

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 5



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



**Subsistema Pozo del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo
Gas. Versión 2.0**

Autor: Camilo Nelson López Zaldívar.

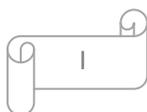
Tutor: Ing. David Tavares Cuevas.

Ciudad de la Habana, junio 2011.

Dedicatoria

Le dedico este trabajo:

- A mi mamá y mi papá por siempre estar presentes dándome su cariño, comprensión y apoyo, por todos los consejos que me han dado para ser una mejor persona y por ser el mayor regalo que Dios me ha dado, que es tener unos padres como ustedes.*
- A mis abuelos por parte de madre por siempre confiar en que cada día puedo ser una mejor persona y por siempre apoyarme incondicionalmente.*
- A la memoria de mi abuela Margó que se que estaría muy orgullosa de mí y a la memoria de mi abuelo José Ramón que aunque la vida no me dio la oportunidad de conocerlo sé que me hubiera dado todo su apoyo para ser lo que soy hoy.*
- A mi tía Amarilis y a mis tíos Kiko y Tambuly que siempre me han ayudado cuando lo he necesitado, por quererme tanto y confiar en mí.*
- A todos mis tíos y tías por todo el ánimo que me han dado todo este tiempo.*
- A todos mis primos que son las personas con las que me he criado y por siempre brindarme su apoyo durante toda mi carrera.*

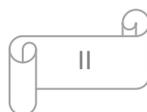


Agradecimientos

Le agradezco:

- *A mis padres por ser mi mayor motor impulsor y guía en todo momento, por todo el apoyo que me han dado toda la vida, por tanta dedicación, sacrificio y amor.*
- *A mis abuelos por parte de madre y a la memoria de mis abuelos por parte de padre que se que estarían muy orgullosos de verme graduado de Ingeniero.*
- *A mis tíos por siempre confiar en mí y desear tanto como yo que la familia tuviera su primer Ingeniero. En especial a mi tía Amarilis y mis tíos Kiko y Jambuly que siempre me han tendido la mano cuando lo he necesitado.*
- *A Mailin por ser la persona con la que he compartido todos estos años en los momentos buenos y malos.*
- *A mis compañeros de trabajo del Team SUPP (Roseli, Aniuvis, Dvd, Hti, Jordan y Reynel) que siempre estuvieron allí para aclarar cualquier duda.*
- *A todos mis colegas de fiestas, locuras y Gym (Los Pintum Boys).*
- *A mi tutor por toda la dedicación y apoyo que me brindó para lograr este trabajo, gracias por ser un amigo más que un tutor.*
- *A todos los que de una forma u otra compartieron conmigo estos cinco años.*

A todos muchas gracias.



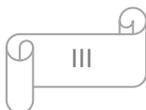
Declaración de Auditoría

Declaro que soy el único autor de este trabajo y 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 2011.

Tutor: David Tavares Cuevas

Autor: Camilo Nelson López Zaldivar



Resumen

RESUMEN

El presente trabajo de diploma aborda la problemática asociada a la gestión de la información resultante del proceso de perforación de pozos de petróleo en tierra. El objetivo de esta investigación consiste en plantear una solución que permita resolver la problemática en cuestión. Para ello se propone implementar la segunda versión del Subsistema Pozo del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas, el cual permita la gestión y el control de la información manipulada en el proceso de perforación de los pozos de petróleo en tierra.

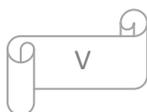
Para dar cumplimiento de los objetivos planteados se abordan los conceptos asociados al problema, se realiza un análisis de los sistemas que afrontan esta problemática a nivel internacional así como de la primera versión del Subsistema Pozo. Se describen las herramientas y tecnologías utilizadas en el desarrollo de la solución. Se exponen las principales características funcionales de la aplicación, mostrándose los componentes que componen el subsistema a través del modelo de implementación.

Palabras clave: Control, gestión, información, perforación.

Índice de Contenido

ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO 1: Fundamentación Teórica.	5
1.1 Introducción	5
1.2 Conceptos asociados al dominio del problema.....	5
1.3 Proceso de perforación de pozos de petróleo en tierra.....	8
1.3.1 Descripción general.....	8
1.3.2 Descripción actual del dominio del problema.....	9
1.4 Análisis de otras soluciones existentes.....	10
1.4.1 SIPP.....	10
1.4.2 MasterView.....	12
1.4.3 InfoPerf.....	15
1.4.4 WellSight v4.5.....	16
1.5 Enfoques de la Programación.....	17
1.5.1 Programación Orientada a Objetos(POO).....	17
1.5.2 Estilos y estándares de codificación.....	18
1.5.3 Framework de desarrollo.....	18
1.5.4 Subsistema Web.....	19
1.6 Conclusiones Parciales.....	20
CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.	21
2.1 Introducción.....	21
2.2 Lenguaje de Programación.....	21
2.2.1 PHP 5.....	21
2.2.2 HTML.....	22
2.2.3 CSS.....	22
2.2.4 JavaScript.....	22
2.3 Entorno Integrado de Desarrollo (IDE).....	24
2.3.1 Zend Studio.....	24
2.4 Framework de Desarrollo.....	24
2.4.1 Symfony.....	24
2.5 Estilos y Estándares de Codificación.....	25
2.6 Sistema Gestor de Base de Datos.....	28
2.6.1 PostgreSQL 8.4.....	28
2.7 Servidor Web.....	29



Índice de Contenido

2.7.1	<i>Apache</i>	29
2.8	<i>Visual Paradigm</i>	30
2.9	<i>Técnicas de Aseguramiento de la Calidad</i>	30
2.10	<i>Conclusiones parciales</i>	31
CAPÍTULO 3: Presentación de la solución y validación		32
3.1	<i>Introducción</i>	32
3.2	<i>Características Subsistema Pozo</i>	32
3.2.1	<i>Actores que interactúan con el subsistema</i>	32
2.2.2	<i>Requisitos Funcionales del subsistema</i>	32
3.2.3	<i>Diagrama de Casos de uso del Sistema</i>	33
3.2.4	<i>Subsistema Pozo</i>	35
3.3	<i>Modelo de implementación</i>	43
3.3.1	<i>Vista de implementación</i>	43
3.3.2	<i>Diagrama de componentes</i>	45
3.4	<i>Elementos arquitectónicos</i>	54
3.4.1	<i>Patrón Arquitectónico</i>	54
3.4.2	<i>Patrones de Diseño</i>	54
3.5	<i>Pruebas del Subsistema Pozo</i>	55
3.5.1	<i>Pruebas de Caja negra</i>	56
3.5.2	<i>Pruebas de Aceptación</i>	56
3.5.3	<i>Descripción de los Casos de Pruebas</i>	57
3.6	<i>Conclusiones Parciales</i>	61
CONCLUSIONES GENERALES		63
RECOMENDACIONES		64
REFERENCIAS BIBLIOGRÁFICAS		65
ANEXOS		67
GLOSARIO DE TERMINOS		70

Índice de Tablas y Figuras

ÍNDICE DE TABLAS

<i>Tabla 1: Relación entre las principales clases y los patrones de diseño.</i>	55
<i>Tabla 2: CP Gestionar Transferencia de Combustible. Crear Transferencia (31).</i>	58
<i>Tabla 3: CP para sección Crear Descripción Litológica (31).</i>	58

ÍNDICE DE FIGURAS

<i>Figura 1: Vista de Despliegue.</i>	11
<i>Figura 2: Diagrama de Caso de Uso del Subsistema Pozo.</i>	34
<i>Figura 3: CU Gestionar Costos Iniciales. Interfaz crear.</i>	37
<i>Figura 4: CU Gestionar Intervalo de Perforación. Interfaz cambiar.</i>	37
<i>Figura 5: CU Gestionar Composición de Herramienta. Interfaz crear.</i>	38
<i>Figura 6: CU Cerrar Día. Interfaz cerrar.</i>	39
<i>Figura 7: CU Gestionar Intervalo de Perforación. Interfaz terminar perforación.</i>	40
<i>Figura 8: Generar Reporte Diario de Perforación. Interfaz mostrar reporte.</i>	41
<i>Figura 9: CU Gestionar Descripción Litológica. Interfaz crear.</i>	42
<i>Figura 10: CU Generar Reporte Diario de Geología. Interfaz mostrar reporte.</i>	43
<i>Figura 11: Diagrama Vista de Implementación.</i>	44
<i>Figura 12: Diagrama de Componentes Gestionar Intervalo.</i>	47
<i>Figura 13: Diagrama de Componentes Gestionar Transferencia de Combustible.</i>	48
<i>Figura 14: Diagrama de Componentes Gestionar Composición de Herramientas.</i>	49
<i>Figura 15: Diagrama de Componentes Gestionar Costos Diarios.</i>	50
<i>Figura 16: Diagrama de Componentes Generar Reporte Diario de Perforación.</i>	51
<i>Figura 17: Diagrama de Componentes Gestionar Descripción Litológica.</i>	52
<i>Figura 18: Diagrama de Componentes Generar Reporte Diario de Geología.</i>	53
<i>Figura 19: Patrón Arquitectónico MVC.</i>	54

Introducción

INTRODUCCIÓN

El petróleo es un recurso natural no renovable que aporta el mayor porcentaje del total de la energía que se consume en el mundo, siendo la fuente de energía más importante de la sociedad actual. Pensar en qué pasaría si se extinguiera repentinamente hace llegar a la conclusión de que se trataría de una verdadera catástrofe: los aviones, los automóviles, autobuses, gran parte de los ferrocarriles, los barcos y centrales térmicas dejarían de funcionar. Hoy más de 80% de la energía que se produce en el mundo es sobre combustible fósiles (petróleo, gas o carbón mineral). Los avances de la ciencia y la técnica, aun no logran disminuir sus costos y aumentar su eficiencia de los métodos de generación de energía usando fuentes renovables.

Actualmente, el agotamiento de las reservas de petróleo constituye un grave problema, pues al ritmo actual de consumo las reservas mundiales conocidas se agotarían en menos de 40 años. Los países dependientes de los combustibles fósiles (en su mayoría subdesarrollados) buscan nuevas formas de energía de menos costo y renovable como la energía solar, eólica, hidroeléctrica, mientras que los países productores de petróleo presionan para que se siga utilizando el petróleo. Según las estimaciones de reservas, a medio plazo, la situación no parece tan alarmante, pues se prevé que con el desarrollo de nuevas tecnologías para la exploración, perforación y extracción del crudo, posibiliten identificar nuevos campos productores.

En la industria del petróleo se maneja gran cantidad de información, asociada a cualquiera de sus procesos. Para poder manejar la información correctamente, se desarrollan sistemas informáticos que permiten realizar un uso efectivo de la misma, contribuyendo directamente a la toma de decisiones. Un sistema de gestión puede ser definido como es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización (1). Como lo expresa el propio concepto los sistemas de gestión (como se les conoce comúnmente) contribuyen a lograr los objetivos de una organización optimizando los procesos, centrándose en el análisis lógico y el enfoque disciplinado de los mismos.

