones interrelacionados, y que tienen una interfaz razonablemente bien definida y pequeña con los demás subsistemas, un subsistemas es un componentes principales de un sistema [13].

1.3 Proceso de elaboración del Informe final del pozo.

1.3.1 Descripción General.

Una vez finalizadas las operaciones en el pozo, en un plazo de 15 días posteriores, se entrega el Informe Final del Pozo. Este informe incluye los siguientes documentos:

- Resumen del pozo.
- Información geológica (Columna Geólogo-Técnica Real).
- Distribución del tiempo.
- Resumen de la Construcción del Pozo.
- Registros de Inclínometría.
- Record de barrenas.
- Composiciones de herramientas.
- Consumo de Productos Químicos.
- Parte diario de operaciones durante la perforación.
- Parte diario del Lodo.
- Parte diario de ensayo y puesta en producción.
- Reporte de gastos.

- Reporte de Encamisado. Camisa Conductora. Implantado en el Procedimiento Encamisado de Pozos.
- Reporte de Encamisado. Camisa Técnica. Implantado en el Procedimiento Encamisado de Pozos.
- Reporte de Encamisado de Liner. Implantado en el Procedimiento Encamisado de Pozos.
- Curva del cronograma de perforación del pozo.

La información que contiene cada uno de los documentos mencionados, son emitidas por el direccional, geólogo, químico y logístico del pozo. El direccional es el encargado de entregar los datos referentes a las barrenas utilizadas en la perforación del pozo. El geólogo propicia la información referente al estudio del fluido obtenido durante la perforación. El químico es el que confecciona el lodo y brinda una lista con todos los productos químicos utilizados para el mismo y un inventario de los productos utilizados y el logístico ofrece la información del combustible utilizado por las plantas que dan electricidad al equipo de perforación.

Toda esta información es entregada al supervisor del pozo, ya que es el encargado de elaborar el informe final del pozo. Una vez confeccionado dicho informe, se le entrega una copia impresa al Archivo Técnico de la EPEPC para su conservación y uso por tiempo indeterminado y otra copia se guarda en el servidor de la EPEPC. El Jefe de Grupo de Perforación es el responsable de dar cumplimiento a lo establecido en el presente procedimiento. Este proceso es de suma importancia, ya que guarda todas las incidencias ocurridas en la perforación y sirve como información base para confeccionar nuevos proyectos de pozo basándose en resultados precedentes.

1.3.2 Descripción actual del dominio del problema.

La Dirección de Intervención y Perforación de Pozos (DIPP), es la encargada de dirigir y monitorear todos los procesos que se realizan en los pozos en perforación. Esta entidad radica en la Empresa de Producción y Extracción de Petróleo del Centro (EPEPC), perteneciendo a la Dirección de Exploración - Producción de la Unión de CUPET. Aquí se realizan y se aprueban los proyectos de los pozos a perforar, los cuales se elaboran por un

equipo multidisciplinario de diferentes ramas, siendo así los responsables de los pozos desde su nacimiento hasta su terminación.

Las actividades de gestión y control de información en los pozos en perforación, la ejecutan los supervisores. El proceso de perforación se efectúa contratando servicios a empresas, las cuales pueden ser cubanas o extranjeras. Estas compañías emiten diariamente un reporte de sus servicios. Los supervisores (jefe del pozo) son los encargados de dirigir el proceso de perforación, además de construir los reportes de los procesos realizados integrados con los que emiten las compañías de servicios.

Actualmente todo el proceso de gestión, control y flujo de información se lleva a cabo manualmente, ya que la DIPP no cuenta con un software que le permita la elaboración de los reportes de manera automatizada. Buscan alternativas viables que propicien un trabajo más factible para satisfacer las necesidades de la empresa.

1.4 Análisis de soluciones existentes.

1.4.1 Strater.

STRATER es una potente e innovadora herramienta para el registro de pozos y el trazado de perforaciones. Con esta herramienta los geólogos ya no tendrán que invertir más tiempo ni dinero para crear registros profesionales de pozos. Strater es una herramienta potente pero sencilla de usar.

Características Principales

Strater ofrece una flexibilidad insuperable para el diseño y formateo de registros. Su avanzada interfaz de usuario permite que el diseño y la visualización de sus datos sean más fáciles que nunca. Strater incluye 13 tipos de registro muy usuales para visualizar sus datos gráficamente: profundidad, línea/símbolo, gráfico cruzado, petrología, barras de zona, barras, porcentajes, postes, postes por clase, gráficos, textos complejos, y registros de construcción

de pozos. Cada uno de los registros puede ser modificado para ajustarse a las necesidades del usuario. Entre algunas de las personalizaciones posibles podemos citar:

- Visualizar registros basados en profundidad o elevación.
- Visualizar profundidad y/o líneas de rejilla variable.
- Añadir barras de escala y títulos.
- Configurar diferentes estilos de líneas de contacto entre unidades litológicas.
- Utilizar curvas para crear perfiles litológicos como perfiles de desgaste.
- Crear registros continuos sencillos o de múltiples páginas.
- Formatear diferentes tipos de registro para obtener presentaciones lo más informativas posible.

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 robusta base de datos que soporta importación de datos de muchas fuentes diferentes, incluyendo ficheros de texto ASCII, ficheros LAS, 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. También puede crear múltiples vistas de perforaciones en un proyecto.

Cada vista de perforación puede contener un diseño de perforación diferente, permitiendo así una flexibilidad inigualable en la presentación de sus datos. El almacenamiento de todas las vistas y tablas de datos de las perforaciones en un único proyecto mantiene toda la información fácilmente accesible. La edición y manipulación de objetos con Strater es muy fácil. Sólo hay que seleccionar algún objeto de la vista de perforación mediante el Gestor de Objetos (Object Manager). El Gestor de Objetos organiza de forma intuitiva y conveniente todos los objetos en una vista en forma de árbol jerárquico. Una vez se ha seleccionado el objeto, editar todas sus propiedades es sencillo gracias al nuevo Inspector de Propiedades

(Property Inspector). Además cualquier cambio se puede ver de forma inmediata [14].**(Anexo 1)**

1.4.2 MasterLog.

MASTER LOG es un software para generar el Registro de Formación, permite crear registros de alta calidad en blanco y negro o a todo color con un mínimo de instrucciones. Una gran cantidad de formatos está Citado y de ser necesario PLS puede sugerir el empleo de uno o más formatos que cumplan con los requerimientos de la compañía operadora. Por otra parte cualquier formato de registro pedido por la operadora puede ser reproducido. Los datos entran al programa por dos vías:

- Datos capturados automáticamente, como datos de perforación, gas total y cromatografía.
- Datos ingresados manualmente tales como litología de ripios, valores de calcimetría, datos de presencia de hidrocarburos, etc.

El software consta de pantallas y menús fáciles de seguir para hacer la entrada de los datos: Litología, Datos de núcleos, etc., y puede generar todo tipo de Registros, incluidos registros de geología, registros de presiones, registros con relaciones de gases, etc. 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 [15].**(Anexo 2)**

1.4.3 Wellflo.

WellFlo™ es una aplicación autónoma, poderosa y simple de usar para diseñar, modelar, optimizar e identificar problemas de pozos individuales de crudo y gas, ya sean naturalmente fluentes o levantados artificialmente. El software tiene también dos sub-aplicaciones internas claves que pueden ser usadas de manera autónoma del resto del programa y ofrecer así al usuario un excelente kit de herramientas de ingeniería. Los archivos de reporte pueden ser generados en formatos separados por tabulaciones para permitir una fácil exportación hacia paquetes de procesamiento de palabra y hoja de cálculo. La capacidad de 'cortar y pegar' del

ambiente Windows™ permite una generación muy eficiente de reportes, incluyendo gráficas y otras capturas de pantallas. Una opción de exportación de gráficos también permite que los gráficos sean guardados directamente a archivo en un número de formatos.

- Hay disponibilidad de generación de archivos de presión fluyente UNIX y DOS para simuladores de reservorio Eclipse, VIP, CHEARS, SimBest I y II, IMEX, MoRes, GCOMP, COM4 y un modo por lotes multi-pozo para algunas salidas.
- Salida de archivos por palabra clave Citado para transferencia de datos a otras aplicaciones.
- La facilidad de exportación de gráficos guarda estos directamente a archivo (formatos BMP, GIF, JPG y TIF).
- Los reportes pueden ser abiertos directamente en Word, Excel, etc.

La profundidad de presión medida o data de presión y tasa pueden ser importadas hacia el repositorio de datos WellFlo™ y graficado en los mismos gráficos que la predicción del modelo. Esto reduce significativamente el tiempo que se toma producir un modelo de data igualada. Una opción de auto-regresión estima el factor de ajuste de correlación de flujo óptimo de acuerdo a un ajuste de mínimos cuadrados con respecto a la data medida [16].
.(Anexo 3)

1.4.4 Análisis de registros de pozos petroleros.

En la UCI se han venido trazando estrategias para establecer convenios con empresas que se dedican a la Industria Petrolera. Dentro de la antigua facultad 9, existía el polo productivo PetroSoft en el cual se encontraba el proyecto de Sistema de Interpretación y Análisis Petrofísicos de los Registros de Pozos (ANPER) dentro del cual se hicieron algunas investigaciones que dieron pasos a la creación de subsistemas de Importación y exportación de registros de pozos.

Características

- Aplicación de desktop.
- Desarrollado en el lenguaje de programación C Sharp (C#).
- Desarrollado bajo la metodología Proceso Unificado de Desarrollo de Software (RUP).
- Permite la carga de ficheros en TXT y las versiones 1.2, 2.0 y la 3.0 de los archivos .LAS
- Permite la conversión de cualquiera de las versiones 1.2 y 2.0 de los ficheros .LAS a la 3.0.
- Permite la exportación del archivo convertido [17]. **(Anexo 4)**

1.4.5 WellSight

Software para administrar la información generada por los ingenieros de fluido de perforación. Esta aplicación brinda acceso completo a historiales de pozos cercanos en el sitio del equipo de perforación y permite generar informes en formato electrónico. Entre las características que presenta este software es que puede generar reportes en inglés, portugués, alemán y ruso y en formato PDF y en los formatos de Microsoft Office. Permite la planificación de pozos futuros con datos de pozos actuales y de historiales y ofrece cinco opciones para exportar datos de las bases de datos locales, en formatos estándar de la industria o formatos específicos para clientes entre ellos se encuentran: aplicaciones de manejo de datos de Landmark (sistemas DIMS® y OpenWells®), DISWIN de Chevron, sistema de reportes DBR de Statoil y software de modelado hidráulico DFGMR de Baroid [7].

1.4.6 Target

La herramienta Target permite la manipulación de una enorme cantidad de datos de la geoquímica, geofísica y geología de la superficie y del subsuelo, sea simple, dentro de un único entorno 3D interactivo, transparentemente vinculado. Integra todos los datos Citados en cada etapa del programa de exploración para mejorar la comprensión de la geología del subsuelo subyacente, para verificar las suposiciones y compartir las ideas con equipos internos y externos de trabajo y brinda la posibilidad de procesar información de varios pozos en perforación.

Características

- **Planos y Secciones de Pozos en Perforación:** Visualiza de forma rutinaria varios pozos en perforación a la vez. Crea fácilmente secciones y planos de pozos de perforación, utilizando la Interfaz Gráfica de Usuario (GUI por sus siglas en inglés) con sus sensibles ajustes por defecto.
- **Importación de Datos:** Los asistentes o “wizards” de importación, de fácil uso, permiten importar los datos de sus pozos de perforación desde un amplio rango de fuentes de datos, incluyendo las plataformas ASCII, ODBC y acQuire. También puede importar datos en formatos CAD y DXF. Cuando usted realiza una importación de datos, la función de validación de los mismos asegura la integridad de cualquier información del subsuelo que usted genere.
- **Integración de Datos de Pozos de Perforación y de la Superficie:** Combina los datos de los pozos de perforación con todos los otros datos e información de la superficie, para crear mapas detallados e integrados de la superficie y del subsuelo [18].

1.4.7 WellView

WellView® es un archivo de pozo que registra todos los cambios y operaciones a través del ciclo de vida del pozo. Con su poderoso estado mecánico y reportes, coloca la información en las manos de las personas que más la necesita sin las limitantes de archivos de papel tradicionales o reportes diarios. Peloton comenzó con el desarrollo de WellView en 1991 para operaciones de pozo y visualización del estado mecánico del pozo. El estado mecánico de WellView le permite visualizar los equipos en el pozo y eventos (lecciones aprendidas, fallas, etc.). Esto le provee una historia visual del pozo. La base de datos de WellView captura todos los datos relacionados con el pozo y sus operaciones así como del agujero y equipo de superficie.

Trabajando con MasterView® de Peloton, WellView es parte de una solución con análisis de ciclo de vida de pozo integrado y capacidad de visualización. Es el sistema con mayor escalabilidad del mercado, tiene el más completo inventario de aplicaciones para importación

de datos Citado, incluyendo la importación completa de DIMS, PERC, WellCore y otros. Es un verdadero sistema para manejar archivos electrónicos de pozos con un completo modelo de datos y anexos. Archivos tipo Excel, PDF, Word, imágenes, etc. pueden ser anexados al archivo del pozo, quedando almacenados dentro de la base de datos. 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 [5].

1.5 Comparación de soluciones existentes.

	STRATER	MASTER LOG	Wellflo	Análisis de registros de pozos petroleros	WellSight	Target	WellView
Tipo de Aplicación	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop
Ficheros que exporta e importa	ASCII, LAS, Excel (.xls)	Adobe Reader (.pdf)	Word (.doc), Excel (.xls)	ASCII, LAS	Adobe Reader (.pdf)	CAD, DXF	Excel, Adobe Reader (.pdf), Word (.doc)
Tipo de Licencia	Propietario	Propietario	Propietario	Libre	Propietario	Propietario	Propietario
Plataforma	Windows	Windows	Windows	Windows/Linux	Windows	Windows	Windows

Tabla 1: Comparación de sistemas que exportan e importan registros de pozos.