Actualmente existen empresas, las cuales se dedican al mercado de la industria del petróleo, muchas de estas entidades se especializan en la realización de software para la automatización de los procesos de la industria petrolera. Estas empresas con sus distintas entidades distribuidas en todo el mundo, intervienen en los diferentes procesos por los que debe pasar el crudo. La industria del petróleo en nuestro país, como en muchos otros, se divide en tres grandes procesos: 1- Exploración-Producción, 2- Refinación y 3- Comercialización. Las entidades especializadas en Exploración-Producción, como su nombre lo indica, 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. En Cuba la unión Cuba-Petróleo (CUPET), es la unión de empresas responsable del desarrollo y

Introducción

mantenimiento de la industria petrolera en Cuba conjuntamente con compañías extranjeras, que operan y prestan servicios (2).

En la Universidad de las Ciencias Informáticas (UCI), se desarrollan sistemas para la automatización de procesos de la Industria del Petróleo. En ella se encuentra el Centro de Desarrollo de Informática Industrial donde se desarrolla el Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas (SIPP), el cual responde a las necesidades del Centro de Investigaciones de Petróleo (CEINPET) y la Dirección de Intervención y Perforación de Pozos (DIPP), entidad responsable de todos los pozos en perforación en tierra del territorio cubano. Este proyecto posee un subsistema definido el cual agrupa las funcionalidades que van a ser utilizadas por los especialistas de la DIPP en los pozos.

Durante el proceso de desarrollo en el marco del proyecto se obtuvo una versión inicial del subsistema. Debido a la poca experiencia y bajo conocimiento del negocio, se pasaron por alto varias relaciones entre los datos que debían existir en la persistencia de los datos (estas relaciones son las que modelan la lógica del proceso a nivel de base de datos). Al introducir estos cambios fue necesario que el subsistema pozo funcionara analizando estas nuevas relaciones y por tanto cumpliera con las reglas del negocio identificadas. Ejemplo de ellas son: la dependencia entre las propiedades de fluido y las pérdidas de fluido, la relación de los equipos del sistema de control de sólidos, la elaboración de las BHA la cual depende de las existencias de barrenas, jar y motores de fondo, entre otros.

En la gestión del combustible se identifica la deficiencia de no gestionarse las transferencias de combustible. Esto afectaba al negocio alterando los cálculos de previsión de consumo, los cuales se utilizan para realizar los abastecimientos a los pozos. Otro de los procesos que no se reflejan en toda su expresión son los relacionados con la gestión de la información geológica. Se profundizó en el proceso de elaboración de las descripciones identificándose los cambios necesarios en el modelo de datos para poder registrarlas.

Durante esta revisión se identifica que no se manejaba correctamente las dependencias entre los datos en el procesos de gestión de la composición de la herramienta; la gestión de los sidetrack¹ era aún deficiente; la gestión de los intervalos de perforación no contaba con la profundidad necesaria, lo cual podía comprometer otras funcionalidades que necesitaran de estos datos para su funcionamiento. En este momento el subsistema pozo quedó con un alcance mayor, por lo que fue necesario estructurar modularmente el subsistema.

¹ Operación que se realiza cuando se da por perdido un hoyo y se decide desviar el pozo hacia otro rumbo. Esto se realiza ante complejidades técnicas o geológicas como: rotura en la herramienta o la perforación de una formación fuera de lo planificado.

Introducción

Por estas razones se hace necesaria una versión superior a la inicialmente negociada y planificada del Subsistema Pozo, con el objetivo de lograr la completitud y optimización del funcionamiento de las operaciones que se realizan en los pozos en perforación a través del sistema. Este redimensionamiento permite que el subsistema asimile a profundidad las reglas del negocio, ajustándose con mayor exactitud a las flexibilidades y restricciones necesarias, para permitir que se gestione la información y el proceso minimizando los errores humanos.

Teniendo en cuenta la **situación problemática** planteada se propone implementar la segunda versión del Subsistema Pozo el cual permita gestionar, respetando las reglas del negocio, la información generada durante la perforación de pozos de petróleo y gas. Por tanto el **problema a resolver** se presenta en el ¿cómo automatizar los procesos de gestión y control de la información consolidada durante el proceso de perforación de pozos de petróleo en tierra respetando las reglas del negocio? Como **objeto de estudio** de la investigación se propone los procesos de gestión y control de la información en la perforación de pozos de Petróleo y Gas, lo que definiéndose como **objetivo general** del trabajo Implementar el Subsistema Pozo Versión 2.0 del Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas teniendo como **campo de acción** la automatización del proceso de gestión y control de la información en el área de perforación de la Industria Petrolera.

Se **defiende la idea** de que con la implementación del Subsistema Pozo en su segunda versión, se podrán mejorar los procesos de gestión y control de la información consolidada durante el proceso de perforación de pozos de petróleo en tierra que fueron solicitados por el cliente. Además de contribuir a un mejor control del proceso, así como mejorar las prestaciones para la toma de decisiones.

El Subsistema Pozo está diseñado para adaptarse a las características del negocio actual de la perforación de pozos en cualquier lugar del mundo. Además de ello se ocupa de elementos que otros sistemas no conciben, lo cual lo dota de un grado de originalidad que puede ser determinante para su comercialización en el futuro como parte de las soluciones para esta rama. Con la implementación del subsistema pozo en su versión 2.0, con este enfoque: permitirá gestionar, controlar y supervisar el proceso en un 80%. De esta manera se dará un paso importante, para el inicio de los desarrollos para esta área en la industria petrolera. En nuestro país no hay indicios de la implantación de un sistema con este fin; con un riesgo tan alto, donde un error puede costar mucho más que tiempo y esfuerzo. Esto implica que la relevancia del resultado va más allá de si mismo y trasciende de las fronteras de su uso para convertirse a corto plazo en un sistema estratégico.

Para lograr el objetivo propuesto se definen las siguientes **tareas de la investigación**:

1. Análisis del estado del arte de los sistemas que gestionan el proceso de perforación.
2. Descripción de las herramientas y tecnologías utilizadas en la implementación de la solución.
3. Implementación del Subsistema Pozo.

Introducción

4. Elaboración del modelo de implementación del subsistema.
5. Realización de las pruebas al Subsistema Pozo.

Para la realización de estas tareas se emplearon los diferentes **métodos de la investigación**.

Métodos Empíricos

- “Observación”: Con el objetivo de obtener un nivel de realidad sobre la información consolidada dentro del proceso de perforación y el funcionamiento del centro en una situación real, lo que constituye el objeto de estudio de esta investigación.

Métodos Teóricos

- “Histórico – Lógico”: Para determinar las tendencias actuales, en este caso se empleó para realizar el estudio de las principales soluciones que den respuesta al problema científico, así como la recopilación de información de las principales empresas de esta rama, lo que nos serviría de guía para la construcción de la solución.
- “Analítico – Sintético”: Permite procesar toda la información en particiones más pequeñas y fácil de comprender, y después compactar esas partes que fueron analizadas formándolas más simples, lo que posibilita la comprensión de la misma, con ideas claras y concisas.

Estructura del contenido:

Para un mejor entendimiento del trabajo se decide estructurar el mismo de la siguiente manera:

- **Capítulo 1: Fundamentación Teórica.** Este capítulo aborda todo lo relacionado con la fundamentación teórica. Se abordan los conceptos asociados al problema, al mismo tiempo se describe el objeto de estudio y la situación problemática; además se presenta la descripción del estado de arte con las soluciones que existen alrededor del objeto de estudio.
- **Capítulo 2: Tendencias y tecnologías actuales a utilizar.** Se abordan las tecnologías utilizadas (herramientas y lenguaje de programación). Además se exponen los estilos y estándares de codificación a utilizar.
- **Capítulo 3: Presentación de la solución y validación.** En este capítulo se le da un mayor enfoque a la solución propuesta, mostrando una serie de diagramas que permitirán un mejor entendimiento de la solución, así como los requisitos funcionales y descripción de las principales funcionalidades de la misma. También se evidencian las pruebas realizadas al subsistema que certifican la calidad de la solución propuesta.

CAPÍTULO 1: Fundamentación Teórica.

1.1 Introducción.

El presente capítulo aborda el objeto de estudio y la situación problemática con mayor amplitud. Se profundiza en la descripción del proceso de perforación, adentrándose más en los que estarán incluidos en la solución. Se realiza un análisis del estado del arte de sistemas a nivel mundial que se encargan de la gestión de la información generada durante el proceso de perforación de pozos de petróleo. Se abordan además los conceptos científico-técnicos que se deben conocer para poder comprender el resultado.

1.2 Conceptos asociados al dominio del problema.

El dominio del problema al que se le da solución en este trabajo es complejo y se requiere de mucha experiencia para poder conceptualizar los términos y conceptos que se manejan. En este epígrafe se exponen los conceptos más importantes para comprender el dominio del problema.

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.

Petróleo: El petróleo es una mezcla heterogénea de compuestos orgánicos, principalmente hidrocarburos insolubles en agua, que se extrae de lechos geológicos continentales o marítimos. También es conocido como petróleo crudo o simplemente crudo. Es un líquido aceitoso, inflamable, cuyo color varía de incoloro a negro.

Pozo de Petróleo y Gas: Entidad donde se realiza la perforación, la cual agrupa un gran número de entidades y trabajadores, como son las compañías de servicio de lodo, direccionales, contratistas, logísticas, entre otras.

Industria del Petróleo: La industria petrolera incluye procesos globales de exploración, extracción, refino, transporte (frecuentemente a través de buques petroleros y oleoductos) y mercadotecnia de productos del petróleo.

Procesos en la Industria del Petróleo: La industria del petróleo se divide en 5 procesos fundamentales: 1- Exploración, 2- Perforación, 3-Producción, 4- Refinación y 5- Comercializan. Estos cinco procesos sustentan todos los procesos que se realizan en la industria que tiene como resultado final los derivados del hidrocarburo. Existen varios métodos para realizar la exploración, actualmente la más usada es la exploración magnética que se basa en el envío y recepción de ondas a través de la superficie terrestre. La perforación más usada en Cuba es la direccional que se trata de perforar el

CAPÍTULO 1: Fundamentación Teórica.

caño hasta aproximadamente 200 metros y darle un ángulo para buscar el reservorio horizontalmente, se usa de tierra al mar. La producción fundamental es la surgente o por bombas de agua, la surgente es en la que el petróleo sale sin ayuda, la otra se realiza inyectando agua en el reservorio y como el petróleo es menos denso sale a la superficie. La refinación se realiza en plantas como la Níco López, se trata de separar el petróleo de otros componentes como el azufre, minerales y gases. La comercialización la realiza CUPET en sus puntos de venta de gasolina y diesel distribuidos en todo el país (3).

Unión CUPET: Es la empresa que se encarga de toda la actividad petrolera desde la exploración hasta la refinación, así como de satisfacer las necesidades del mercado nacional de hidrocarburos, a partir del incremento de la extracción de crudos nacionales.

Exploración-Producción: Área de Dirección de CUPET, la cual engloba tres procesos dentro de la industria del petróleo: la exploración, la perforación y la producción.

Perforación: Proceso más importante dentro de la Industria del Petróleo, ya que proporciona la certeza de la existencia del hidrocarburo, aunque existe una gran interdependencia entre los procesos, ya que del éxito de uno de ellos, depende el éxito del siguiente, la importancia de este proceso de debe principalmente a que aproximadamente 1 de cada 7 pozos en perforación logra encontrar y producir petróleo. En la actualidad el proceso de perforación se utilizan técnicas y tecnologías de alta complejidad como son: la perforación utilizando fluido de perforación, perforación direccional (horizontal), el encamisado y cementación de pozo, la utilización de motores, el registro a los pozos, por solo citar algunos. Muchas estas tecnologías son provistas por compañías de servicio.

Empresa de Producción y Extracción de Petróleo del Centro: entidad con la finalidad principal de regular la explotación de los yacimientos del centro del país, debido a que la mayoría de los pozos de petróleo de encuentran en la zona del centro es la empresa más grande de este tipo en el país a la cual se le subordina la dirección de intervención y perforación de pozos de petróleo.

Dirección de Intervención y Perforación de Pozos: Única entidad en Cuba responsable de, como su nombre lo indica, dirigir el proceso de perforación e intervención de todos los pozos de petróleo en nuestro país que se encuentran en ese estado, recibe diariamente los partes de los mismos sobre el avance de la perforación.

Compañías de Servicio: Son empresas que se especializan en brindar uno o varios servicios indispensables para poder lograr el objetivo de la perforación. Estas empresas brindas servicios de direccional, lodo, registros, sistemas de monitoreo en tiempo real, perforación; esto se explica dado que en la actualidad a los dueños de los campos le es factible contratar estos servicios, que ellos

CAPÍTULO 1: Fundamentación Teórica.

mismo ejecutarlos todos. Estas compañías son muchos más eficientes y sus integrantes son especialistas en la rama en que se desempeñan.

Caño de Perforación: Los pozos se perforan por tramos, es decir, no se llega a la profundidad final de un solo golpe se planifican intervalos de perforación y cada cual tiene sus características. El avance también se programa por caños (tramos), ya que no se avanza hasta que no se encamisa y cementa un tramo del pozo.

Perforación Direccional (Horizontal): Es una técnica sumamente utilizada en la actualidad. Esto se debe que la mayoría de los reservorios en la actualidad se encuentran en el mar o en lugares donde no se puede emplazar la plataforma, por lo tanto es imposible perforar de la forma convencional (vertical). Se llama dirección ya que mediante cálculos matemáticos a la herramienta de perforación se van dando pequeños ángulos a medida de que avanza la perforación, lo cual hace que se vaya haciendo una curva (parábola en términos matemáticos) logrando así avanzar en profundidad y en desplazamiento.

Registro de Pozo: El registro de pozo es un técnica moderna que consiste en llevar una herramienta hacia diferentes lugares del pozo que se está perforando, para recopilar datos geológicos que corroboren las lecturas que se hacen, con el sistema de monitoreo en tiempo real y el geólogo. Este proceso es de suma importancia ya que se puede pensar que el pozo se encuentra en un estado o formación y este proceso es el lo corrobora.

Intervalo de Perforación: Fases planificadas en un pozo a perforar, las cuales deben ser cumplidas y se encuentran recogidas en el proyecto del pozo. Estos intervalos/etapas/fases tienen características únicas como: diámetro, tipo de formación, tipo de camisa, profundidad por el largo del instrumento, profundidad por la vertical, desplazamiento, ángulo, entre otros.

Fluido de Perforación (Lodo de Perforación): Es una sustancia comúnmente llamada Lodo el cual es indispensable para el éxito del proceso. Es un compuesto químico fabricado a pie de pozo por lo químicos el cual posibilita: estabilidad a las formaciones, regula el peso sobre la barrena, ayuda a bajar las temperaturas en la herramienta, limpia el agujero (**caño de perforación**) sacando los cortes a la superficie, lo cual hace que el fondo del pozo no se eliminen muchos residuos de rocas que quedan atascadas en el caño de perforación.

Herramientas: Como su nombre lo indica son las herramientas que se colocan en la Composición de Herramientas para realizar la perforación, ejemplo: barrenas, conexiones, estabilizadores, jar, motores de fondo, power drive, bit sub, entre otros. También es usado para referirse a. Composición de Herramienta + Tubería de Perforación.

CAPÍTULO 1: Fundamentación Teórica.

Barrenas: Son las herramientas encargadas de taladra el hoyo durante la perforación. Normalmente en pozos dirigidos, trabajan en conjunto con un motor de fondo (3).

Jar: Herramienta que se utiliza para en caso de atascamiento y/o estrechamiento en el caño de perforación poder despegar la tubería y destrabar la herramienta. El Jar es especie de un martillo que emite un golpe a la formación y al esta removerse se libera la herramienta y la tubería.

Motor de Fondo: Herramienta que trabaja en conjunto con la barrena, con el objetivo de contribuir a aumentar las revoluciones por minuto de la barrena contrarrestando el torque que ejerce la formación contra el movimiento de la barrena, utilizando el fluido de perforación como elemento que hace que el motor rote sobre su propio eje, haciendo circular el fluido por dentro del motor. También se utiliza para dar el desvío necesario (construcción de ángulo) en pozo dirigidos.

Composición de la Herramientas (BHA): Es la unión de herramientas que se construye para perforar un tramo determinado. Como régimen de seguridad siempre debe llevar un jar y si el pozo se está desviando un motor de fondo.

Torque y Arrastre: Es un proceso que ocurre cuando se maniobra con la herramienta, y puede ser perforando, deslizando, levantando con y sin rotación, bajando con y sin rotación y realizando una operación llama backreaming.

Camisa de Revestimiento: Son tuberías huecas que se van colocando por caños a medida que avanza el pozo. Este proceso es de los más antiguos pero no deja de ser importante, además de la manera en que hace en la actualidad es todo utilizando herramientas guiadas por los especialistas. Estas tuberías aseguran el tramo perforando permitiendo que no colapse el pozo.

Cementación de Pozo: Es el proceso que sigue luego de encamisar el caño, esto se hace para reforzar las paredes del pozo, entre la pared y la camisa. Esto le proporciona más seguridad a la camisa, además de evitar que la camisa no quede estabilizada en el caño.

Reportes: Informes que genera el sistema SIPP a través del registro de la información resultado de los proceso y actividades diarias que se realiza en el pozo (3).

1.3 Proceso de perforación de pozos de petróleo en tierra.

1.3.1 Descripción general.

El objeto de estudio lo constituyen los principales procesos de gestión y control de las actividades realizadas en los pozos de petróleo en perforación. Actualmente no se cuenta con un sistema capaz de gestionar y controlar las informaciones que se generan en los diferentes procesos de negocio. Estos procesos son realizados por Compañías de Servicio, las cuales pueden ser cubanas y extrajeras. Cada una de estas compañías emite diariamente un reporte de sus servicios. Los Supervisores son los

CAPÍTULO 1: Fundamentación Teórica.

encargados de dirigir el proceso de perforación, además de construir diariamente los reportes de los procesos realizados, integrando todos los reportes entregados por las compañías de servicio.

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 se encuentra radicada 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 la responsable de los pozos desde su nacimiento hasta su terminación.

La DIPP es la encargada de contratar las Compañías de Servicio que van a trabajar en el pozo. Se contratan compañías que brindan servicios de Direccionales, Lodo, Perforación (Contratistas), Registro, Geología. En Cuba se hace uso de un sistema llamado WellWizard, el cual se utiliza para monitoreo en tiempo real de la perforación; este servicio lo presta la compañía mixta CUBALOB. Esta característica hace que en los pozos en perforación exista un ambiente multi-roles donde cada compañía se especializa en una parte del proceso. Los supervisores de pozos son los responsables de todo lo que ocurre en el pozo y por ende los encargados de hacer el cierre de información diaria. A las 12 AM se reciben los partes diario de las compañías y esto se recoge en un reporte de cierre de información diaria. Los reportes de las compañías se revisan y se firman; de esta manera las compañías tienen constancia del trabajo realizado en un día.

Luego del cierre de información, en los pozos en tierra de CUPET se envía un resumen del reporte diario, llamado parte diario de perforación, el cual se analiza en la DIPP en el consejo de las 8:30 AM. Analizando este parte, se toman las decisiones pertinentes, con respecto a las operaciones en los pozos. Este parte contiene información referente al avance diario, parámetros de la barrena, inventario de material pesante, estado del combustible, parámetros de dirección del pozo, las actividades realizadas en el día, así como las necesidades del pozo para continuar la perforación. Durante la perforación se realizan los procesos de encamisado y cementación, pruebas y terminación. Estos procesos generan sus propios reportes dado que se realizan en un momento determinado de la perforación.

Luego de la terminación y ensayo, los pozos inician la fase de producción, pasan a manos de una de las empresas de producción y extracción (Centro, Occidente o Majagua), pero esta entidad continua participando, ya que es la encargada de contratar y monitorear los servicios de Intervención de Pozos, que se realizan con el objetivo de reparar o realizar trabajos de control sobre los pozos que ya están en producción (4).

1.4 Análisis de otras soluciones existentes.

1.4.1 SIPP.

Está concebido para ser el sistema único de gestión del proceso de perforación y de la gestión de la información resultado de la perforación en los pozos de petróleo en tierra en la Unión CUPET. Además a mediano plazo, planea introducirse en el mercado como una solución nueva, revolucionaria y adaptable a las necesidades específicas de las empresas. Cuando otros competidores intentan colocarse con una solución general, SIPP se enfoca en las necesidades del usuario final; es totalmente adaptable y ajustable. Este sistema propone una gestión del proceso y manejo de información, transparentes a la compañía o empresa que lo utilice, dado que, fue desarrollado sobre la base de los procesos genéricos identificados, dentro del área de la perforación en tierra.