Luego de este estudio se puede apreciar que las aplicaciones STRATER, MASTER LOG, Wellflo, WellSight, Target y WellView han sido probadas a nivel mundial teniendo gran competitividad y las empresas que los desarrollan tienen gran experiencia y conocimientos en los procesos referentes a la industria petrolera; mientras que el sistema desarrollado en la UCI no se llegó a implantar por la tecnología utilizada para su desarrollo. Las seis primeras están liberadas bajo licencias propietarias, eso dificulta el acceso al funcionamiento interno de estos

software, para poder analizar y reutilizar las funcionalidades que brindan y adaptarlas a las necesidades del subsistema que se va a desarrollar, por lo que implantar una de estas soluciones implicaría que el país tenga que invertir grandes sumas de dinero en la adquisición de sus licencias para mantenerlos actualizados. Además de ello habría que pagar las modificaciones que se le deberían hacer a parte de estos reportes para que se ajusten a las características del proceso y los informes en Cuba. En el caso del sistema desarrollado en la UCI, no cumple con los requisitos expuestos por el cliente de SIPP. Al código fuente de esta aplicación se tuvo acceso, pero el mismo al no estar regido por un estándar de codificación, se dificulta entender el funcionamiento interno del mismo.

Por tales razones se valida la necesidad de implementar un subsistema para exportar archivos en .xls y.txt que se ajusten correctamente a los archivos que se manejan actualmente en el informe final de los pozos perforados en tierra. Además de exportar los reportes y partes diarios que se realizan en la DIPP y los pozos durante la perforación.

1.6 Caracterización de los registros de pozos en perforación.

Existen diferentes tipos de reportes de pozos utilizados en el proceso de perforación; Reporte diario de perforación, geología, operativo oficial e Inclínometría entre otros y cada uno brinda una información específica. A continuación se caracterizan los distintos reportes mencionados:

Reporte Diario de Perforación: Informe en formato Excel que se emite con toda la información referida a la actividad de perforación. En él se incluyen los tiempos en cada operación, las operaciones, los parámetros mecánicos, las herramientas de perforación, la construcción del pozo y el control de gastos [19].**(Anexo 5)**

Reporte de Geología: Informe en formato Excel compuesto por dos secciones fundamentales. Una de las secciones es Datos Generales, en la cual se recoge la información referente a la persona que recibe el reporte, la altura sobre el nivel del mar, el costo y el presupuesto asignado. La otra sección es información geológica en la que se incluye la última

muestra descrita a la profundidad, la información referente a las rocas de las formaciones geológicas (litotipos), los intervalos de perforación, diámetro y zapato.(Anexo 6)

Parte Operativo de Perforación Oficial: Permite tener una noción del estado del pozo que está operando, en este se puede encontrar información referente al avance de la perforación, tiempo que se lleva perforando y profundidad al cierre.(Anexo 7)

Reporte de Inclínometría: Este informe recoge la información referente a la profundidad de medición y vertical, la orientación norte-sur y este-oeste, latitud y longitud del pozo así como la profundidad bajo el nivel del mar. Dicho reporte se puede encontrar en formato Excel (.xls) y en texto plano (.txt).(Anexo 8)

1.7 Comunicación entre sistemas.

1.7.1 Tipos y niveles de Comunicación.

La comunicación fue definida como la habilidad de un sistema o producto de trabajar con otros sistemas o productos sin esfuerzo especial del usuario [20]. Por otro lado, la norma ISO 16100 define la interoperabilidad del software industrial como la habilidad de compartir e intercambiar información para cumplir una relación funcional de una aplicación específica a través del uso de una interfaz común [20].

El intercambio entre aplicaciones es un factor importante para el envío de información, esta se puede efectuar mediante formatos estándar, protocolos e interfaces [21]. La comunicación puede ser a nivel de hardware o a nivel de software [22] . Para la interoperabilidad a nivel de software se definen los siguientes estándares de comunicación:

- **HTML/XHTML:** Para formatos de documentos estructurados.
- **SQL:** Especificación con múltiples generaciones de diseño y además menos variantes oficiales.
- **IP:** Para transmitir paquetes de datos en una red.
- **TCP:** Para implementar flujo de datos en IP.

- **PDF/X y PDF/A:** Especificaciones originarias de Adobe Systems Incorporated para algunos formatos de documentos (PDF).
- **OpenDocument:** Para documentos ofimáticos incluyendo documentos de texto, hojas de cálculo, diseños vectoriales y presentaciones [22].

1.7.2 Ficheros estándar de comunicación en la Industria petrolera.

Los archivos también denominados ficheros; es una colección de información (datos relacionados entre sí), localizada o almacenada como una unidad en alguna parte de la computadora. Los archivos son el conjunto organizado de informaciones del mismo tipo, que pueden utilizarse en un mismo tratamiento; como soporte material de estas informaciones [23].

La implantación de sistemas informáticos en la industria petrolera, ha pasado hacer de gran uso e importancia debido al gran volumen de información que manejan. Esta información refleja los resultados de los diferentes procesos que se llevan a cabo dentro de la industria. Debido al almacenamiento de la misma en sistemas gestores de bases de datos, se hace necesaria la extracción de la información para posterior análisis por parte de los especialistas. Para ello, se determinaron un grupo de formatos de ficheros estándar, correspondientes a cada uno de los diferentes procesos por los que debe pasar el crudo, entre los que se encuentran: los ASCII, .LAS, .XLS, .PDF, .TXT, .DOC, CAD y DXF.

En los procesos descritos en la primera iteración del desarrollo de SIPP, se plantea la necesidad de importar y exportar datos. En estos mismo se describen los formatos estándar que utilizan para emitir los partes y los reportes una vez extraída la información de la base de datos (formatos xls y txt), así como las plantillas que se utilizan para cada uno de ellos. Al permitir la carga y extracción de los datos en estos tipos de ficheros, brinda la posibilidad a los usuarios de manipular información en un formato más conocido por ello, facilitándole el análisis para la toma de decisiones.

1.7.3 Como se manipulan los ficheros para la comunicación entre sistemas.

Los archivos como colección de datos sirven para la entrada y salida a la computadora y son manejados con programas.

Estos no requieren de un tamaño predeterminado; lo que significa que se pueden hacer archivos de datos más grandes o pequeños, según se necesiten. Los archivos pueden ser registros y por esto en un registro se deben especificar los campos, el número de elementos y el número de caracteres en una cadena; por esto se denotan como "Estructuras Estáticas".

Existen distintos métodos para acceder a los datos de los registros, entre ellos se encuentra el secuencial, el directo, por índice y de forma dinámica. El método que se utiliza para tener acceso a la información de los partes y reportes en el subsistema a desarrollar es el directo, ya que cada archivo puede leerse y escribirse de forma directa solo con saber su dirección en el fichero o por transformaciones de la clave de registro en el número relativo del registro a acceder [23].

1.8 Conclusiones Parciales.

Luego de abordar todos los elementos plasmados en el capítulo se puede concluir que los sistemas existentes a nivel nacional e internacional, permitió conocer las extensiones en las que se pueden exportar e importar la información que se generan en los mismos, bajo que tipo de licencia fueron desarrollados así como la plataforma en la que se pueden ejecutar; esto representó la base teórica general para poder elaborar una propuesta de solución, capaz de responder a las necesidades del cliente y de la industria. Se identifican un amplio número de conceptos los cuales permiten profundizar en elementos técnicos, necesarios para que la solución maneje correctamente los términos del negocio. La caracterización de los diferentes tipos de archivos de registros de pozos en perforación, permite conocer los tipos de reportes con los que se va a trabajar durante la investigación y de esta manera ajustarse a las plantillas utilizadas por la entidad cliente.

CAPÍTULO 2: Herramientas y Tecnologías.

2.1. Introducción.

Las TIC son herramientas computacionales e informáticas que procesan, almacenan, sintetizan, recuperan y presentan información de forma variada. Su principal propósito es transformar el entorno, para adaptarlo mejor a las necesidades y deseos del hombre. En este capítulo se describen y caracterizan las tecnologías y tendencias actuales a utilizar para implementar un subsistema de manejo de archivos para la exportación e importación de registros de pozos, así como los estándares y estilos de codificación aplicados durante la creación del mismo.

2.2. Sistemas Web.

La programación Web, parte de las siglas WWW, que significan World Wide Web o telaraña mundial. Para realizar una página con la programación Web, se deben tener claros, tres conceptos fundamentales los cuales son, el URL (Uniform Resource Locators), que es un sistema con el cual se localiza un recurso dentro de la red, este recurso puede ser una página web, un servicio o cualquier otra cosa. En resumen el URL no es más que un nombre, que identifica una computadora, dentro de esa computadora un archivo que indica el camino al recurso que se solicita.

El HTTP (Hypertext Transfer Protocol) es el protocolo encargado de llevar la información que contiene una página Web por toda la red de internet y por último el lenguaje necesario cuya funcionalidad es la de representar cualquier clase de información que se encuentre almacenada en una página Web, este lenguaje es el HTML (Hypertext Markup Language por sus siglas en inglés) [24].

2.2.1. Aplicación Web.

Una aplicación web es un software, en el cual un usuario por medio de un navegador, realiza peticiones a una aplicación remota accesible a través de Internet o a través de una intranet y que recibe una respuesta que se muestra en el propio navegador [25].

Características de las Aplicaciones Web

Las aplicaciones web tienen diferentes características, entre ellas que son aplicaciones basadas en el modelo Cliente/Servidor, están formadas por diferentes módulos, independientes entre sí. Se ejecutan en el servidor o en el cliente y pueden estar implementados utilizando lenguajes o tecnologías diferentes y para tener la información organizada suelen utilizar bases de datos [26]. Estas aplicaciones se encuentran alojadas en un servidor web y son accesibles mediante el internet, usando un navegador web. Otras de las características de estos sistemas es que la lógica de estas aplicaciones se ejecuta en el servidor, mientras que el cliente solamente representa los datos y el acceso al sistema puede ser público o restringido. La actualización del sistema no afecta ni depende del cliente y son aplicaciones multiplataforma, ya que pueden ejecutarse en cualquier Sistema Operativo que tenga un servidor web o un navegador Web [11].

Ventajas

Las aplicaciones web ofrecen grandes ventajas las cuales pueden ser utilizadas al máximo para darle solución a los problemas existentes en las Empresas de Petróleo, estas son:

- **Facilidad de manejo:** Ya que la interacción con el usuario se establece en base a elementos a los que está cada vez más acostumbrado, páginas web, que le permiten conocer la funcionalidad del sistema con poco esfuerzo.
- **Accesibilidad:** Las aplicaciones web son accesibles desde cualquier punto de la red, lo cual significa que un usuario autorizado (se pueden establecer controles de acceso) puede utilizarla si dispone de cualquier conexión a Internet (salvo el caso de que se trate

de una aplicación que funcione en una red TCP/IP propia, en cuyo caso será necesario disponer de acceso a la misma).

- **Portabilidad:** Los navegadores web se han desarrollado para todo tipo de máquinas, por lo que cualquier usuario de internet, dispone de la herramienta básica para lanzar la aplicación.
- **Facilidad de desarrollo:** En este sentido, hay dos aspectos a destacar que determinan el desarrollo de este tipo de sistemas. En primer lugar, la descomposición intrínseca en componentes, así como al hecho de que en cierta medida, algunos aspectos que tienen que ver con el carácter distribuido de la aplicación están resueltos de antemano (por ejemplo el protocolo HTTP controla el acceso a datos en el servidor) [26].

2.2.2. Framework de desarrollo.

En el desarrollo de software, un framework es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y a unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio [27].

Se puede decir, que el uso de un framework permite que el proceso de desarrollo de una aplicación sea más acelerado, la reutilización de códigos ya existentes y promover buenas prácticas de desarrollo como el uso de patrones. También facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

Características

Las características fundamentales de los frameworks de desarrollo son:

- **Abstracción de URL y Sesiones:** No es necesario manipular directamente las URL ni las sesiones, el framework ya se encarga de hacerlo.
- **Acceso a datos:** Incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en BBDD, XML, etc.
- **Controladores:** La mayoría de los frameworks implementan una serie de controladores para gestionar eventos, como una introducción de datos mediante un formulario o el acceso a una página. Estos controladores suelen ser fácilmente adaptables a las necesidades de un proyecto concreto.
- **Autenticación y control de acceso:** Incluyen mecanismos para la identificación de usuarios mediante nombre de usuario y contraseña y permiten restringir el acceso a determinadas páginas, teniendo en cuenta los privilegios de acceso de los usuarios [28].

2.2.2.1. Symfony.

Symfony es un framework completo diseñado para optimizar gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, 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.

Características

Las características fundamentales de Symfony es que es fácil de instalar y configurar en la mayoría de las plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y *nix estándares), es independiente del sistema gestor de base de datos y sencillo de usar en la mayoría de los casos, y es lo suficientemente flexible como para adaptarse a los casos más complejos.

Es basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional, sigue la mayoría de mejores prácticas y patrones de diseño para la web, preparado para aplicaciones empresariales y adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.

El código es fácil de leer, incluye comentarios de phpDocumentor y permite un mantenimiento muy sencillo, además es fácil de extender, lo que favorece su integración con librerías desarrolladas por terceros [29].

Con la utilización de Symfony se acelera el proceso de desarrollo. Este framework incluye helpers (contenedores de códigos) que brindan la posibilidad de minimizar el código utilizado ya que encapsulan grandes bloques de códigos en llamadas simples y permite la validación y relleno automático de datos.

2.2.3. Estilos y estándares de codificación.

Para un mejor entendimiento del código, se hace uso de estilos y estándares de codificación. Un estándar de codificación es el establecimiento de un único estilo del código para todo el proyecto, es establecer una normalización de la programación de forma tal, que al trabajar en un proyecto cualquiera de las personas involucradas en el mismo tenga acceso y comprenda el código. En otras palabras define la escritura y organización del código fuente de un programa.

Los estándar de codificación definen la forma en que se organizan los archivos, la sangría, comentarios, declaraciones, afirmaciones, el espacio en blanco, las convenciones de nomenclatura, las prácticas de programación, los principios de programación, las normas de programación general, etc.

Muchas personas leen el código de un programa para ayudar a mejorar la legibilidad de su código fuente y hacer más fácil el mantenimiento del software. Para estos lectores es

fundamental que el programa esté bien redactado, con estilo, para que su significado sea claro. Existe una serie de reglas básicas que ayudan a conseguir un texto satisfactorio [30].

2.2.3.1. Descripción de los estilos y estándar de codificación.

Symfony trae definido sus propios estilos y estándar de codificación:

- Nunca use las tabulaciones en el código, la sangría se hace por pasos de dos espacios.
- No ponga espacio después de un paréntesis de apertura y antes de un cierre.
- Las llaves van siempre en su propia línea.
- Utiliza las llaves para indicar la estructura de control del cuerpo, independientemente del número de declaraciones que contenga.
- Symfony está escrito con PHP5, por lo que cada método de definición de clase o miembro debe declarar explícitamente su visibilidad utilizando las palabras clave privada, protegida o pública.
- En un cuerpo de la función, las declaraciones de retorno debe tener una línea en blanco antes de él para aumentar la legibilidad.
- Todo lo que se comente en una línea debe estar en este formato.
- Evite la evaluación de las variables dentro de cadenas, en lugar de optar por la concatenación.
- Utilice minúsculas constantes de PHP: falso, verdadero y nulo. Lo mismo ocurre con array(). Al contrario, siempre use cadenas en mayúsculas con las constantes definidas por el usuario, como la define („MY_CONSTANT", 'foo / bar '). Mejor, trate de utilizar siempre las constantes de clase.
- Para comprobar si una variable es nula o no, no utilice la función de PHP `is_null ()`:
- Formato de archivo:

Para aquellos archivos que contienen sólo código PHP los tags de demarcación (“<? “) no estarán permitidos, además no es requerido por PHP y omitirlos nos previene de algún accidente ocasionado por un espacio en blanco.

- **Parámetros:**

Los parámetros van siempre en minúsculas.

- **Variables:**

El nombre de las variables debe estar compuesto de caracteres alfanuméricos, el carácter underscore (`_`) está permitido. Siempre tiene que comenzar con letra minúscula. Además siempre debe inicializarse y sobre todo deben tener nombres significativos.

- **String literales:**

Cuando se le asigna un texto literal (sin contenido de variables) se utilizarán comillas dobles.

- **Concatenación:**

Para concatenar String se utilizará el operador `“.”` (Punto), con un espacio entre medio para mejorar la lectura.

- **Control de flujo:**

En las declaraciones `if/then/else` deberá tener un espacio antes y después del paréntesis condicional, lo mismo se aplica al `elseif`, a continuación un ejemplo que lo ilustra.

- **Documentación:**

Las complicadas funciones y métodos deberán tener un bloque de documentación. El mismo será entre `/**/` cuando sean de 2 líneas en adelante y `//` cuando sea una sola línea [30].

Este framework de desarrollo, obliga al programador desarrollar códigos legibles y fáciles de entender y mantener. Por esta razón se hará uso del estándar de codificación definidos por la comunidad de desarrollo de Symfony y citados anteriormente.

2.3. Paradigma de Programación.

2.3.1. Programación Orientada a Objeto.

La programación orientación a objetos (POO) es un paradigma de programación que facilita la creación de software de calidad por sus factores que potencian el mantenimiento, la extensión y la reutilización del software generado bajo este paradigma. La POO trata de amoldarse al modo de pensar del hombre y no al de la máquina. Esto es posible gracias a la forma racional

con la que se manejan las abstracciones que representan las entidades del dominio del problema, y a propiedades como la jerarquía o el encapsulamiento.

El elemento básico de este paradigma no es la función (elemento básico de la programación estructurada), sino una entidad denominada objeto. Un objeto es la representación de un concepto para un programa, y contiene toda la información necesaria para abstraer dicho concepto: los datos que describen su estado y las operaciones que pueden modificar dicho estado, y determinan las capacidades del objeto [31].

Características

Las características que tiene la POO son:

Abstracción: Es una descripción simplificada o especificación de un sistema que enfatiza algunos de los detalles o propiedades del sistema, mientras suprime otros.

Encapsulación: Es el proceso de ocultar todos los detalles de un objeto que no contribuyen a sus características esenciales.

Modularidad: Es la propiedad de un sistema que ha sido descompuesto en un conjunto de módulos coherentes e independientes.

Jerarquía o herencia: Es el orden de las abstracciones organizado por niveles.

Tipificación: Es la definición precisa de un objeto de tal forma que objetos de diferentes tipos no puedan ser intercambiados o, cuando mucho, puedan intercambiarse de manera muy restringida.

Concurrencia: Es la propiedad que distingue un objeto que está activo de uno que no lo está.

Persistencia: Es la propiedad de un objeto a través de la cual su existencia trasciende el tiempo (es decir, el objeto continúa existiendo después de que su creador ha dejado de existir)

y/o el espacio (es decir, la localización del objeto se mueve del espacio de dirección en que fue creado).

Polimorfismo: Comportamientos diferentes, asociados a objetos distintos, pueden compartir el mismo nombre, al llamarlos por ese nombre se utilizará el comportamiento correspondiente al objeto que se esté usando. O dicho de otro modo, las referencias y las colecciones de objetos pueden contener objetos de diferentes tipos, y la invocación de un comportamiento en una referencia producirá el comportamiento correcto para el tipo real del objeto referenciado. Cuando esto ocurre en "tiempo de ejecución", esta última característica se llama asignación tardía o asignación dinámica. Algunos lenguajes proporcionan medios más estáticos (en "tiempo de compilación") de polimorfismo, tales como las plantillas y la sobrecarga de operadores de C++.

Herencia: Las clases no están aisladas, sino que se relacionan entre sí, formando una jerarquía de clasificación. Los objetos heredan las propiedades y el comportamiento de todas las clases a las que pertenecen. La herencia organiza y facilita el polimorfismo y el encapsulamiento, esto permite a los objetos ser definidos y creados como tipos especializados de objetos preexistentes. Estos pueden compartir (y extender) su comportamiento sin tener que volver a implementarlo. Esto suele hacerse habitualmente agrupando los objetos en clases y estas en árboles que reflejan un comportamiento común. Cuando un objeto hereda de más de una clase se dice que hay herencia múltiple [32].

Ventajas de la POO

La POO brinda grandes ventajas para el desarrollo del software ya que fomenta la reutilización y extensión del código, permitiendo crear sistemas más complejos. Este paradigma permite que se relacione el sistema al mundo real, facilita la creación de programas visuales y la construcción de prototipos. Agiliza el desarrollo de software, facilita el trabajo en equipo y el mantenimiento del software [33].

2.4. Lenguaje de Programación.

2.4.1. PHP.

PHP (Hypertext Preprocessor) es un lenguaje script para el desarrollo de páginas web dinámicas del lado del servidor, cuyos fragmentos de código se intercalan fácilmente en páginas HTML, debido a esto, y a que es de Open Source (código abierto), es el más popular y extendido en la web [34].

Características

Entre las características primordiales de PHP se tiene que:

- Presenta soporte para una gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, SybasemSQL, Informix, entre otras.
- Se integra con varias bibliotecas externas, permitiendo generar documentos en PDF (documentos de Acrobat Reader) y analizar código XML.
- Ofrece una solución simple y universal para las paginaciones dinámicas de la web de fácil programación.
- Perceptiblemente más fácil de mantener y poner al día, que el código desarrollado en otros lenguajes.
- Soportado por una gran comunidad de desarrolladores, como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.
- Con PHP se puede hacer cualquier cosa que podemos realizar con un script CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas [35].

Luego de un amplio estudio de los distintos lenguajes de programación web por parte del arquitecto de software, se determina para programar el Subsistema de Manejo de Archivo utilizar PHP como lenguaje de programación del lado del servidor. Además PHP tiene la

capacidad de conectarse a la mayoría de los motores de bases de datos que se utilizan en la actualidad, es multiplataforma y otro motivo de su uso es que permite aplicar técnicas de programación orientada a objetos.

2.5. Entorno Integrado de Desarrollo.

2.5.1. Zend Studio.

Se trata de un programa orientado a desarrollar aplicaciones web, además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. El programa entero está escrito en Java, lo que permite a Zend lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS.

Zend Studio consta de dos partes en las que se dividen las funcionalidades, la parte del cliente y la del servidor. La del cliente contiene la interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP [36].

Características

Zend Studio presenta las siguientes características:

- Soporte para PHP 4 y PHP 5.
- Resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase.
- phpDocintegrado.
- Plegado de código (comentarios, bloques de phpDoc, cuerpo de funciones y métodos e implementación de clases).
- Inserción automática de paréntesis y corchetes de cierre.
- Sangrado automático y otras ayudas de formato de código.
- Emparejamiento de paréntesis y corchetes.

- Detección de errores de sintaxis en tiempo real.
- Funciones de depuración: Botón de ejecución y traza, marcadores, puntos de parada (*breakpoints*), seguimiento de variables y mensajes de error del intérprete de PHP. Permite también la depuración en servidores remotos (requiere ZendPlatform).
- Instalación de barras de herramientas para Internet Explorer y Mozilla Firefox (opcional).
- Soporte para la gestión de grandes proyectos de desarrollo.
- Manual de PHP integrado.
- Soporte para control de versiones usando CVS o Subversión (a elección del desarrollador).
- Cliente FTP integrado.
- Soporte para navegación en bases de datos y ejecución de consultas SQL [37].

Se acuerda utilizar Zend Studio como IDE por incluir todos los componentes necesarios durante el ciclo de vida de una aplicación en PHP, editor, análisis, depuración, optimizadores de código y herramientas de base de datos. Zend Studio permite agilizar el desarrollo web y simplificar proyectos complejos. Es un IDE muy completo ya que acelera nuestro trabajo, cuenta con un buen depurador e infinitas opciones que permiten un desarrollo profesional de nuestras aplicaciones.

2.6. Sistema Gestor de Base de Datos.

Un Sistema Gestor de Base de Datos (SGBD) es un software que permite la utilización y/o la actualización de los datos almacenados en una o varias Base de datos [38]. Por lo tanto debe permitir definir una base de datos: especificar tipos, estructuras y restricciones de datos, construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD y manipular la base de datos: realizar consultas, actualizarla, generar informes [11].

2.6.1. Postgres SQL.

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Soporta consultas (queries) complejas, incluyendo subselects,

integridad referencial (llave foránea), disparadores (triggers), vistas (views), integridad Transaccional (ACID) y control de versionado concurrente (MVCC). El desarrollo de PostgreSQL es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (Grupo de Desarrollo Global de PostgreSQL) [11].

Es el SGBD de código abierto más potente del mercado, es seguro e incluye los siguientes principios de integridad: Atomicidad, Consistencia, Aislamiento y Durabilidad [39].

Características

Entre las principales características que presenta el sistema gestor de base de datos se encuentran:

- El modelo es de cliente/servidor (hoy en día el más común, pero no el único).
- Su lenguaje procedimental es muy similar al PL de Oracle, logrando una migración más amena.
- Se adapta al estándar SQL: 2003.
- Posee Multi-Versión de Control de Concurrencia MVCC. Fue una de las pioneras (la primera fue InterBase).
- Herencia de tablas.
- Puntos de recuperación avanzados (savepoints, replicación asincrónica).
- Optimizador de consultas.
- Juegos de caracteres UNICODE e internacionalización.
- Restricciones, evita eliminaciones accidentales.
- Transacciones BEGIN – END- SAVEPOINTS.
- Permite anidar consultas avanzadas.
- Conexiones encriptadas vía SSL.
- Dominios, archivos de configuración (clustering), catálogos (tablespaces), esquemas (schemas) y otros objetos de agrupación.
- TOAST (atributos comprimidos largos, transparente al usuario).
- Soporte de funciones de ventana y recursivas [39].

Se utiliza PostgreSQL como gestor de base de datos porque es un sistema estable, de alto rendimiento, 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 ya las existentes.

2.7. Servidor Web.

Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados [40].

2.7.1. Apache.

Apache es uno de los mejores y el más utilizado entre los servidores web existentes gracias a su gran estabilidad y confiabilidad.

Características

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita de código abierto.
- Es un servidor altamente configurable de diseño modular.
- Trabaja con gran cantidad de lenguajes como Perl, PHP y otros de script.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.
- Es configurable en la creación y gestión de log.
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que se destaca PHP, un lenguaje de programación del lado del servidor [41].

2.8. Técnicas de Aseguramiento de la Calidad.

La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, portabilidad, usabilidad, seguridad e integridad. El aseguramiento de calidad del software es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza que el producto (software) requiere para satisfacer los requerimientos dados de calidad por parte del cliente. Existen varias técnicas para realizar esta labor:

- **Prácticas colaborativas de desarrollo:** Son aquellas actividades que, realizadas por equipos de desarrollo y sin poner en juego jerarquías, se utilizan para mejorar la calidad del desarrollo de software.
- **Repositorios compartidos:** Es fundamental contar con un repositorio único de código y documentación. Un repositorio centralizado debe tener, al menos, funcionalidades para poder actualizar código fuente de más de un origen y dar marcha atrás en caso de necesitarlo, hacia cualquier versión anterior.
- **Refactorizaciones:** Tienen que ver con el cambio de un diseño una vez que se desarrolló una funcionalidad y la misma ya está codificada. Las refactorizaciones buscan mejorar el diseño del código ya escrito sin cambiar el comportamiento observable o externo. Se hacen para:
 - Mejorar código, haciéndolo más comprensible y legible, recordando que el código se lee mucho más de lo que se lo escribe.
 - Eliminar duplicaciones de código o de comportamiento, para que cada cambio afecte una sola porción de código.
 - Mantener alta la calidad del diseño.
- **Pruebas de aceptación parciales:** Para tener un producto lo más deseable posibles fundamental contar con las pruebas de aceptación. Estas pruebas deben ser escritas por personas que desconocen el funcionamiento interno del sistema y asistidas por gente especializada en pruebas [42].