Características:

- **Gestión y control de la información resultado del proceso de perforación:**
 - Control de las herramientas, combustible, costos (Operaciones y de Servicios contratados), productos químicos, etc.
 - Registro de actividades, metraje, aditivos añadidos, pérdidas de propiedades del fluido de perforación, dirección del pozo (Inclinometría), entre otros.
 - Registro geológico de las formaciones, descripciones de las muestras (sartas de perforación).
- **Administración de Seguridad.**
- **Generación de Reportes.**
- **Manejo de Archivos:** El sistema permite Exportar e Importar archivos xls y txt. Lo cual permite coexistir con otros sistemas en el ambiente de trabajo.
- **Gestión de Nomencladores:** Posibilita que el sistema funcione con una base común de conocimiento e información, lo cual posibilitará homogenizar el contenido de la información que se maneja.
- **Gráficas de Indicadores de Perforación.**
- **Aplicación Distribuida:** El sistema está distribuido por cada uno de los nodos que intervienen en el proceso (Pozo y DIPP). Permite que el sistema sea ligero y con mayor rendimiento, posibilita el trabajo sin conexión.
- **Seguridad (Confiabilidad, disponibilidad e integridad):** Esta característica es muy importante para las aplicaciones Web ya que estas deben garantizar el nivel de seguridad donde se cumplan estos tres requisitos. La confiabilidad se garantiza creando cuentas de acceso para cada uno de los usuarios, además de encriptarse toda la información sensible de los mismos. El sistema siempre debe estar disponible y esto se garantiza con la distribución de la aplicación por los nodos que intervienen en los proceso de negocio (Pozo y DIPP).

CAPÍTULO 1: Fundamentación Teórica.

- **Ligera y amigable:** Los entornos de trabajos son sencillos y solamente contienen los elementos para realizar la labor según el rol que tenga el usuario. No se sobrecargan los espacios de trabajo con publicidad, ventanas emergentes o banners.
- **Modulada:** Minimiza el costo de soporte y mantenimiento. Además permite realizar estas tareas y la aplicación permanece disponible.
- **Alto Rendimiento:** La optimización interna de las peticiones, sin la utilización de sentencias SQL, si no utilizando una herramienta de ORM (Object Relationship Mapping, por sus siglas en inglés).
- **Desarrollada usando herramientas de software libre:** La mayoría de las herramientas son desarrolladas y liberadas bajo los términos de licencias GPL (Software Libre), además de estar soportadas por comunidades internacionales de desarrollo lo cual garantiza el avance y actualización de las mismas.
- **Multiplataforma:** La aplicación puede ser desplegada sobre cualquier sistema operativo, ya que el servidor que se usa posee esta capacidad.

SIPP está implementado en plataforma web, por las características del proceso, donde existe un ambiente multi-roles en un mismo espacio de trabajo físico. De esta manera no es necesario instalar en cada una de las PC clientes el sistema, solo con tener un navegador y conectarse al servidor ya se está trabajando en el sistema. Otro elemento a destacar son las potencialidades de la web para la presentación de información, que es mucho más versátil que la plataforma de escritorio. Para mejor entendimiento del ambiente de despliegue se presenta la siguiente vista:

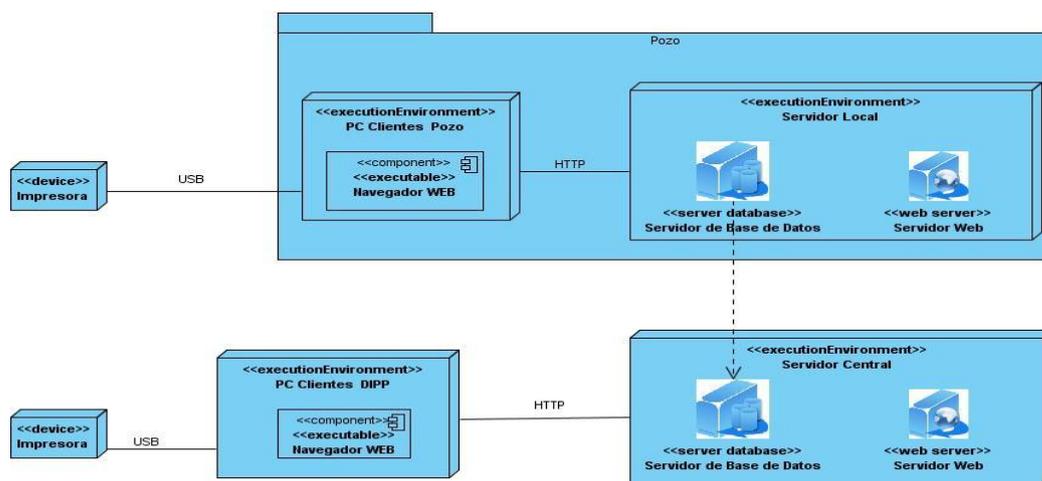


Figura 1: Vista de Despliegue.

[Anexo 1](#)

CAPÍTULO 1: Fundamentación Teórica.

1.4.2 MasterView.

Es una base de datos corporativa para los datos de la perforación y de las operaciones de pozo, la cual suministra un flujo ininterrumpido de datos entre las aplicaciones de la empresa Peloton y productos externos. MasterView consolida los datos entre programas que dependen el uno del otro para dar a los usuarios una vista completa del ciclo de vida del pozo, y empleando tecnologías robustas y comprobadas para garantizar que todos los componentes del back-end trabajen y escalen como se espera.

Las alertas de MasterView y los Objetos de Integración suministran información completa y precisa en una forma consistente; ofreciendo una vía poderosa y flexible para automatizar el flujo de trabajo al reconciliar datos entre diferentes sistemas de información. Emplea el mismo modelo de seguridad de base de datos desarrollados para WellView, RigView y SiteView. La seguridad es consistente para todas las aplicaciones de MasterView y puede ser administrada usando el mismo grupo de credenciales para la base de datos (5).

MasterView provee:

- Precisión de datos e integración.
- Tiempo de programación reducido.
- Apoyo y entrenamiento por Peloton.
- Protección mejorada de las inversiones de desarrollo.
- Integración hecha a la preferencia del cliente.
- Habilidades de desarrollo re-utilizables en todos los productos de Peloton.
- Independencia de base de datos---Oracle, SQL Server, MS Access.
- Fácil implementación (5).

[Anexo 2](#)

WellView

WellView es un archivo de pozo corporativo completo. Desde la petición de perforar un pozo hasta el abandono, registra todos los cambios y operaciones a través del ciclo de vida del pozo. Con su poderoso estado mecánico, reportes y sistemas de búsqueda, WellView coloca la información en las manos de la gente que más la necesita sin las limitantes de archivos de papel tradicionales o reportes diarios.

Enfocado en las operaciones de producción del pozo y estado mecánico histórico. Para abarcar más en el ciclo de vida de un pozo, la empresa Peloton expandió este componente para manejo de datos de perforación en 1997. La versión actual crece con la fortaleza de versiones pasadas mejorando el modelo de datos, herramientas de administración y capacidades analíticas (5).

CAPÍTULO 1: Fundamentación Teórica.

Principales características:

- Permite el análisis de ciclo de vida de pozo integrado y capacidad de visualización.
- Personalización de los reportes diarios rápidamente
- Administración datos del pozo.
- Visualización datos de fondo de yacimiento.
- Permite ver las intervenciones del pozo.

[Anexo 3](#)

RigView

Es un sistema interactivo para la planificación de los programas de trabajo de los equipos de perforación y complementación capaz de administrar cargas de trabajo de pozos tales como perforación, y reacondicionamiento, servicio y pruebas. Siendo un componente de MasterView, su base de datos de perforación y operaciones de pozos. RigView se integra de forma ininterrumpida con MasterView y los sistemas locales permitiéndoles a los usuarios compartir horarios y archivos, dar prioridad a trabajos, administrar notificaciones de cambio y crear reportes personalizados.

La interfaz interactiva de RigView, el Explorador de Equipos, le permite:

- Mantener un poderoso horario de trabajo de taladro que está siempre al día
- Arrastrar y colocar gráficamente trabajos para reprogramarlos
- Compartir archivos, crear consultas de base de datos, agrupar y administrar datos de equipos y de archivos
- Ver y administrar fechas plazo de trabajos, estatus de completación, fechas cruciales alcanzadas y pre-requisitos
- Revisar la disponibilidad de contratos de taladro y de proveedores
- Crear reportes personalizados y resumen de operaciones
- Notificar socios, proveedores y otros de cambios de horarios y programas
- Sistema de Horario de Trabajo de Taladros - Interactivo, sistema de horario de trabajo de taladro visual, administración de datos de taladro y consultas de base de datos flexible, programación de horarios de trabajo de equipo, dinámico, reporte interactivo (5).

[Anexo 4](#)

SiteView

Tiene un enfoque amplio, a partir de la información superficial como lo permiten, la construcción, recuperación y rehabilitación, a través de instalaciones completas y seguimiento de proyectos y gestión de activos. Gestión Integral de Operaciones y Gestión de proyectos de principio a fin con las estimaciones presupuestadas y de campo, planos del proyecto, hitos, averías de proveedores y costos,

CAPÍTULO 1: Fundamentación Teórica.

y los paquetes acumulativos de múltiples sitios. Coleccionando los datos completos y presentando informes permite a los administradores rastrear toda la información de la superficie terrestre, incluidos los propietarios de tierras, alteración de la superficie de arrendamiento, y la historia de contacto.

Con SiteView, puede realizar un seguimiento de cada pieza de equipo de superficie en el sitio, de un alto nivel en detalles granulares. Hechos básicos tales como la posición, el recuento, marca, modelo, tipo y número de serie se definen, así como los detalles específicos de atributos técnicos como el uso de combustible, el tipo de conductores, la bomba de caballos de fuerza, y el tubo de escape de diámetro. También puede realizar un seguimiento todos los componentes auxiliares o secundarios, como la instrumentación, controles, o válvulas asociados a un determinado equipo. Contribuye a optimizar el uso del equipo, ayudando a analizar el tiempo de ejecución y el tiempo de inactividad (5).

Las principales características de SiteView son las siguientes:

- Bueno resumen del ciclo de vida: base de datos completa SiteView de seguimiento de todas las operaciones del sitio, incluyendo la construcción, medio ambiente, instalaciones, tuberías, gestión de residuos y el seguimiento de los costos.
- Los datos de canalización de seguimiento: Es posible grabar la tubería completa desde el nivel alto hasta el detalle los atributos de cada segmento de tubería que comprende la tubería.
- Seguimiento de las emisiones: Introduzca las emisiones para cada pieza de equipo, ya sea regulada por el permiso o no. Las emisiones totales se puede enrollar en cualquier período de tiempo, teniendo en cuenta el tiempo real de análisis de emisiones y de gestión, para las instalaciones o tuberías.
- Transferencia de Materiales: Pista de las transferencias de material para grabar el movimiento de los equipos y el inventario de un sitio a otro. Para una historia completa de informes, los ingenieros de las instalaciones y los gestores de activos pueden ver cuando el equipo fue recibido en el lugar, instalar, eliminar y fuera del sitio.
- Visualización del sitio del pozo: Visualización de los sitios, así se hace fácil con bocetos de sitios que representan los equipos de superficie y las características topográficas en cada etapa de la construcción (5).