Una vez explicadas las técnicas de aseguramiento de la calidad del código, los repositorios compartidos permiten que los desarrolladores del proyecto trabajen simultáneamente sobre los archivos de forma distribuida. Los repositorios compartidos se encargan de unificar las modificaciones que puedan hacer cada uno de los desarrolladores creando un historial de la evolución del desarrollo del sistema.

Las pruebas de aceptación permiten que el cliente evalúe el funcionamiento de la aplicación en un periodo determinado y así ir corrigiendo los errores encontrados.

2.9. Conclusiones Parciales.

Una vez descritas las técnicas, tecnologías y herramientas a utilizar para lograr el objetivo planteado, se concluye que el uso del framework Symfony, permite simplificar en costo de tiempo y esfuerzo el desarrollo del módulo de Manejo de Archivo. Establecer estilos y estándar de codificación, ayuda a un mejor entendimiento del código para próximas versiones y para lograr una mayor libertad tecnológica se ha tenido en cuenta los principios de la comunidad de software libre.

Capítulo 3: Enfoque y Validación de la solución propuesta.

3.1 Introducción.

En el presente capítulo se abordan los principales elementos que contribuyeron a la construcción de la solución propuesta. De esta manera se definen los requisitos funcionales y no funcionales del subsistema propuesto, se complementa el modelo de implementación y se describen las funciones de las librerías utilizadas para el trabajo con archivos. También se verifica el comportamiento interno del sistema y se validan los requisitos establecidos.

3.2 Requisitos.

Según la IEEE (Standard Glossary of Software Engineering Terminology) un requisito es la condición o capacidad que debe poseer un sistema o un componente de un sistema para satisfacer un contrato, un estándar, una especificación u otro documento formalmente impuesto [43]. Se puede definir que un requisito son todas las ideas que los miembros de un equipo o un cliente tenga acerca de lo que debe hacer un sistema o como debe funcionar. Son una descripción de las necesidades o deseos de un producto.

Los requisitos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas [44]. Los requisitos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento [44].

Los Requisitos Funcionales (RF) y los Requisitos no Funcionales (RNF) que se explican en el siguiente epígrafe son los pertenecientes a la implementación del subsistema.

3.2.1 Requisitos Funcionales.

- **RF1:** El subsistema debe permitir que los usuarios puedan cargar ficheros en formato txt y xls. Este requisito se ajusta a la necesidad de importar los datos de la inclinometría (surveys³) tanto del proyecto como de la perforación real.
- **RF2:** El subsistema debe permitir que los usuarios exporten ficheros en formato txt y xls [45]. Este requisito se refiere a la necesidad de exportar el Parte Diario de Perforación, el Reporte Diario de Perforación, el Reporte de Geología, así como la inclinometría plan y real de los pozos en perforación.

Ambos requisitos deben ser implementados ajustándose a las características de cada uno de los entornos donde se despliega el sistema (Pozo y DIPP). De esta manera se garantiza una mayor personalización y por tanto más usabilidad para el usuario final.

3.2.2 Requisitos No Funcionales.

RNF: Seguridad

- **Confidencialidad:** La información manejada por el sistema está protegida de acceso no autorizado y divulgación.
- **Integridad:** La información manejada por el sistema es objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma es considerada igual a la fuente o autoridad de los datos.
- **Disponibilidad:** Los usuarios autorizados (autenticados por dominio y 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.

³ Archivo que contiene datos de la dirección del pozo



RNF: Rendimiento

- El sistema debe responder en un tiempo relativamente rápido a las peticiones del usuario (menos de 3 segundos).
- El sistema necesita las PC clientes con 256 de RAM y un navegador que soporte Java script.
- El sistema necesita un servidor de base de datos de (4 o superior) gigas de RAM.

RNF: Interfaz

- Interfaces amigables, fáciles de interactuar con ellas.

RNF: Hardware (Servidos Web)

- Procesador Pentium D 2x2 caché. 3.2 GHz.
- Memoria RAM \geq 4 – 8 GB.

RNF: Hardware (Cliente Web)

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

RNF: Software

- Sistema Operativo Windows 2000/XP/Vista y Linux.
- Lenguaje de programación PHP.
- Entorno de desarrollo Zend Studio [45].

3.3 Diagrama de Caso de Uso del Sistema.

Un diagrama de Caso de Uso del Sistema (DCUS) describe lo que hace un sistema desde el punto de vista de un observador externo, debido a esto, un diagrama de este tipo generalmente es de los más sencillos de interpretar en UML, ya que su razón de ser se concentra en un Qué hace el sistema.

El diagrama de Caso de Uso del Sistema (DCUS) ayuda al cliente, los usuarios y los desarrolladores a llegar a un acuerdo sobre cómo utilizar el sistema. La mayoría de las aplicaciones tienen muchos tipos de usuarios. Cada tipo de usuario se representa mediante un actor. Los actores utilizan el sistema al interactuar con los casos de uso [17].

3.3.1 Diagrama de Caso de Uso del Sistema.

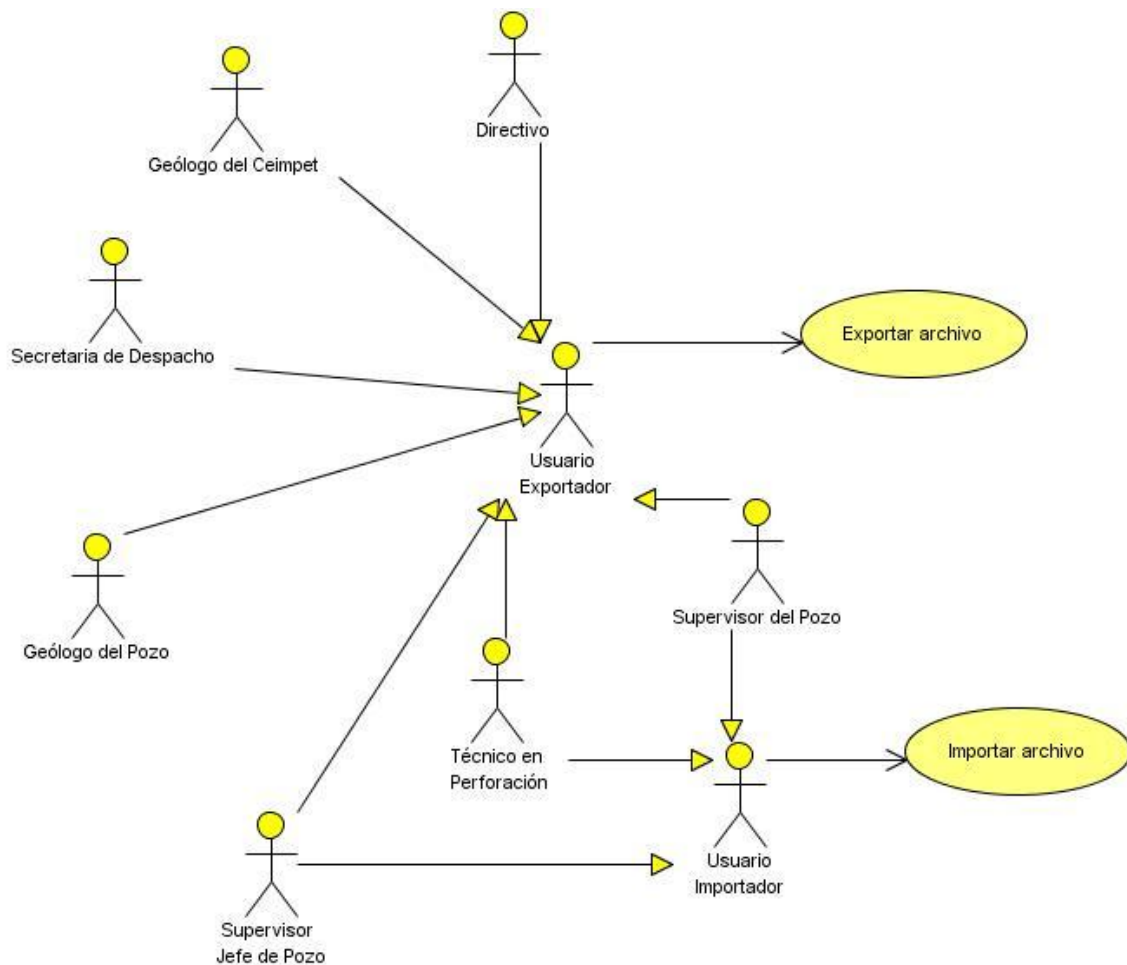


Diagrama 1: Caso Uso del Sistema

3.3.2 Los actores del sistema.

No todos los actores representan a personas. Pueden ser actores otros sistemas o hardware externo que interactuará con el sistema. Cada actor asume un conjunto coherente de papeles

cuando interactúa con el sistema. Un usuario físico puede actuar como uno o varios actores, desempeñando los papeles de esos actores en su interacción con el sistema. Los actores se comunican con el sistema mediante el envío y recepción de mensajes hacia y desde el sistema según éste lleva a cabo los casos de uso a medida que se define lo que hacen los actores y lo que hacen los casos de uso, se realiza una clara separación entre las responsabilidades de los actores y las del sistema. Esta separación ayuda a delimitar el alcance del sistema [17].

3.3.2.1 Descripción de los Actores.

Actor	Descripción
<p align="center">Geólogo del Cienpet</p>	<p>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.</p>
<p align="center">Secretaría de Despacho</p>	<p>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.</p>
<p align="center">Geólogo del Pozo</p>	<p>Encargado de elaborar el Reporte Diario de Geología del Pozo. Este se entrega al supervisor del pozo para que posteriormente se elabore el reporte diario de perforación del pozo.</p>
<p align="center">Supervisor del Pozo</p>	<p>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.</p>
<p align="center">Directivo</p>	<p>Este actor representa a los directivos de la</p>

	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 2: Descripción de los actores del sistema

3.3.3 Descripción de los casos de uso.

Exportar Archivo

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	RF 2
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 introducir el nombre del archivo y la extensión con la que será salvado.
2. El usuario introduce el nombre, selecciona la extensión.	
3. El usuario selecciona la opción de guardar.	3.1 El sistema exporta el archivo y muestra un mensaje del éxito de la operación.

Prototipo de Interfaz (ver anexo 9)

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

Importar Archivo

Caso de Uso:	Importar Archivo
Actores:	Usuario
Resumen:	El caso de uso se inicia cuando el actor necesita importar 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	RF 1
Prioridad	Crítico

Flujo Normal de Eventos

Sección “Buscar Información Específica”

Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la Opción “Importar Archivo”.	1.1 El sistema muestra una ventana de “búsqueda” para seleccionar el archivo a importar por el sistema.
2. El usuario selecciona el archivo.	2.1 El sistema verifica que la extensión del archivo sea soportada. 2.2 El sistema envía un mensaje del éxito de la operación.

Flujos Alternos

Acción del Actor	Acción del Actor
2a. Error al cargar el archivo	2a.1 Muestra un mensaje: “Extensión del archivo no soportada”

Prototipo de Interfaz (ver anexo 10)

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

3.4 Modelo de Implementación.

El modelo de implementación describe cómo los elementos del modelo de diseño se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructura y módulos Citados en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y como dependen los componentes unos de otros [46].

3.4.1 Vista de implementación.

La vista de implementación describe como se implementan los componentes físicos mostrados en vistas de distribución agrupándolos en subsistemas organizados en capas y jerarquías y muestra además las dependencias entre éstos. Básicamente, se describe el mapeo desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos [47].