[Anexo 5](#)

Integración con WellView

- Compartir información pertinente entre WellView y SiteView. La información fluye mucho mejor entre la perforación del pozo y los grupos de superficie, aprovechando el flujo de trabajo optimizado los datos de WellView.

Multi-Site Información

CAPÍTULO 1: Fundamentación Teórica.

- Crear informes a través de cualquier número de sitios seleccionados. Para una visión de gran alcance en las operaciones en general, utilizar los informes acumulativos y de resumen para comparar los sitios que coinciden con los mismos criterios.

Búsqueda avanzada

- Crea fácilmente consultas para encontrar sitios que coincidan con uno o muchos criterios. Rápidamente identificar y extraer los sitios específicos dentro de una base de datos, independientemente de cuántos sitios están presentes.

Consolidar los usuarios finales

- Muchos grupos de usuarios pueden aportar datos al archivo maestro, reducir la duplicación de esfuerzos y la centralización de la captura de datos para que cada grupo posterior se basa en los esfuerzos de la anterior (5).

1.4.3 InfoPerf.

Es un sistema que se encarga de la gestión de la información en los pozos en perforación. Posee una interfaz de usuario sencilla y amigable que maneja grandes volúmenes de datos complejos y de perforación de perforación que aparecen en un formato conciso y de fácil acceso por diferentes tipos de usuarios. Permite además la visualización de la perforación de pozos progreso a través de un período de tiempo y comparar dicha evolución con las curvas de perforación de referencia. Proporciona información detallada relacionada con maniobras, bien por fecha y hora, incluyendo la profundidad alcanzada por cada uno.

Con un módulo de gráficos que muestra diferentes aspectos del proceso de perforación, tales como curvas de avance, curvas de poco rendimiento, y las curvas de costo. Los gráficos están vinculados directamente a su origen de datos, permitiendo a los usuarios acceder a los datos detallados, directamente. Incluye una función de exportación de datos simples que se pueden utilizar como fuente para los informes de acuerdo a cada función y las necesidades de los usuarios. El sistema permite definir diferentes entidades al cargar las tablas maestras, la configuración de las bandas, tipos de componentes de BHA, carcasa de tipos, de revestimiento y de poco diámetro, áreas, yacimientos de petróleo, y otros datos que permite la carga correcta y completa de los informes diarios (6).

Las principales características que este sistema permite son:

- Gestión de datos y proceso de perforación y reacondicionamiento de seguimiento.
- Simple y amigable interfaz de usuario.
- Aplicación modular, con funcionalidades diferentes y los usuarios de destino.
- Las costas y gastos de seguimiento.
- Módulo de artes gráficas: el progreso de perforación, rendimiento y los costos.
- Datos históricos de pozos.

CAPÍTULO 1: Fundamentación Teórica.

- Consultas definidas por el usuario.
- Informes técnicos y la información de costos (6).

[Anexo 6](#)

1.4.4 WellSight v4.5.

El software WELLSIGHTMR4.5 se utiliza para administrar la información técnica captada en las operaciones de Baroid Fluid Services. La información generada por ingenieros de fluidos de perforación se transfiere continuamente a los sitios de oficinas Baroid para ser monitoreados, reconciliados y por últimos almacenados con seguridad en una base de datos Oracle en Houston, Texas. Aproximadamente 60 oficinas de Baroid en 52 países recogen información y la usan para analizar condiciones de perforación, desempeño de productos y potenciales brechas tecnológicas. La aplicación WELLSIGHT está diseñada para brindar al ingeniero de campo acceso completo a historiales de pozos cercanos en el sitio del equipo de perforación, obtener datos y generar informes en formato electrónico.

Las principales características que este posee son:

- Monitoreo de fluidos de perforación.
- Análisis de las propiedades del lodo.
- Rastreo de inventarios.
- Consumo de productos químicos.
- Recoge la información litológica.
- Mediciones de direccionales.
- Contabilidad de volúmenes
- Generación de informes de todo lo mencionado anteriormente.
- Generador grafico para construir la columna litológica.
- Posee un generador de reportes que permite hacerlos diarios, mensuales, trimestrales y anuales.
- Permite la exportación de los datos (7).

[Anexo 7](#)

Luego de haber estudiado más de 10 sistemas que actualmente se utilizan en el mundo en el área de la perforación de pozos y hacer profundizado en estos cuatro se obtiene como resultado que:

- Todos los sistemas, a excepción de SIPP, están liberados bajo licencias privativas y aplican políticas restrictivas sobre los cambios a sus software.
- Estos sistemas propietarios, poseen renombre internacional y han sido certificados bajo normas de calidad como es el caso de InfoPerf, en el marco de la empresa InfOil, la cual certificó el estándar de calidad ISO 9001, en el 2008.

CAPÍTULO 1: Fundamentación Teórica.

- Estos productores de software extranjeros poseen todos como mínimo más de 10 años en el desarrollo de estos sistemas.
- Los sistemas no se adaptan al entorno, aunque manejan estándares y normas generales, están diseñados para que el usuario tenga que adaptarse a él.
- Todos a excepción de SIPP están enfocados a la gestión de la información no del proceso de perforación.
- Todos los sistemas, a excepción de SIPP, no potencian el diseño del sistema orientado a roles y no prestan total atención a la interrelación entre las compañías de servicio y los supervisores o administradores de los pozos, en el proceso de gestión de la información.
- Todos a excepción de SIPP, están desarrollados en plataforma de escritorio, lo cual tiene una estrecha relación con el resultado anteriormente planteado.
- Existen puntos donde SIPP y estos sistemas convergen como: gestión del proceso de cambios de BHA, generación de informes (reportes), consulta de la información diaria, visualización de los indicadores de perforación y exportar datos en. xls y txt.

1.5 Enfoques de la Programación.

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), 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 siguiente concepto dentro de la programación Web, es el protocolo encargado de llevar la información que contiene una página Web por toda la red de internet, como es el HTTP (Hypertext Transfer Protocol).
- 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). En la programación Web, el HTML es el lenguaje que permite codificar o preparar documentos de hipertexto, que viene a ser el lenguaje común para la construcción de una página Web (8).

1.5.1 Programación Orientada a Objetos (POO).

La programación orientada a objetos trata de amoldarse al modo de pensar de los hombres intentando simular el mundo real a través de objetos. Los objetos se crean en general a partir de constructores,

CAPÍTULO 1: Fundamentación Teórica.

los cuales mediante la definición de una clase crean una instancia de un objeto de dicha clase. Las principales características que definen a la programación orientada a objetos 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.
- **Jerarquía de clases:** Es el orden de las abstracciones organizado por niveles. Esto quiere decir que, como existen clases que heredan propiedades y métodos de otras clases, se forma una jerarquía entre las clases que conforman un programa.
- **Persistencia:** Es la propiedad de un objeto a través de la cual su existencia trasciende el tiempo (el objeto persiste incluso después de que su creador ha dejado de existir) y el espacio (la localización del objeto se mueve del espacio de dirección en que fue creado).
- **Herencia:** Es la relación que existe entre las clases formando una jerarquía. Esto permite definir propiedades y métodos de un objeto a partir de las propiedades y los métodos de los objetos de las clases con las que está relacionada.
- **Polimorfismo:** Es lo que le permite a los programadores definir un mismo método en varios objetos, pero con distintos comportamientos (9).

1.5.2 Estilos y estándares de codificación.

Se definen estándares de codificación porque un estilo de programación homogéneo en un proyecto permite que todos los participantes lo puedan entender en menos tiempo y que el código en consecuencia sea sostenible.

Por lo general los estándares de programación definen la forma en que deben ser declaradas las variables, las clases, los comentarios, en algunos estándares especifica que datos deben incluirse acerca del programador y de los cambios realizados al código fuente, etc.

El código de un programa lo leen muchas personas, bien para repararlo, bien para ampliarlo o, simplemente, para evaluarlo. Para estos lectores es fundamental que el programa esté bien redactado, con estilo, para que su significado sea claro. Existen una serie de reglas básicas que ayudan a conseguir un texto satisfactorio (10).

1.5.3 Framework de desarrollo.

Un framework es una estructura de software compuesta por componentes personalizables e intercambiables para el desarrollo de una aplicación. Lo que permite que el proceso de desarrollo de una aplicación sea más acelerado, ya que encapsula operaciones complejas en instrucciones sencillas

CAPÍTULO 1: Fundamentación Teórica.

listas para ser utilizadas en un nuevo proyecto. Además estos proponen un conjunto de patrones que ayudan distribuir, organizar y estructurar el sistema en componentes lógicos; por lo que las soluciones que proponen dichos patrones son de estrictos cumplimiento durante el proceso de desarrollo de la aplicación.

Características:

Características fundamentales con que cuentan los framework de desarrollo web, las que permiten el amplio uso de estos y su gran aceptación para la elaboración de sistemas web.

- **Abstracción de URLs y sesiones:** No es necesario manipular directamente las URLs 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 Base de Datos (PostgreSQL, Oracle, MySql, MS SQL, etc.), XML, etc.
- **Controladores:** La mayoría de frameworks implementa 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 usuario y contraseña, y permiten restringir el acceso a determinadas páginas a determinados usuarios (11).

1.5.4 Subsistema Web.

Se entiende por sistema web o aplicación web, a un conjunto de páginas web (estáticas o dinámicas), entendiéndose por página web estática a aquella en que es mostrada siempre al cliente de la misma forma sin importar las veces que sea llamada y por página web dinámica a la página cuyo contenido cambia en dependencia de los permisos de acceso que tenga el usuario que la solicita, de la cantidad de veces que sea llamada por el usuario, etc.

Características:

Las características principales de una aplicación web son:

- Se encuentra alojada en un Servidor Web.
- Son accesibles mediante el internet, usando un navegador web.
- La lógica del sistema se ejecuta en el servidor, mientras que el cliente solamente representa los datos.
- Acceso inmediato y desde cualquier lugar: las aplicaciones basadas en tecnologías web no necesitan ser descargadas, instaladas y configuradas. Además pueden ser accedidas desde cualquier computadora conectada a la red desde donde se accede a la aplicación.

CAPÍTULO 1: Fundamentación Teórica.

- Actualización: las aplicaciones web siempre se mantienen actualizadas y no requieren que el usuario deba descargar actualizaciones y realizar tareas de instalación.
- Compatibilidad multiplataforma: ya que una misma versión de la aplicación puede correr sin problemas en múltiples plataformas como Windows, Linux, Mac, etc.
- Menos requerimientos de hardware: este tipo de aplicación no consume (o consume muy poco) espacio en disco y también es mínimo el consumo de memoria RAM en comparación con los programas instalados localmente. Tampoco es necesario disponer de computadoras con poderosos procesadores ya que la mayor parte del trabajo se realiza en el servidor en donde reside la aplicación.

1.6 Conclusiones Parciales.

Luego de haber abordado los aspectos esenciales para abordar con más profundidad el objeto de estudio, los conceptos asociados al problema y haber realizado el estudio del arte, se llegan a las siguientes conclusiones:

- El área de negocio de la perforación de pozos en tierra, comprende elementos complejos desde el punto de vista técnico, que no serán dominados con la profundidad requerida sin un estudio profundo y sistemático del proceso enfocado a la práctica.
- El estado actual del sistema SIPP, en comparación con otros sistemas a nivel internacional aún está por debajo de las prestaciones que estos brindan; su conceptualización, mejora aspectos como: Supervisión y control del Proceso y enfoque de proceso.
- En futuras versiones de SIPP se debe hacer énfasis en conceptualizar y desarrollar un módulo o subsistema que soporte el trabajo en la planificación y proyección de nuevos pozos, análisis de datos históricos.

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

2.1 Introducción.

En el presente capítulo las herramientas y tecnologías utilizadas para el desarrollo del subsistema pozo. Se abordan elementos como el lenguaje de programación, servidor de aplicación, servidor de base de datos, entorno integrado de desarrollo, herramienta case, entre otros. También se exponen las técnicas de aseguramiento de la calidad del código fuente y el estilo y estándar de codificación seleccionado.

2.2 Lenguaje de Programación.

2.2.1 PHP 5.

PHP es un lenguaje interpretado que está diseñado para desarrollo web y puede ser incrustado dentro de código HTML. El código PHP generalmente es ejecutado en un servidor web, lo que le permite ser invisible al navegador y al cliente. Debido a que es liberado bajo una licencia considera software libre este puede ser utilizado en la mayoría de los servidores web y en casi todos los sistemas operativos sin costo alguno.

Principales características:

- Tiene soporte para casi todas las bases de datos por ejemplo (MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras).
- Es un lenguaje multiplataforma por lo que puede ser utilizado sobre cualquier sistema operativo.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Permite aplicar técnicas de programación orientada a objetos (12).

PHP v5.0 incluye además de las de las versiones anteriores:

- Incorpora el Zend Engine 2.
- Nueva extensión SOAP para una fácil creación de servicios Web.
- Soporte para el motor de bases de datos SQLite.
- Mejoras en rendimiento.
- Manejo de excepciones.
- Permitir generar código productivamente en el menor tiempo posible (13).

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

2.2.2 HTML.

HTML, HyperText Markup Language por sus siglas en Ingles, constituye uno de los lenguajes más utilizado para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto de las páginas, así como para complementar el texto con objetos tales como imágenes.

Principales características:

- Las etiquetas tienen que seguir un orden piramidal, las primeras que se abren son las últimas que se cierran, estas pueden contener atributos los cuales definen las propiedades del elemento.
- No tiene distinción entre mayúsculas y minúsculas. Cuando es importante hacerlo, como al poner un título o un atributo, hay que ponerlo entre comillas dobles.
- HTML tiene unas reglas estructurales que indican dónde pueden y no pueden ir los elementos, esto obliga a tener un orden lógico en la escritura del código.
- Los espacios, tabulaciones, líneas en blanco y retornos de carro del documento HTML se ignoran, tomándose como un único espacio en blanco. Esto permite añadir espacios para aumentar la claridad del documento (14).

2.2.3 CSS.

CSS es un lenguaje de hojas de estilos en cascada creado para mejorar la interfaz gráfica de las aplicaciones web. Con el uso de CSS se hace posible separar la presentación del contenido. La creación de hojas de estilo es fácil pues solo se necesita tener un conocimiento básico de HTML para poder diseñar un estilo propio que sea agradable para el cliente.

Estos son los aspectos positivos del uso de CSS:

- Control de la presentación de muchos documentos a partir de una única hoja de estilo, la que se puede utilizar con tan solo poner un link en el documento HTML.
- Un control más preciso de la presentación de la página web.
- La posibilidad de aplicación de diferentes presentaciones a diferentes tipos de medios (pantalla, impresión, etc...).
- Las numerosas técnicas avanzadas y sofisticadas.
- Con la utilización de ciertos programas, como Top StyleLite 3, la creación de una hoja CSS es muy sencilla y eficaz (15).

2.2.4 JavaScript.

Javascript es un lenguaje dinámico orientado a objetos con soporte de funciones anónimas, y cerraduras. Precisamente definir a javascript de esta manera genera típicamente más confusión que

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

no conocer las características de javascript, entonces voy a intentar explicar una a una estas características con ejemplos concretos y fáciles de comprender, que resalten la esencia de cada característica del lenguaje y nos permitan ver más allá sus posibilidades de utilización.

Características:

- Lenguaje dinámico: se dice que Javascript es un lenguaje dinámico ya que posee objetos dinámicos, estos son objetos que pueden variar su definición y comportamiento en tiempo de ejecución. Todos los objetos en javascript son dinámicos.
- Closures: las closures o cerraduras en javascript son una estructura del lenguaje que no posee una sintaxis fácil de explicar, pero que básicamente nos permite acceder a variables que fueron declaradas fuera de la función que estemos escribiendo, aún después de que la función mayor hubiese terminado su ejecución.
- Funciones lambda o anónimas: javascript nos permite crear funciones sin nombre (anónimas) y poder asignarlas a una variable, ya que las funciones también son un objeto. Por esta razón podemos pasar funciones como parámetros a otras funciones, e inclusive podemos devolver funciones como resultado de una función.
- Herencia Prototipal: la herencia prototipal es uno de los aspectos más discutidos de javascript, y se basa en el hecho de que cada objeto en javascript (y en javascript todo es un objeto) posee un prototipo, como miembro de sí mismo. Por lo tanto, todo objeto(A) que se cree a partir de un objeto (B), heredará todos los miembros del prototipo de ese objeto (B). Finalmente si se solicita un miembro m sobre un objeto determinado y este objeto no posee una definición de su miembro m, se le solicitará ese miembro m a su prototipo, el cual es común a todos los objetos de ese tipo (16).

Después de un estudio de los diferentes lenguajes de programación Web existentes realizado por el grupo de arquitectos, se determinó que para el desarrollo del Subsistema Pozo en su versión 2.0, se utilice PHP 5.3 como lenguaje de programación del lado del servidor. Este es un lenguaje libre, capaz de conectarse con la mayoría de las bases de datos que se utilizan en la actualidad, este permitirá tener una mejora considerable de la cache² lo que posibilitará un mejor rendimiento del subsistema, este lenguaje permite aplicar el paradigma de POO. Como lenguajes de programación del lado del cliente se utilizará HTML para la confección de las vistas, CSS que es un lenguaje que permite proporcionarle una interfaz agradable al cliente, aplicándole estilos a la aplicación. JavaScript, el cual permite crear un ambiente dinámico y hacer uso de él para realizar validaciones del lado del cliente.

² Cache: Memoria en la que se almacena una serie de datos para su rápido acceso.

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

2.3 Entorno Integrado de Desarrollo (IDE).

2.3.1 Zend Studio.

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones Web en lenguaje PHP. El programa, 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. Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. La del cliente contiene el 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 (17).

Principales características:

- Soporte para PHP 4 y PHP 5.
- Contiene una ayuda contextual con todas las librerías y funciones de PHP.
- Tiene completamiento de código generando nombre de funciones y parámetros que esta reciben.
- Soporte para gestión de grandes proyectos de desarrollo.
- Ofrece una ayuda a la hora de escribir, pues permite marcar a que elementos dentro del código pertenecen los inicios y cierre de etiquetas, paréntesis o llaves.
- Posee una gran integración con el control de versiones.
- Contiene un Bookmarks que brinda la posibilidad de encontrar rápidamente en función de cualquier archivo.

Se decide utilizar el Zend Studio como IDE, producto que este permite agilizar el desarrollo web y simplificar proyectos complejos. Es un IDE muy completo con resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase, así como detección de errores de sintaxis en tiempo real y un manual de PHP integrado. Además de contar con una gran compatibilidad con el control de versiones.

2.4 Framework de Desarrollo.

2.4.1 Symfony.

Symfony es un framework diseñado para agilizar el desarrollo de las aplicaciones web. Este fue desarrollado por completo con PHP 5; se puede integrar a los ORM (Propel y Doctrine) los cuales crean una capa de abstracción de la base de datos que permite a los desarrolladores realizar las consultas SQL de una forma más sencilla. Es compatible con los gestores de base de datos más utilizados en la actualidad (MySQL, PostgreSQL, Oracle, entre otros). Es libre y que encapsula

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

operaciones complejas en instrucciones sencillas ya que este incluye helpers (contenedores de código) que brindan esta posibilidad, así como de la realización de validaciones y autocompletado de código (18).

Características:

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y *nix estándares).
- Independiente del sistema gestor de bases de datos. Su capa de abstracción y el uso de Propel, permiten cambiar con facilidad de SGBD en cualquier fase del proyecto.
- Utiliza programación orientada a objetos, de ahí que sea imprescindible PHP 5.
- Sencillo de usar en la mayoría de casos, aunque es preferible para el desarrollo de grandes aplicaciones Web que para pequeños proyectos.
- Aunque utiliza MVC (Modelo Vista Controlador), tiene su propia forma de trabajo en este punto, con variantes del MVC clásico como la capa de abstracción de base de datos, el controlador frontal y las acciones.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador sólo 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 adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con las bibliotecas de otros fabricantes.
- Una potente línea de comandos que facilitan generación de código, lo cual contribuye a ahorrar tiempo de trabajo (18).

El uso del framework Symfony, unido a Propel, posibilita el uso de las tablas de la base de datos como objetos, lo que facilita la manipulación de los datos en la lógica de negocio contenida en el modelo. La abstracción de la base de datos provee de un conjunto de clases de acceso a datos organizada que contribuye a la organización de las clases del modelo. La estructura propuesta de MVC (Modelo-Vista - Controlador), aplicando el patrón controlador frontal por defecto, permite que las peticiones a la aplicación se hagan a través de un único punto de acceso, dejando a las clases controladoras la única responsabilidad de la lógica de aplicación. De esta manera se aligera el trabajo de los controladores y estas clases pueden desempeñar mejor su responsabilidad como clases.

2.5 Estilos y Estándares de Codificación.

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

Symfony tiene definido sus propios estilos y estándares de codificación en su página principal.