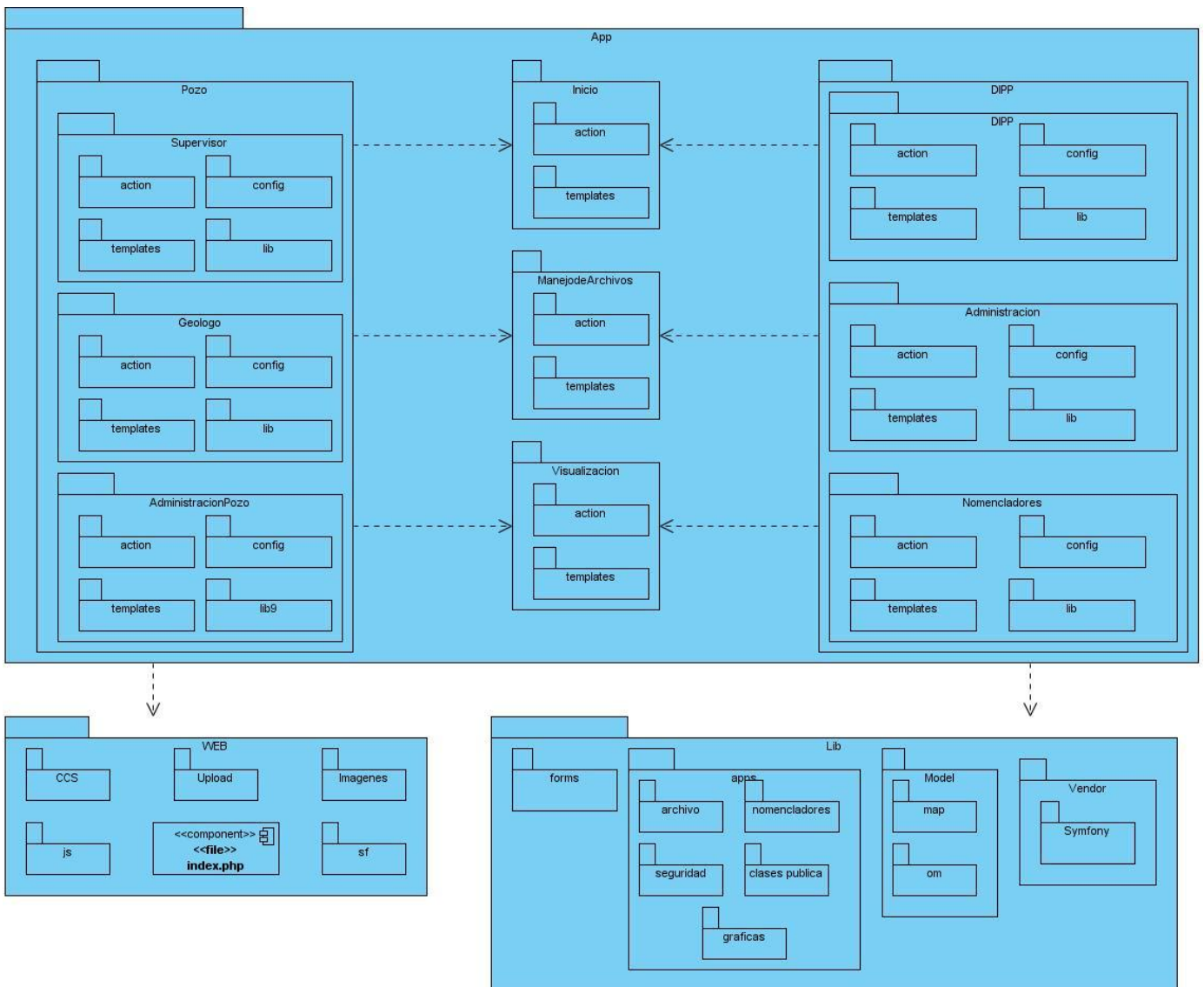


Diagrama 2: Vista de Implementación

En el modelo de implementación se puede apreciar la relación existente entre los elementos del sistema. Como los módulos del Pozo y de la DIPP hacen uso de maneras sincrónicas del subsistema de Manejo de Archivos a través de las plantillas (templates) definidas en el mismo, y se observa como el subsistema de manejo de archivo hace uso de las clases definidas dentro de los paquetes Weby Lib.

3.4.2 Diagrama de Componente.

Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases del modelo de diseño. Durante la creación de componentes en un entorno de implementación particular, éstos pueden tener varios estereotipos para reflejar el significado real de los componentes. Los componentes presentan las siguientes características:

- Tienen relaciones de traza con los elementos del modelo de implementación.
- Es normal que un componente implemente varios elementos como por ejemplo varias clases.
- Los componentes proporcionan las mismas interfaces que los elementos del modelo que implementan.
- Puede haber dependencia de compilación entre componentes, denotando que componentes son necesarios para compilar un componente determinado [11].

3.4.2.1 Descripción general de los componentes.

1. Java Script.



Es el fichero donde se crean las funciones para la creación de campos dinámicos y se realizan las validaciones de los datos entrados.

2. CSS



Son los ficheros que le dan formato de diseño a la aplicación. Son el estilo de la aplicación.

3. ActionClass



Engloban toda la lógica de la aplicación. Son las clases controladoras y se encargan de interactuar con el modelo y ejecutar las vistas.

4. ModelClass



Se encargan del acceso a datos (Peer) y contienen la lógica del negocio.

5. Propel



Symfony utiliza Propel como ORM (Mapeo de Objetos Relacional) y Propel utiliza Creole como capa de abstracción de Base de Datos.

3.4.2.2 Diagramas de componentes del subsistema Manejo de Archivos.

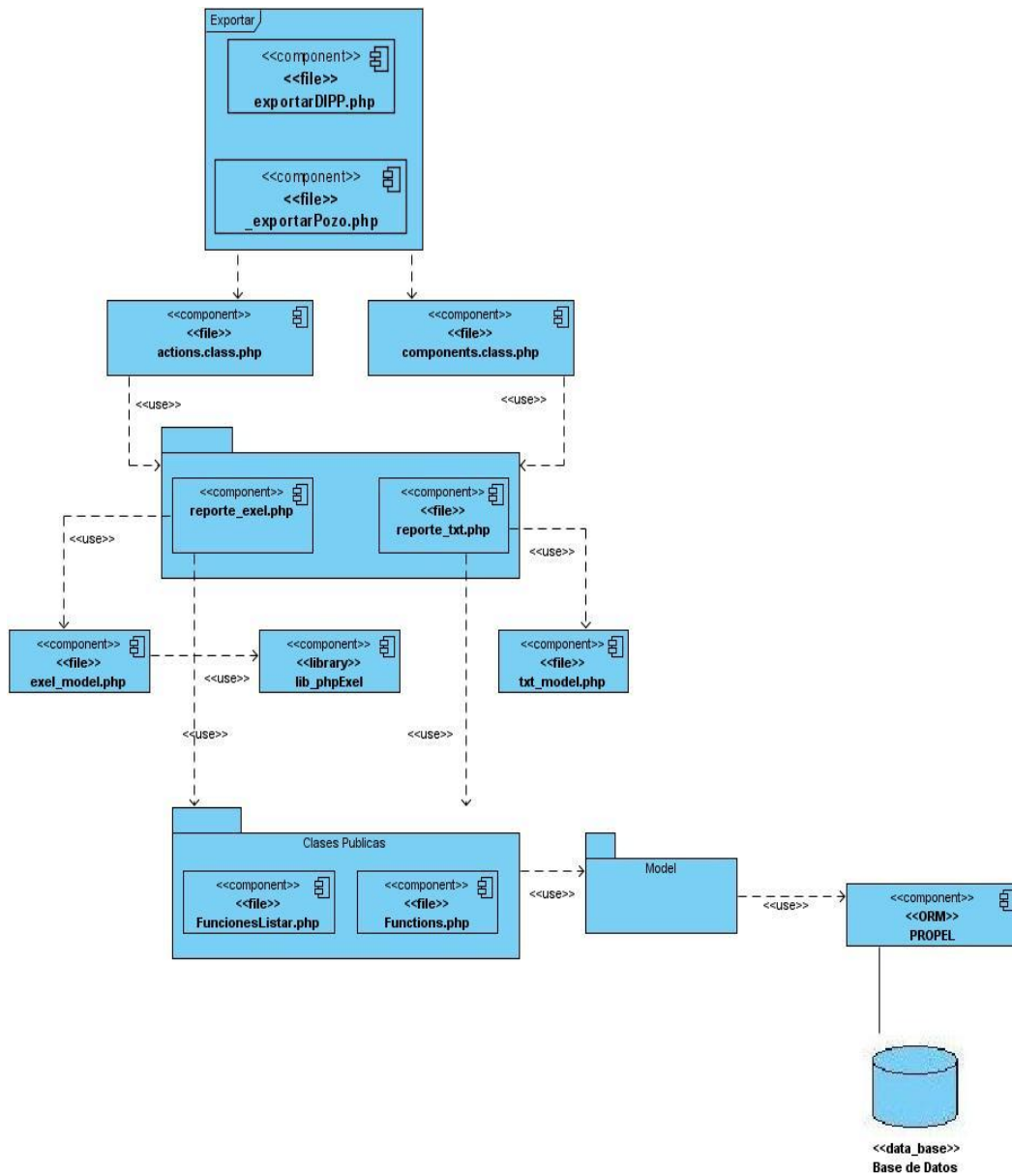


Diagrama 3: Exportar Archivo

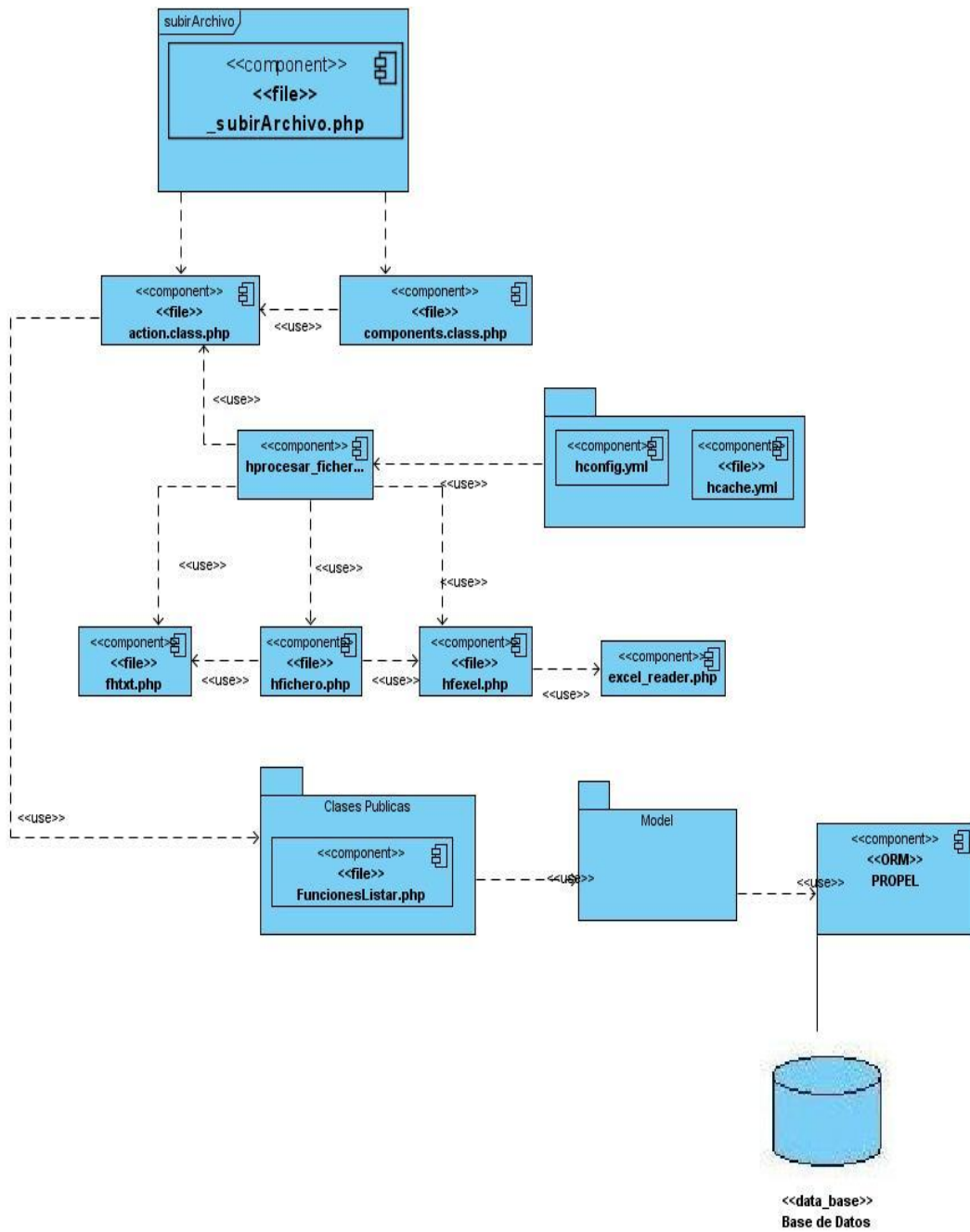


Diagrama 4: Importar Archivo

3.5 Manejo de Archivo.

La lectura de archivo en PHP, se realizó utilizando una librería llamada *Excel_Reader* la cual permite la carga de ficheros de texto plano y hojas de cálculos. Mientras que la escritura se efectuó de dos formas, para los archivos .txt se utilizó las funciones de exportación que incluye el lenguaje definido y la de los archivos .xls se realizó mediante la librería *PHPExcel*.

Principales características de PHPExcel:

- Meta datos (autor, título, descripción,...).
- Agrega más hojas de cálculo (pestañas).
- Inserta valores y formulas a celdas individuales.
- Combinar celdas.
- Proteger rango de celdas con contraseña.
- Poder establecer alto y ancho de celdas.
- Poder establecer diseño a celdas (fuente, borde, fondo,...).
- Soporte para fijar celdas.
- Poder agrupar celdas/columnas.
- Insertar o borrar celdas/columnas.
- Asignar nombre a un rango.
- Soporte para referencias entre pestañas.
- Agregar imágenes (con la posibilidad de cambiar su estilo: sombra, posición,..).
- Agregar comentarios a una celda.
- Opciones de impresión.
- Opciones de salida en distintos formatos (Excel 2007, BIFF8 (Excel 97 y superior), CSV (Comma Separated Values), HTML, PDF).
- Opciones de lectura en distintos formatos (Excel 2007, BIFF5 (Excel 5.0 / Excel 95), BIFF8 (Excel 97 o superior), CSV (Comma Separated Values) [48].

Entre las principales funcionalidades utilizadas para la salida de archivos están:



Para los ficheros xls (hojas de cálculos):

- ***excel_open(\$direccion)***: Esta función dado el directorio del servidor donde se cargara el archivo, retorna un objeto PHPExcel (Una hoja de cálculo).
- ***value_cell (\$cell, \$value)***: Permite escribir en la hoja de cálculo pasándole por parámetros, la celda a partir de donde se va a escribir y que es lo que se va a escribir como tal.
- ***exportar (\$direccion, \$name)***: Método que salva el Excel en una dirección y con un nombre especificado por el usuario.

Para los ficheros txt (procesadores de texto plano):

- ***read (\$filename)***: Método que permite abrir un fichero.
- ***write (\$filename,\$texto)***: Esta función permite escribir en el fichero.
- ***exportar (\$upload_dir,\$name)***: Posibilita guardar el fichero en una dirección y con un nombre especificado por el usuario.

Para importar archivo se utiliza la función:

- ***procesar_fichero (\$dir)***: Esta función permite cargar un archivo ya sea txt o xls, a través de una dirección.