- Nunca use las tabulaciones en el código. La sangría se hace por pasos de 2 espacios:

```
<?php
class sfFoo
{
    public function bar()
    {
        sfCoffee::make();
    }
}
```

- No ponga espacios después de un paréntesis de apertura y antes de un cierre:

```
<?php
if ($myVar == getRequestValue($name)) // correcto
if ( $myVar == getRequestValue($name) ) // incorrecto
```

- Llaves van siempre en su propia línea.
- Utilice llaves para indicar la estructura de control del cuerpo, independientemente del número de declaraciones que contiene.
- Symfony está escrito en 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 uno comente en una línea debe estar en este formato:

```
<?php
// space first, with no full stop needed
```

- 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:

```
<?php
class sfCoffee
{
    const HAZ_SUGAR = true;
}
var_dump(sfCoffee::HAZ_SUGAR);
```

- Para comprobar si una variable es nula o no, no utilice la función de PHP is_null ():

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

```
<?php
if (null !== $coffee)
{
    echo 'I can haz coffee';
}
```

- 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.

```
<?php
$a = "Texto de ejemplo";
```

- Concatenación:

Para concatenar Strings se utilizará el operador “.” (punto), con un espacio entre medio para mejorar la lectura:

```
<?php
$company = 'Zend' . ' ' . 'Technologies';
```

- 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:

```
<?php
if ($a != 2)
{
    $a = 2;
}
elseif ($a == 3)
{
    $a = 4;
}
else
{
    $a = 6;
}
```

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

➤ 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 (10).

Se utiliza el estilo y estándar de codificación propuesto por el framework Symfony, dado su fácil utilización y simple aplicación, lo cual contribuye a que el código pueda ser entendido más fácil por desarrolladores. Un framework no restringe al desarrollador en el uso de estilos y estándares, mientras que utilizar los que él propone contribuye a comprender mejor la propia implementación del framework, por lo que de cierto modo se gana en comunicación a través del código entre los que utilizan el framework y los que lo desarrollan.

2.6 Sistema Gestor de Base de Datos.

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Por 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.
- Manipular la base de datos: realizar consultas, actualizarla, generar informes.

2.6.1 PostgreSQL 8.4.

PostgreSQL es un potente motor de base de datos que presenta atomicidad, es consistente y su costo de adquisición es nulo por ser una herramienta de código abierto. Es altamente extensible, ventaja que le permite soportar distintos tipos de datos complejos (19). Por lo que se puede afirmar que es un sistema gestor de base de datos de avanzada de los más utilizados a nivel mundial. Este presenta las siguientes ventajas:

- Restauración de bases de datos en procesos paralelos, que acelera la recuperación de un respaldo hasta 8 veces mayor.
- Nuevos métodos del ejecutor para anti joins y semijoins (EXISTS y NOT EXISTS).
- Privilegios por columna, que permiten un control más granular de datos confidenciales.
- Reducción de la memoria en el manejo de los disparadores.
- Configuración de lenguajes por bases de datos, lo cual hace a PostgreSQL más útil en entornos con múltiples idiomas.
- Actualizaciones desde 8.3 a 8.4 con muy bajo tiempo de inactividad, gracias al uso de pg-migrator beta.
- Nuevas herramientas para monitoreo de consultas, que le otorgan a los administradores mayor información sobre la actividad del sistema.

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

- Consultas recursivas (WITH RECURSIVE).
- LIMIT puede estar basado en una subconsulta.
- Nuevo control de estructuras con la cláusula CASE.
- Soporta argumentos por defecto en funciones (19).

2.7 Servidor Web.

Un servidor web es un programa que atiende y responde a las diferentes peticiones realizadas por los usuarios a través de los navegadores, el cual permite construir páginas web usando el protocolo HTTP o el protocolo HTTPS. En este trabajo se hará uso de un servidor web debido a que en el entorno que debe operar el sistema existen altas cifras de clientes solicitando un mismo servicio.

2.7.1 Apache.

Apache es un programa de servidor HTTP Web de código abierto. Actualmente es uno de los servidores Web más utilizados en la red. Es un software libre multiplataforma, con soportes de bases de datos SQL.

Características:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código fuente, abierto.
- Apache es un servidor altamente configurable de diseño modular.
- Trabaja con gran cantidad de Perl, PHP y otros lenguajes de script.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.
- Tiene una alta de configuración en la creación y gestión de registros.
- 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.
- Multi-hilos. Esto permite atender varias peticiones al servidor concurrentemente.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor (20).

Se utiliza el Apache como servidor web pues hoy en día se considera la plataforma web libre más utilizada del mundo, pues aumentan cada día el número de usuarios que permiten este código fuente abierto en su infraestructura, es muy usado en nuestra universidad con amplia disposición de documentos e información, además de tener gran aceptación en toda la red. Su característica multi-hilos hace que las aplicaciones web tengan un tiempo de respuesta mejor, lo cual influye positivamente en la aceptación de los clientes.

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

2.8 Visual Paradigm.

Visual Paradigm es una herramienta CASE profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue.

- Permite generar código en lenguajes tan utilizados como: Java, C++, PHP, C #, Delphi, Perl.
- Permiten incorporar los dibujos de Visio en cualquier diagrama de UML. Muchas herramientas CASE realizan solo diagramas de UML normales, Visual Paradigm posibilita modelar hardware dominio-específico, el software, conectando una red de computadoras los componentes usando sus iconos, más allá de las anotaciones de UML normales.
- Es multiplataforma, Visual Paradigm está disponible en muchas plataformas incluso Windows, Linux, el Mac OS X, etc (21).

Para la realización de este trabajo se ha decidido utilizar Visual Paradigm ya que a diferencia de otras herramientas de este tipo esta es una herramienta multiplataforma que utiliza “UML”: como lenguaje de modelado. El lenguaje de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML.

2.9 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, mantenibilidad, 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 (22). 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 (22).

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 (22).

CAPÍTULO 2: Tendencias y tecnologías actuales a utilizar.

Refactorizaciones: Ayuda a resaltar o recuperar la claridad 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 (22). 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: Las pruebas de aceptación permiten que el cliente evalúe el funcionamiento de la aplicación. Estas pruebas deben ser escritas por personas que desconocen el funcionamiento interno del sistema, asistidas por gente especializada en pruebas (22).

2.10 Conclusiones parciales.

En este capítulo se realizó un análisis de las técnicas, tecnologías y herramientas a utilizar para el cumplimiento del objetivo de este trabajo, se arriban a las siguientes conclusiones:

- Utilizar un Framework de Desarrollo reducirá el costo de tiempo y esfuerzo para la implementación del Subsistema Pozo del Sistema de Manejo Integral de Perforación de Pozos de Petróleo y Gas Versión 2.0.
- El uso herramientas y tecnologías de software libre multiplataforma permitirá la portabilidad del resultado.
- La definición de estilos y estándares de codificación dará la posibilidad, de un mejor entendimiento del código fuente.
- La implementación de técnicas de aseguramiento de la calidad del código durante el proceso de desarrolló, contribuirá a la calidad final del producto.

CAPÍTULO 3: Presentación de la solución y validación.

3.1 Introducción.

En este capítulo se presenta las características funcionales de la solución obtenida. Se expone el diagrama de casos de uso del sistema, los componentes, su descripción y los diagramas que los relacionan; todo esto correspondiente a al modelo de casos de uso del sistema y al modelo de implementación, respectivamente. Se exponen los resultados de las pruebas de caja negra realizadas, así como las actas de aceptación del cliente recibidas en última prueba de aceptación.

3.2 Características Subsistema Pozo.

3.2.1 Actores que interactúan con el subsistema.

Los trabajadores que interactúan con el subsistema Pozo son:

- **Supervisor del Pozo:** Es el responsable de la supervisión y control del proceso de perforación en el pozo y tiene permisos para interactuar con toda la información, excepto para introducir los datos de geología.
- **Geólogo del Pozo:** Es el encargado de gestionar la información geológica del pozo³. También se le da la posibilidad de tener acceso a la información de la perforación pero solo a través de los reportes.

2.2.2 Requisitos Funcionales del subsistema.

Los requisitos funcionales expresan características que un sistema debe permitir. Un requisito funcional define el comportamiento interno de un software: cálculos, manipulación de datos y otras funcionalidades que muestran cómo serán llevados los casos de uso a la práctica.

El sistema debe permitir:

RF#1: Control del proceso de perforación de inicio a fin: El sistema debe permitir que se inicie y se termine la perforación del pozo. Esto debe posibilitar que el sistema pueda establecer el tiempo de duración del pozo y por tanto realizar los análisis correctos, en relación a la información gestionada diariamente. Se controlan las actividades como: manejar costos iniciales, cambiar de intervalo, entre otros.

³ Es el resultado del análisis de los cortes (sarta de perforación). Esta información está compuesta por tramos de profundidades entre los cuales se le asigna un valor porcentual del tipo de roca que compone el tramo perforado.

CAPÍTULO 3: Presentación de la solución y validación.

- RF#2: Control de inventarios: El sistema debe permitir llevar un control básico de los inventarios de las principales herramientas utilizadas en la perforación (barrenas, jar y motores de fondo), así como de los productos químicos utilizados en el lodo de perforación⁴.
- RF#3: Controlar el combustible: El sistema debe permitir tener un control estricto del estado del combustible. Este estado puede ser modificado tanto por los abastecimientos, usos o transferencias del mismo. El control de combustible es indispensable, ya que todo en un pozo funciona con electricidad y la electricidad la generan plantas de diesel en el sitio.
- RF#4: Gestión de herramientas: Este requisito se refiere a la posibilidad de que el sistema brinde la opción de guardar todas las composiciones de herramientas utilizadas en la perforación del pozo. Esto fundamenta la necesidad de tener un acceso histórico a ellas, para un mejor soporte para la toma de decisiones.
- RF#5: Gestión y control de procesos diarios: El sistema debe dar la posibilidad de gestionar la información que se genera en un pozo diariamente. También debe permitir al supervisor del pozo tener conocimiento de las informaciones que no han sido registradas.
- RF#6: Control de productos químicos: Este requisito expresa la necesidad de tener un control del uso de los productos químicos y el inventario restante para utilizar en el pozo. Es de suma importancia este control, dado que un faltante de un producto químico indispensable para controlar parámetros de las rocas puede provocar un colapso en la perforación.
- RF#7: Gestión de los datos geológicos: Se necesita que el sistema contenga los datos relacionados con el análisis de los cortes, el cual es realizado por el geólogo de pozo. De esta manera se tendrá directamente y diariamente los datos geológicos de los pozos y no se tiene que esperar a la emisión del parte por el CEINPET.
- RF#8: Generación de reportes: El sistema debe ser capaz de generar reportes asociados a los indicadores de tiempo, costos, pérdidas, geología, combustible, productos químicos y los reportes diarios predefinidos por la DIPP.

3.2.3 Diagrama de Casos de uso del Sistema.

Para comprender de manera más fácil la solución propuesta y para que el lector tenga una idea más clara de la relación entre los Casos de Uso de este subsistema y la interacción de estos con los actores que allí laboran a continuación se muestra el diagrama de CU del Sistema Pozo.

⁴ Sustancia compuesta por productos químicos, el cual garantiza la vida del pozo en perforación. Con este fluido se controlan parámetros de peso, densidad, flujo y hasta los parámetros de dirección del pozo tomados por la herramienta de medición.

CAPÍTULO 3: Presentación de la solución y validación.