Una vez desarrollado el subsistema de manejo de archivos, al integrarse con la solución SIPP actúa como un componente aplicado a los dos entornos del negocio. Para exportar archivos en el ambiente de la DIPP funciona con las variables (tipo, formato, pozo, reporte, fecha), estas permiten que se obtengan los reportes y partes con la información deseada en formato txt y xls de los datos que se generan durante la perforación.

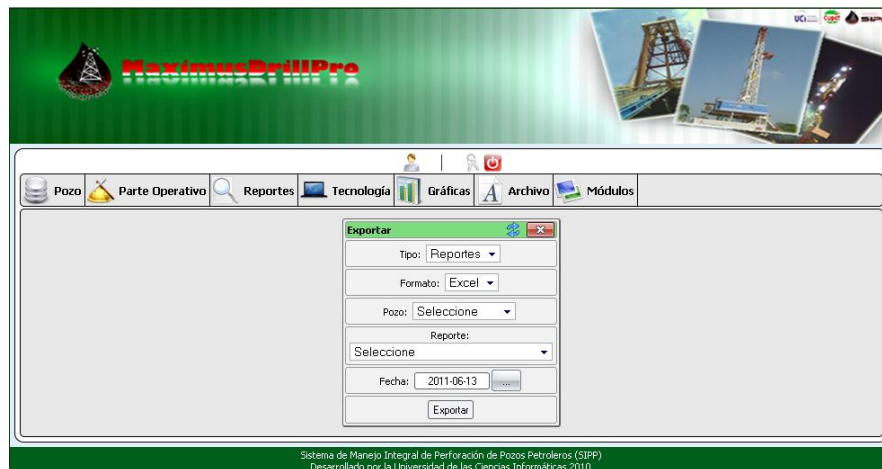


Fig.1: Interfaz de Exportar en la DIPP

En el ambiente del Pozo se manejan las variables (formato, reporte, fecha), las cuales posibilitan obtener en hojas de cálculos y en un procesador de texto plano la información referente al pozo que se está perforando. Y para importar Inclínometría, se examina la dirección del archivo que se quiere importar, se sube al servidor y una vez allí se puede modificar y procesar. Esta funcionalidad sigue el mismo comportamiento en ambos subsistemas, DIPP y Pozo.

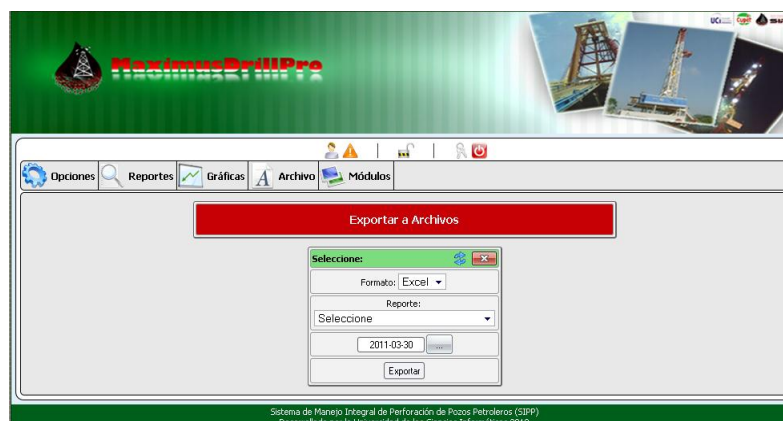


Fig. 2: Interfaz de Exportar en el Pozo

3.6 Elementos de la arquitectura.

La arquitectura es una vista estructural de alto nivel, que define estilo o combinación de estilos para una solución. Se puede decir que la arquitectura es esencial para el éxito o fracaso de un proyecto. Además la arquitectura de un software es necesaria para comprender el sistema, organizar el desarrollo del mismo, fomentar la reutilización y controlar la evolución del proyecto [19].

3.6.1 Patrón Arquitectónico.

Un patrón arquitectónico no es más que un esquema genérico probado para solucionar un problema en particular, el cual es recurrente dentro de cierto contexto. Este esquema se especifica describiendo los componentes, con sus responsabilidades y relaciones [49].

Para la construcción del subsistema se utilizó el patrón Modelo-Vista-Controlador (MVC), el cual permite separar la aplicación en tres componentes, la capa de presentación, la lógica del negocio y el acceso a datos. Las vistas (capa de presentación) representan la lógica por la cual los datos son manipulados para visualizarlos o para emitir reportes, es la interfaz mediante la cual el modelo muestra los datos al usuario. El controlador (lógica del negocio) es el que se encarga de controlar todas las interacciones entre el modelo y sus vistas, traduce las interacciones de los usuarios a acciones que serán implementadas por el modelo. El modelo (acceso a datos) representa los datos, las operaciones y la lógica del negocio de la aplicación [49].

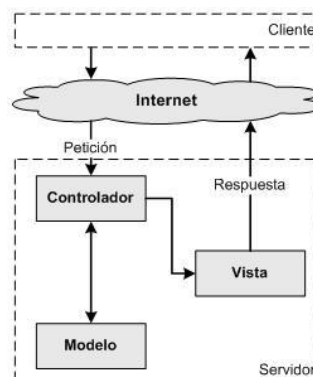


Fig. 3: Patrón Arquitectónico MVC

3.6.2 Patrones de Diseño.

Los patrones de diseños no son más que un par de problema-solución a través de los cuales se hacen los códigos de las aplicaciones existentes reutilizables. Esto permite que se puedan aplicar a diferentes problemas de diseño en distintas situaciones [50]. Los Patrones de Software Generales para la Asignación de Responsabilidad (GRASP) describen los principios fundamentales de diseño de objetos para otorgar responsabilidad [51].

Seguidamente se describen los patrones que se pusieron en práctica en el diseño de la aplicación.

3.6.2.1 Patrón Experto.

Este patrón es el principio básico de asignación de responsabilidades. El mismo indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo se logra un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento). El mismo mantiene el encapsulamiento, los objetos utilizan su propia información para llevar a cabo sus tareas. Se distribuye el comportamiento entre las clases que contienen la información requerida. Son más fáciles de entender y mantener [52].

Este patrón se evidencia en las clases *reporte_excel* y *reporte_txt* que son expertas ya que contiene la información necesaria para concebir la estructura de los reportes en hojas de cálculos y procesadores de texto plano.

3.6.2.2 Patrón Bajo Acoplamiento.

Este patrón asigna una responsabilidad para mantener el bajo acoplamiento que no es más que tratar de que una clase no dependa de muchas otras, minimizando la dependencia entre las clases y facilita la reutilización. Decir que este patrón no puede verse separado del patrón Experto y del patrón Alta Cohesión. Al ponerse en práctica este patrón se reduce el impacto de cualquier cambio, se pueden comprender los objetos de manera aislada [52].

Este patrón se pone de manifiesto en las clases *reporte_excel*, *reporte_txt* y *hfichero*.

3.6.2.3 Patrón Alta Cohesión.

Este patrón define que una clase tiene responsabilidades moderadas en un área funcional y colabora con otras clases para llevar a cabo las tareas. Una clase con alta cohesión tienen un número relativamente pequeño de métodos, con funcionalidad altamente relacionada, y no realiza mucho trabajo. Colabora con otros objetos para compartir el esfuerzo si la tarea es extensa. Una clase con alta cohesión es ventajosa porque es relativamente fácil de mantener, entender y reutilizar.

Este patrón permite que las clases realicen trabajos más específicos, que sean más fáciles de entender y que los cambios que se realicen en ellas no afecten tanto [52].

3.6.2.4 Patrón Controlador.

El patrón controlador sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto es para aumentar la reutilización de código y a la vez tener un mayor control. Se recomienda dividir los eventos del sistema en el mayor número de controladores para poder aumentar la cohesión y disminuir el acoplamiento [52].

3.7 Pruebas del Subsistema Manejo de Archivo.

Las pruebas son elementos críticos para la calidad del software. La importancia de los costos asociados a los errores, promueve la definición y aplicación de un proceso de pruebas minuciosas y bien planificadas. Las pruebas permiten validar y verificar el software entendiendo como validación del software el proceso, externo al equipo de desarrollo, que determina si el software satisface los requisitos, y verificación como el proceso interno que determina si los productos de una fase satisfacen las condiciones de dicha fase [53].



El software puede probarse de distintas formas. Las pruebas que se pusieron en práctica para aprobar el correcto funcionamiento del subsistema de manejo de archivos, fueron las llamadas pruebas de caja negra y las pruebas de aceptación por parte del cliente de SIPP.

En los siguientes epígrafes se explica en qué consiste cada una de ellas.

3.7.1 Prueba de Caja Negra.

Las pruebas de caja negra son las realizadas por los usuarios, y son las que miden el correcto funcionamiento y cumplimiento de los requisitos definidos para el sistema una vez concluido. Se basan en encontrar errores en cinco categorías, estas son: (Funciones incorrectas o ausentes, Errores de Interfaz, Errores en estructuras de datos o en accesos a bases de datos externas, Errores de rendimiento y Errores de inicialización y terminación). Por lo tanto estas pruebas constituyen la validación de la aplicación.

Pressman plantea lo siguiente:

La validación del software se consigue mediante una serie de pruebas de caja negra que demuestran la conformidad con los requisitos. Un plan de prueba traza la clase de pruebas que se han de llevar a cabo, y un procedimiento de prueba define los casos de prueba específicos en un intento por descubrir errores de acuerdo con los requisitos. Tanto el plan como el procedimiento estarán diseñados para asegurar que se satisfacen todos los requisitos funcionales, que se alcanzan todos los requisitos de rendimiento, que la documentación es correcta e inteligible y que se alcanzan otros requisitos (por ejemplo, portabilidad, compatibilidad, recuperación de errores, facilidad de mantenimiento) [53].

3.7.2 Prueba de Aceptación.

La prueba de aceptación del usuario es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido [54].

Esta prueba puede aplicarse durante semanas o meses y puede ir desde un informal “Paso de Prueba” hasta la ejecución sistemática de una serie de pruebas bien planificadas [53]. Con el objetivo de descubrir los errores que aparezcan se encuentran las pruebas Alfa y Betas.

Prueba Alfa: Se lleva a cabo, por un cliente, en el lugar de desarrollo. Se usa el software de forma natural con el desarrollador como observador del usuario y registrando los errores y los problemas de uso. Las pruebas alfa se llevan a cabo en un entorno controlado [53].

Prueba Beta: Se lleva a cabo por los usuarios finales del software en los lugares de trabajo de los clientes. A diferencia de la prueba alfa, el desarrollador no está presente normalmente. Así, la prueba beta es una aplicación “en vivo” del software en un entorno que no puede ser controlado por el desarrollador. El cliente registra todos los problemas (reales o imaginarios) que encuentra durante la prueba beta e informa a intervalos regulares al desarrollador [53].(Anexo 11)

3.7.3 Descripción de los Casos de Pruebas.

El objetivo principal de los casos de pruebas es conseguir un conjunto de pruebas que tengan la mayor probabilidad de detectar errores en el software. Un caso de prueba es un conjunto de entradas de pruebas, condiciones de ejecución y resultados esperados desarrollados para verificar una función esperada. A continuación se muestran los casos de pruebas diseñados para probar las funcionalidades del software:

ID del escenario	Escenario	Variable 1 Formato	Variable 2 Reporte	Variable 3 Fecha	Respuesta del sistema	Resultado de la prueba
1.1	Exportar archivo con reporte de Inclinometría	V EXEL	V “Inclinometría Real”	N/P	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. (Ruta) <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio

		V TXT	V "Inclinometría Real"	N/P	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta)</i> <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V EXEL	V "Inclinometría Plan"	N/P	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta)</i> <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V TXT	V "Inclinometría Plan"	N/P	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta)</i> <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
1.2	Exportar archivo, sin especificar Reporte	V EXCEL	I vacío	NP	El sistema muestra en una nueva ventana el mensaje "Debe seleccionar el campo "Reporte", aceptar el mensaje, este se cierra y se mantiene la interfaz que le permite al usuario seleccionar reporte.	Satisfactorio
1.3	Exportar archivo con otro reporte	V EXCEL	V Reporte Diario de Geología.	V 30/11/2010	El Sistema muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta)</i>	Satisfactorio



CAPÍTULO 3: ENFOQUE Y VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

					<i>El Sistema cierra la ventana y exporta el archivo.</i>	
		V EXCEL	V Reporte Diario de Perforación	V 20/11/2010	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. (<i>Ruta</i>) <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V EXCEL	V Reporte operativo de perforación oficial	V 30/11/2010	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. (<i>Ruta</i>) <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V EXCEL	V Reporte operativo de perforación oficial 4:00pm	V 25/11/2010	El Sistema desactiva el campo Fecha. Muestra un mensaje de confirmación Muestra una nueva ventana para guardar el archivo que se desea exportar. (<i>Ruta</i>) <i>El Sistema cierra la ventana y exporta el archivo.</i>	Satisfactorio
EC1.4	Cancelar la operación de exportar.	V TXT	V Reporte operativo de perforación oficial 4:00pm	V 2/12/2010	Se muestra una ventana con el mensaje de confirmación, el sistema cierra la ventana y mantiene la interfaz previa	Satisfactorio
EC1.5	Introducir fechas fuera de los días	V EXCEL	V Reporte operativo	V 2/12/2011	El sistema muestra un mensaje indicando que "La fecha debe ser menor o igual	Satisfactorio



CAPÍTULO 3: ENFOQUE Y VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

	activos del pozo		de perforación oficial 4:00pm		a la fecha final del pozo”	
	V EXCEL	V Reporte operativo de perforación Oficial	V 2/12/2009	El sistema muestra un mensaje indicando que “La fecha debe ser mayor o igual que la fecha final del pozo”	Satisfactorio	

Tabla 3: Caso de Prueba exportar archivo desde el módulo de Pozo.

ID del escenario	Escenario	Variab le 1 Tipo	Variab le 2 Formato	Variab le 3 Pozo	Variab le 4 Reportes	Varia ble 5 Fecha	Respuesta del sistema	Resulta do de la prueba
1.1	Exportar archivo en la DIPP con reporte de Inclínometría	V Reporte	V EXCEL	V Varadero Oeste 1002	V “Inclínometría Real”	N/P	El Sistema oculta el campo Fecha, Muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V Reporte	V TXT	V Varadero Oeste 1002	V “Inclínometría Plan	N/P	.El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el archivo.</i>	Satisfactorio
1.2	Exportar archivo en la DIPP con otro Reporte	V Reporte	V EXCEL	V Varadero Oeste 1002	V Reporte diario de	V 5/11/2010	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el</i>	Satisfactorio



CAPÍTULO 3: ENFOQUE Y VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

					geología		archivo.	
		V Reporte	V EXCEL	V Varadero Oeste 1002	V Reporte diario de perforación	V 9/12/ 2010	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V Reporte	V EXCEL	V Varadero Oeste 1002	V Reporte operativo de perforación oficial	V 19/1 1/20 10	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V Reporte	V EXCEL	V Varadero Oeste 1002	V Reporte operativo perforación4 :00pm	V 12/1 2/20 10	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. <i>(Ruta). Cierra la ventana y exporta el archivo.</i>	Satisfactorio
1.3	Exportar archivo en la DIPP dejando campos vacíos.	V Reporte	V EXCEL	I vacío	V Reporte operativo perforación4 :00pm	V 12/1 2/20 10	El sistema muestra en una nueva ventana el mensaje "Debe seleccionar el campo "Pozo", aceptar el mensaje, este se cierra y se mantiene la interfaz que le permite al usuario seleccionar reporte.	Satisfactorio
		V Reporte	V TXT	V Varadero	I vacío	V 12/1 2/20	El sistema muestra en una nueva ventana el mensaje "Debe seleccionar el campo "Reporte",	Satisfactorio

				Oeste 1002		10	aceptar el mensaje, este se cierra y se mantiene la interfaz que le permite al usuario seleccionar reporte.	
EC1.4	Cancelar la operación de Exportar archivo en la DIPP (Tipo Reporte)	V Reporte	V TXT	V Varadero Oeste 1002	V Reporte operativo perforación oficial 4:00pm	V 2/12/2010	El sistema cierra la ventana y mantiene la interfaz donde se generó el evento.	Satisfactorio

Tabla 4: Caso de Prueba Exportar Archivo desde el módulo de DIPP con la opción Reporte.

ID del escenario	Escenario	Variable 1 Tipo	Variable 2 Partes	Variable 3 Fecha	Respuesta del sistema	Resultado de la prueba
1.1	Exportar archivo en la DIPP con la opción Parte	V Partes	V Parte operativo de perforación oficial	V 11/11/2010	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. (<i>Ruta</i>). <i>Cierra la ventana y exporta el archivo.</i>	Satisfactorio
		V Partes	V Parte operativo de perforación 4:00pm	V 1/12/2010	El Sistema muestra un mensaje de confirmación. Muestra una nueva ventana para guardar el archivo que se desea exportar. (<i>Ruta</i>). <i>Cierra la ventana y exporta el archivo.</i>	Satisfactorio
1.2	Exportar archivo en la DIPP	V Partes	I (vacío)	V 5/11/2010	El sistema muestra en una nueva ventana el mensaje "Debe seleccionar el campo	Satisfactorio



CAPÍTULO 3: ENFOQUE Y VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

	dejando campo Parte vacío				"Parte",	
1.3	Cancelar la acción Exportar archivo en la DIPP (Tipo Parte)	V Partes	V Parte operativo de perforación 4:00pm	V 1/12/2010	El sistema cierra la ventana y mantiene la interfaz donde se generó el evento.	Satisfactorio

Tabla 5: Caso de Prueba Exportar Archivo desde el módulo de DIPP con la opción Partes.

Con la realización de los casos de prueba para cada caso de uso, se procede a la realización de las pruebas de caja negra, las mismas brindarán al cliente seguridad y conformidad con las funcionalidades del sistema.

3.8 Conclusiones Parciales.

Una vez terminado el capítulo correspondiente al resultado de la investigación, en el mismo se representan aspectos como el diagrama de caso de usos del sistema y el modelo de implementación, el cual permite describir, organizar y estructurar los elementos Citados en el entorno de la programación. Se realiza un estudio de los elementos de la arquitectura del subsistema, permitiendo identificar los patrones de diseño presente los cuales son mecanismos efectivos para la reutilización del código y proporcionan alta calidad al software y las pruebas realizadas a la solución propuesta, permitieron que se obtuviera una aplicación que cumple con las funcionalidades identificadas.



Conclusiones Generales

Con el desarrollo de la investigación se logró cumplir el objetivo propuesto, se desarrolló la implementación del Subsistema de Manejo de Archivos del Sistema para el Manejo Integral de la perforación de Pozos de Petróleo y Gas. A continuación se mencionan los resultados obtenidos a lo largo de la investigación.

El estudio realizado a nivel nacional e internacional del estado del arte, permitió conocer acerca de las diferentes aplicaciones existentes para la exportación e importación de registros de pozos. La correcta selección y utilización de las herramientas y tecnologías, permitió que el resultado obtenido cumpliera con los requisitos presentados por el cliente. El diseño del modelo de implementación, permitió describir, organizar y estructurar los elementos Citados en el entorno de la programación y las pruebas realizadas a la solución propuesta constituyen una guía para la calidad del sistema, obteniéndose un sistema funcional que cumple con los requisitos del cliente.

Recomendaciones

Teniendo en cuenta el resultado obtenido en la investigación y basado en la experiencia adquirida, se proponen las siguientes recomendaciones:

- Extender las funcionalidades del sistema, basándose en la importación de otros tipos de registros de pozos y la exportación de los mismos.
- Permitir en nuevas versiones exportar registros de pozos en otros tipos de formatos.
- Realizar pruebas de caja blanca para medir la calidad y complejidad del código.

Referencias Bibliográficas

1. Cuevas, D.T., Análisis y Diseño del Sistema de Manejo Integral de Perforación de Pozos. 2009.
2. Definición de Interoperabilidad. 2005 [Citado 2011 9 de junio]; Disponible en: <http://www.mastermagazine.info/termino/5416.php>.
3. Definición de Fichero. 2007 [Citado 2011 9 de junio]; Disponible en: <http://www.definicionabc.com/general/fichero.php>.
4. Registro De Pozos. 2010 [Citado 2011 9 de junio]; Disponible en: <http://www.buenastareas.com/ensayos/Registro-De-Pozos/538940.html>.
5. Peloton well focused. [Citado 2011 9 de junio]; Disponible en: <http://www.peloton.com/es/default.asp?id=29>
6. Schlumberger. Drilling Software 2011 [Citado 2011 9 de junio]; Disponible en: <http://www.slb.com/services/software/drilling.aspx>.
7. Halliburton, Software WELLSIGHTMR v4.5. 2007.
8. Contreras, J.M., Redefinición de la Industria Petrolera Latinoamericana, in En el Entorno de la Globalización:El caso de la Industria petrolera Venezolana.
9. Info-Petroleo. [Citado 2010 7 de diciembre]; Disponible en: <http://www.info-petroleo.com/>.
10. Fernández, Y.A.P., Propuesta de Arquitectura de Software para la Empresa de Perforación y Extracción de Petróleo de Occidente. 2009, Universidad de las Ciencias Informáticas: Ciudad de la Habana. p. 13.
11. Paneque, J.R.A., Módulo Pozo del Sistema de Manejo Integral de Perforación de Pozos. 2010, Universidad de las Ciencias Informáticas.: Ciudad de la Habana.
12. Tamaris, P.R. Registros de Pozos. 2010 [Citado 2010 7 de diciembre]; Disponible en: <http://www.buenastareas.com/ensayos/Registro-De-Pozos/538940.html>.
13. González, O.M. Diseño de los Objetos. [Citado 2010 3 de febrero]; Disponible en: <http://www.monografias.com/trabajos14/disenio-sistemas/disenio-istemas.shtml#DEFIN>.
14. Capmany, M.A. Strater Potente e innovador software para el registro de pozos y trazado de perforaciones. 2010 [Citado 2010 7 de diciembre]; Disponible en: <http://www.addlink.es/productos.asp?pid=430>.

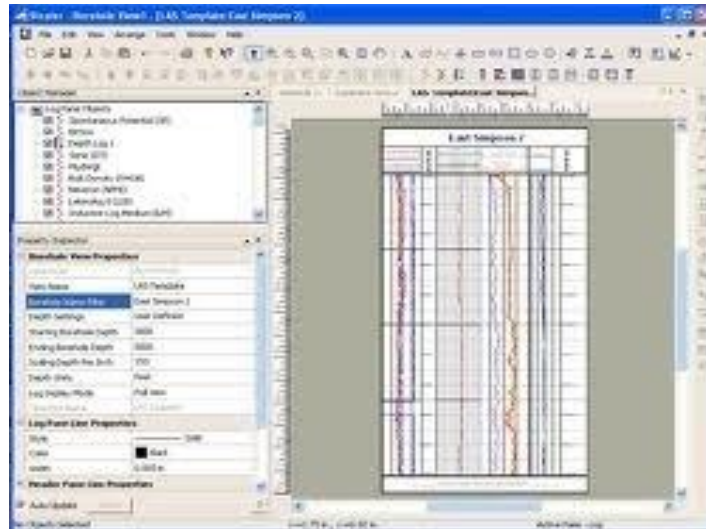
15. Programas de Elaboración de Registros geológicos [Citado 2010; Disponible en: http://www.petrokemls.com/index.php?option=com_content&task=view&id=60&Itemid=34].
16. WellFlo™ Software de Ingeniería Petrolera, in Diseño, modelado, optimización e identificación de problemas en pozos individuales de petróleo y gas, de flujo natural o levantados artificialmente. 2008.
17. Torres, M.Y.G., Subsistema de importación y exportación de registros de pozo. 2010, Universidad de las Ciencias Informáticas: Ciudad de la Habana.
18. Target, Software para la Exploración del Subsuelo.
19. Correa, O.M.R. and A.N. Sánchez, Solución Informática a Pegaduras de Tuberías en la Perforación de Pozos Petroleros. 2010, Universidad de las Ciencias Informáticas.: Ciudad de la Habana.
20. Escoto, R.P., La interoperabilidad del software y aplicaciones en redes de empresas.
21. Diana Santillan Ceja, M.V. Software Propietario y Libre. [Citado 2011 7 de junio]; Disponible en: <http://www.mitecnologico.com/Main/SoftwarePropietarioYLibre>.
22. libre, L.e. Estándra abierto. 2011 [Citado 2011 7 de junio]; Disponible en: http://es.wikipedia.org/wiki/Est%C3%A1ndar_abierto.
23. Camacho, L. Archivos. [Citado 2011 8 de junio]; Disponible en: <http://manipulaciondeficheros/arch.shtml>.
24. Lenguajes de Programación. 2009 [Citado 2011 1 de marzo]; Disponible en: <http://www.lenguajes-de-programacion.com/programacion-web.shtml>.
25. Mora, S.L. Programación de aplicaciones web Historia, principios básicos y clientes web. [Citado 2011 1 de marzo]; Disponible en: <http://www.editorial-club-universitario.es/libro.asp?ref=367>.
26. Morales, P.C. Desarrollo de Aplicaciones Distribuidas basadas en Tecnologías Web. 1999 [Citado 2011 1 de marzo]; Disponible en: <http://trevinca.ei.uvigo.es/~pcuesta/publicaciones/TecWeb.pdf>.
27. La enciclopedia libre. 5 de marzo del 2011 [Citado; Disponible en: <http://es.wikipedia.org/wiki/Framework>].
28. Gutiérrez, J.J. ¿Qué es un framework web? [Citado 2011 5 de marzo]; Disponible en: http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
29. Fabien Potencier, F.Z., Symfony la guía definitiva.

30. Development HowToContributeToSymfony. 1998 [Citado 2011 1 de marzo]; Disponible en: <http://trac.symfony-project.org/wiki/HowToContributeToSymfony#CodingStandards>.
31. Introducción a la Programación Orientada a Objeto. [Citado 2011 5 de marzo]; Disponible en: http://zarza.usal.es/~fgarcia/doc/tuto2/I_1.htm.
32. Izquierdo, L.R., Introducción a la Programación Orientada a Objetos.
33. Programación orientada a objetos. [Citado 2011 5 de marzo]; Disponible en: http://java.ciberaula.com/articulo/tecnologia_orientada_objetos/.
34. Cuberaula. Introducción, definición y evolución de PHP. [Citado 2011 6 de marzo]; Disponible en: http://php.ciberaula.com/articulo/introduccion_php/.
35. Hinostroza, R.R. Características de PHP. 2005 [Citado 2011 6 de marzo]; Disponible en: <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>.
36. Alvarez, M.A. Editor web orientado a la programación de páginas PHP, con ayudas en la gestión de proyectos y depuración de código. 2003 [Citado 2011 6 de marzo]; Disponible en: <http://www.desarrolloweb.com/articulos/1178.php>.
37. libre, L.e. Zend Studio. 6 de marzo del 2011 [Citado; Disponible en: http://es.wikipedia.org/wiki/Zend_Studio.
38. García, R.M.M., Sistemas de Bases de Datos. septiembre del 2005, Villa Clara. 165.
39. PostgreSQL Capacitación Nivel 1. 22 de febrero del 2011 [Citado; Disponible en: http://www.postgresql.org.pe/articles/introduccion_a_postgresql.pdf.
40. Qué es un servidor web ? 2005-2011 [Citado 2011 21 de marzo]; Disponible en: <http://www.misrespuestas.com/que-es-un-servidor-web.html>.
41. Una Introducción a APACHE. 2010 [Citado 2011 21 de marzo]; Disponible en: http://linux.ciberaula.com/articulo/linux_apache_intro/.
42. Fontela, C. Técnicas de aseguramiento de la calidad del producto. 2009 [Citado 2011 20 de mayo]; Disponible en: <http://cysingsoft.wordpress.com/2008/06/24/tecnicas-de-aseguramiento-de-la-calidad-del-producto/>.
43. Procesos de la ingeniería de requerimientos. [Citado 2011 26 de marzo]; Disponible en: http://www.google.com/url?sa=t&source=web&cd=4&ved=0CC0QFjAD&url=http%3A%2F%2Ffantares.itmorelia.edu.mx%2F~jcolivar%2Fcourses%2Fpm10a%2Fpm_u1.doc&rct=j&q=definicion%20de%20requerimiento%20por%20la%20IEEE&ei=ajiOTdaDH-GH0QG1xZSrCw&usg=AFQjCNFW4xWEv7DfOWvMZVRikce9KvvS_Q&cad=rja.

44. Johany. Ingeniería de requerimientos. Ingeniería de Software. [Citado 2011 26 de marzo]; Disponible en: <http://www.monografias.com/trabajos6/resof/resof.shtml>.
45. Especificación de Requisitos. 27 de marzo [Citado; Disponible en: http://portal.cedin.prod.uci.cu/tab/system_show.
46. Acuña, K.B. Diseño e implementación del sistema. [Citado 2011 26 de marzo]; Disponible en: <http://www.eumed.net/libros/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.
47. Lasso, A. Arquitectura de Software. 2004 [Citado 2011 26 de marzo]; Disponible en: http://eisc.univalle.edu.co/materias/Material_Desarrollo_Software/Arquitectura%20de%20Software.htm.
48. PHPExcel – Lee y escribe Excel's con PHP. 2009 [Citado 2011 27 de marzo]; Disponible en: <http://fuubar.wordpress.com/2009/07/27/phpexcel-lee-y-escribe-excels-con-php/>.
49. Scribd. Patrones Arquitectónicos 2011 [Citado 2011 7 mayo]; Disponible en: <http://es.scribd.com/doc/34279909/Puntos-Examen-Patrones-Arquitectonicos-Secc01>.
50. ¿Qué son los patrones de diseño?. El patrón Observador 2005 [Citado 2011 7 mayo]; Disponible en: <http://chuidiang.blogspot.com/2005/12/qu-son-los-patrones-de-diseo-el-patrn.html>.
51. Gutierrez, J.A.S. Patrones GRASP (Patrones de Software para la asignación General de Responsabilidad). Parte II. 2007 [Citado 2011 7 mayo]; Disponible en: <http://jorgesaavedra.wordpress.com/category/patrones-grasp/>.
52. Ivar Jacobson, G.B., James Rumbaugh, El proceso unificado de desarrollo de software. 2000.
53. Pressman, Un enfoque Práctico. 2002.
54. Sobre la disciplina de prueba. 2011 11 de mayo 2011 [Citado; Disponible en: http://eva.uci.cu/file.php/259/Curso_20102011/Semana_9/Conferencia_7/Materiales_Basicos/Sobre_la_disciplina_de_Prueba.pdf.

Anexos

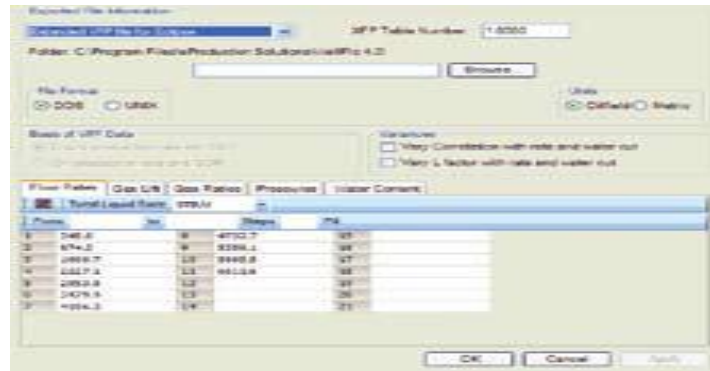
Anexo 1: STRATER



Anexo 2: MASTER LOG



Anexo 3: Wellflo



Anexo 4: Análisis de registros de pozos petroleros

Subsistema de Importación y Exportación de Registros de Pozos

Archivo Funcionalidades

Versión Información Parámetros **Curvas** Matriz de Datos

Nemotécnico	Unidades de Medida	API CODE	Descripción
DEPT	M		DEPTH (BOREHOLE) {F10.1}
DEVI	DEG		Hole Deviation {F13.4}
HAZI	DEG		Hole Azimuth {F13.4}
CGR	GAPI		Computed Gamma Ray {F13.2}
POTA			Potassium {F13.4}
SGR	GAPI		Gamma Ray {F13.2}
THOR	PPM		Thorium {F13.4}
URAN	PPM		Uranium {F13.4}
HTEM	DEGC		HTC Temperature {F13.2}
GR	GAPI		Gamma-Ray {F13.2}
NPHI	V/V		Thermal Neutron Porosity (Ratio Method) {F13.3}
CFTC	HZ		Corrected Far Thermal Counting Rate {F13.2}
CNTC	HZ		Corrected Near Thermal Counting Rate {F13.2}
SP	MV		SP Shifted {F13.2}
C1	MM		Caliper 1 {F13.2}
C2	MM		Caliper 2 {F13.2}
DI_HRLT	MM		HRLT Computed Diameter of Invasion {F13.2}
RT_HRLT	OHMM		HRLT Computed True Resistivity {F13.2}
RXO_HRLT	OHMM		HRLT Computed Invaded Zone Resistivity {F13.2}
RM_HRLT	OHMM		HRLT Mud Resistivity {F13.2}
RLA0	OHMM		HRLT Borehole Corrected Resistivity 0 {F13.2}
RLA1	OHMM		HRLT Borehole Corrected Resistivity 1 {F13.2}
RLA2	OHMM		HRLT Borehole Corrected Resistivity 2 {F13.2}
RLA3	OHMM		HRLT Borehole Corrected Resistivity 3 {F13.2}
RLA4	OHMM		HRLT Borehole Corrected Resistivity 4 {F13.2}
RLA5	OHMM		HRLT Borehole Corrected Resistivity 5 {F13.2}
HCAL	MM		HRCC Cal. Caliper {F13.2}
HDRA	G/C3		HRDD Density Correction {F13.3}
PEFZ			HRDD Standard Resolution Formation Photoelectric f
PHOZ	G/C3		HRDD Standard Resolution Formation Density {F13.3}

Anexo 5: Reporte Diario de Perforación

Reportes Perf Diarios - Via Blanca Norte - 101.xls [Modo de compatibilidad] - Microsoft Excel												
Inicio Insertar Diseño de página Fórmulas Datos Revisar Vista												
1	Nombre del Pozo		Via Blanca Norte -101		Equipo	GW-139		Day:	3	Date:	13/05/2009	
2	Prof:	334	m	Avance:	269	m	Op. @:	6-90	Perforando @ 385m			
3	Costos M. Total:		Presup:	\$8.430.000	Día:	\$34.722	Acum:	\$155.299	%	2%	Combustible (Ltr)	
4	Costos CUC:		Presup:	\$6.026.100	Día:	\$20.660	Acum:	\$66.442	%	1%	Día	5.880
5	Costo del Lado:		Presup:	\$480.000	Día:	\$2.903	Acum:	\$12.667	%	3%	Acum	13.279
6	Zapato:		m	Jar de Perforación	Motor N°S:		Inv	83.440				
7	Próximo :	13 38"	550	m	N°S	Schlumberger	Hrs. Acum:	37,75	Horas:	Cap	46.580	
8	Recibido por:		Alexis				De: Humberto / Jimmy					
9	Hora	Actividad					Hora	Actividad				
10	00:00	Se perfora de 125m @ 330m/d										
11		- 2:00 hrs en conexiones.										
12		- 4:00 hrs Trabajando la hta. Circulando plderas limpiaza										
13	24:00											
14												
15												
16												
17												
18												
19												
20												
21												
<p>Próximamente 24h. Avanzar en la perforación con posible cambio de bitumen</p> <p>Distribución del Tiempo: - Presupuesto: - Semana 1 - Día 3 - Día 3 -</p>												

Anexo 6: Reporte de geología

Nombre del pozo, Reporte Diario de Geología, Fecha(s) [Solo lectura] [Modo de compatibilidad] - Microsoft Excel											
Inicio Insertar Diseño de página Fórmulas Datos Revisar Vista											
Siglas del equipo											
Nombre completo del pozo					REPORTE DIARIO DE GEOLOGÍA						
Siglas oficiales (UWI)											
Siglas del equipo											
Fecha (Ver Comentario)					Hora (VC)						
DATOS GENERALES											
Rep. #	Reportado a: Ravi Rodriguez Mansueta								SECC:		
OLA #	Reportado por:								KEI		
SP#	Supervisor de perforación:								CL		
Procesamiento (Muestra Test)	Centro areminal (JM)								REP DE		
FB: /	Formación en el fondo:								Operación y la hora de el reporte:		
FB: /	Chal:										
INFORMACION GEOLOGICA											
Topes y bases de formaciones y marcadores:											
Fac/Marcador	Prof por muestra	FMD	E/E	Profesor in EP	Comida	Fac/Marcador	Prof por muestra	FMD	E/E	Profesor in EP	Comida
Ma Blanca	100	90									
Anaco	830										
Última muestra descrita a la profundidad:											
Entrada		Entrada		Entrada		Fluj		Bitux:		Bitux:	
Última cambio: Zapato a fluj											
Resumen de la litología en las últimas 24 hrs:											
Descripción abreviada y sus %											

Anexo 7: Parte Operativo de Perforación Oficial

Parte Operativo de Perforación		Fecha: 24/3/09
Parte Operativo de Perforación Fecha: 24/3/09		
Pozo	Bacuranao 100	
Operador	CUPET	
Equipo	GW-122	
Trunking / Teléfono	275 fig / 052804107	
Profundidad al cierre, m	2687	
Diámetro barrena, pulg	8 1/2	
Avance, m	25	
Tiempo perforando, h	7,75	
TVD, m	1446,65	
Desplazamiento, m	1727,88	
Ángulo / Azimut	76,58 / 315,40	
Den. del lodo (gr / cm3)	1,43	
Vol. lodo tanques, m3	70	
Litología a 2680mMD.	Claystone -15% Serpentinitas-30%	
	Areniscas polimicticas- 40%	
	Pedemal -10%, Calizas-5%	
Lectura de gas y profundidad, % / m	0,56% de 2670-2687m.	

Anexo 8: Reporte de Inclinometría

Plan #1 HBE-102 (Proposal)

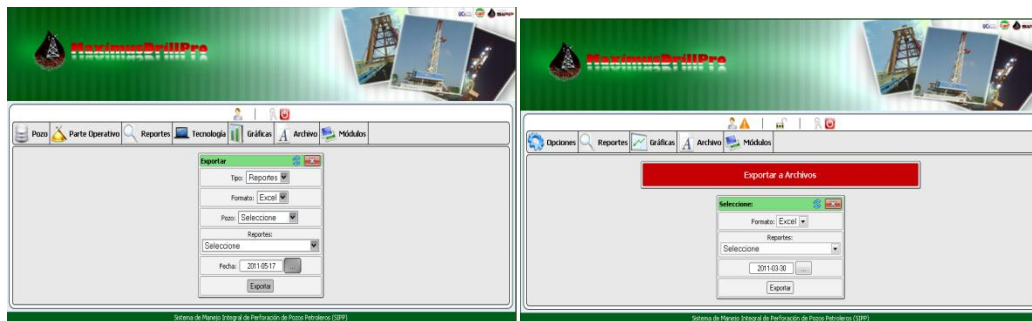
Company : Cupet Date : 3/22/2010
 Well : VMD101 File : VMD101.P1
 Location : Vazadero Oeste Structure, Vazadero Field
 Reference : TR028078

MD Elevation : 10.9 TWI : Gr Elevation : 1.70
 Slope :
 Comment : corrected MD to 10.9 ft / 3.28 m

Vertical Section Calculated Along Azimuth 353.95°

MD	Inc	Asi	TVD	SubSea	Lat	Dep	V'Sect	D'lag	Closure
m	deg	m	m	m	m	m	m	m	m
303.00	0.00	0.00	303.00	-292.10	0.00	0.00	0.00	0.00	0.00
370.59	0.30	120.95	370.59	-292.49	-0.09	0.16	-0.11	0.13	0.18
430.74	0.24	146.62	430.74	-293.54	-0.27	0.35	-0.31	0.07	0.45
449.07	0.56	20.08	449.07	-293.97	-0.22	0.41	-0.26	1.14	0.46
459.50	1.67	6.75	459.50	-294.60	-0.03	0.44	-0.05	3.53	0.44
469.12	3.15	356.90	469.11	-295.21	0.37	0.45	0.32	4.72	0.58
478.42	4.41	357.80	478.59	-295.89	0.99	0.43	0.94	3.99	1.08
486.06	5.56	355.97	486.01	-297.11	1.61	0.37	1.76	3.79	1.85
497.58	6.97	354.97	497.42	-298.52	2.84	0.27	2.80	4.48	2.86
507.03	7.53	351.92	506.83	-299.83	4.03	0.13	3.99	5.19	4.03
516.46	8.24	350.54	516.07	-301.47	5.24	-0.05	5.21	5.23	5.34
526.14	8.83	349.61	525.75	-303.85	6.70	-0.26	6.69	1.63	6.71
535.60	9.34	350.37	535.10	-304.20	8.12	-0.51	8.13	1.95	8.14
545.08	9.75	349.81	544.45	-303.85	9.62	-0.79	9.66	2.69	9.67
554.74	9.99	344.95	553.97	-303.07	11.25	-1.14	11.31	1.24	11.31
564.37	10.42	347.91	563.44	-302.84	12.92	-1.61	13.02	2.03	13.02
572.89	11.13	347.04	572.88	-304.99	14.70	-1.80	14.62	3.47	14.83

Anexo 9: Interfaz de Exportar



Anexo 10: Interfaz de Importar