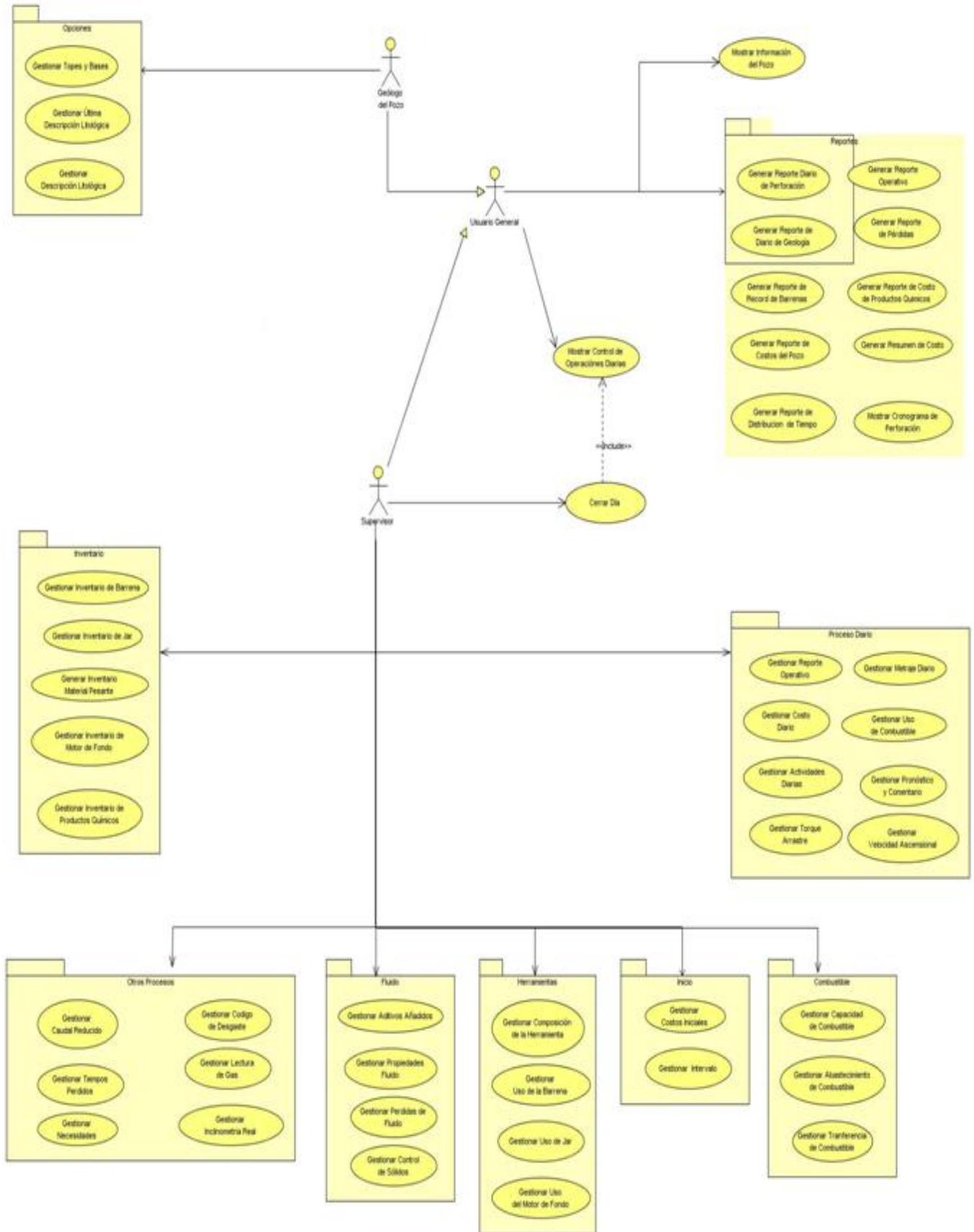


Figura 2: Diagrama de Caso de Uso del Subsistema Pozo.

3.2.4 Subsistema Pozo.

El subsistema pozo está dividido en dos módulos:

El módulo Supervisor: es en donde se encuentran implementados los Casos de Uso (CU) relacionados con el proceso de gestión y control de las informaciones recogidas en los pozos de petróleo en perforación en tierra.

Está formado por los siguientes CU:

- Gestionar Costos Iniciales.
- Gestionar Intervalo de Perforación.
- Generar Cronograma de Perforación.
- Gestionar Inventario de Barrenas.
- Gestionar Inventario de Motor de Fondo.
- Gestionar Inventario de Jar.
- Gestionar Inventario de Productos Químicos.
- Gestionar Inventario de Material Pesante.
- Gestionar Capacidad de Combustible.
- Gestionar Abastecimiento de Combustible.
- Gestionar Transferencia de Combustible.
- Gestionar Composición de Herramientas.
- Gestionar Propiedades de la Barrena.
- Gestionar Jar.
- Gestionar Motor de Fondo.
- Gestionar Reporte Operativo del Pozo.
- Gestionar Metraje Diario.
- Gestionar Costos Diarios.
- Gestionar Uso Diario de Combustible.
- Gestionar Actividades Diarias.
- Gestionar Velocidad Ascensional.
- Gestionar Torque y Arrastre.
- Gestionar Pronósticos y Comentarios.
- Gestionar Propiedades del Fluido.
- Gestionar Aditivos Añadidos.
- Gestionar Control de Sólidos.
- Gestionar Pérdidas.
- Gestionar Caudal Reducido.
- Gestionar Tiempos Perdidos.

CAPÍTULO 3: Presentación de la solución y validación.

- Gestionar Necesidades.
- Gestionar Código de Desgaste.
- Gestionar Lectura de gas.
- Gestionar Inclínometría.
- Generar Reporte Diario de Perforación.
- Generar Reporte Operativo del Pozo.
- Generar Reporte Diario de Geología.
- Generar Reporte de Perdidas.
- Generar Record de Barrenas.
- Generar Costo de Productos Químicos.
- Generar Costos del Pozo.
- Generar Resumen de Costo.
- Generar Distribución de Tiempo.

El módulo Geólogo: es donde se encuentran los Casos de Uso (CU) relacionados con los procesos de descripción de las muestras (cortes), formaciones encontradas en la perforación y la elaboración del reporte diario de geología.

Está formado por los siguientes CU:

- Gestionar Topes y Bases.
- Gestionar Descripción de la Última Muestra.
- Gestionar Descripción Litológica.
- Generar Reporte Diario de Geología

Aunque SIPP es una solución orientada a la medida, este subsistema está diseñado para que se adecue a las necesidades del proceso y no del cliente en específico. Esto se logra, en parte, orientando el sistema a roles: Supervisor de Pozo y Geólogo de Pozo, y aceptando las reglas y restricciones del negocio. Durante el proceso de estudio del arte, se identificó como una de las deficiencias de los sistemas actuales el hecho de que se ocupan solamente de la persistencia de los datos, su manipulación y control de errores, olvidando que el proceso (perforación de pozos en tierra), contiene un conjunto de reglas y restricciones, las cuales también deben ser valoradas. En el subsistema pozo se realizan diferentes procesos asociados al negocio, los cuales agrupan diferentes funcionalidades:

CAPÍTULO 3: Presentación de la solución y validación.



Figura 3: CU Gestionar Costos Iniciales. Interfaz crear.

Iniciar Perforación: Este proceso se refiere al momento inicial cuando se le coloca por parte del grupo técnico de la DIPP el nuevo pozo listo para perforar. En ese momento ya es posible que el supervisor de pozo pueda acceder a su espacio de trabajo. En estas condiciones el pozo, correspondiente, solo tiene los datos asociados al proyecto de pozo, por lo cual se necesitan una cierta cantidad de datos, para que el sistema “comprenda” que físicamente se ha iniciado la perforación en el pozo. Este proceso agrupa los CUs (Casos de Uso) Gestionar Costos Iniciales, Gestionar Capacidad de Combustible y Gestionar Intervalo. Luego de realizar estas acciones en el sistema este habilitará la vista de control de operaciones y permitirá realizar las demás funcionalidades del subsistema.

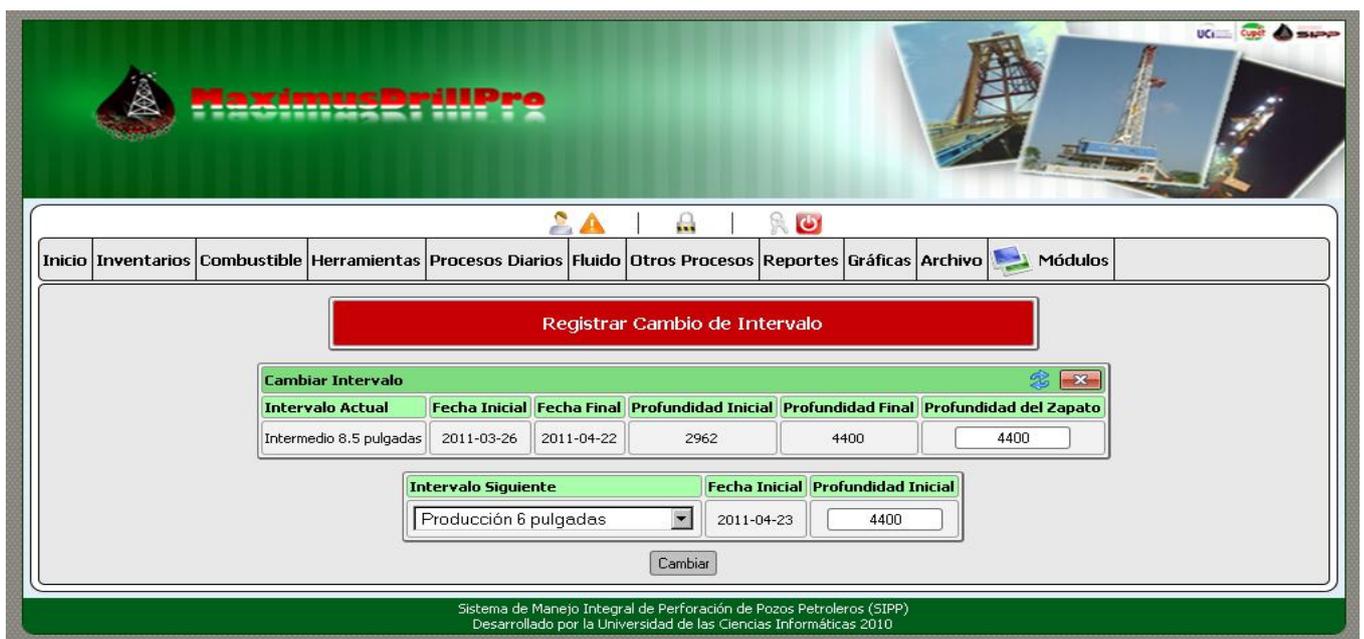


Figura 4: CU Gestionar Intervalo de Perforación. Interfaz cambiar.

CAPÍTULO 3: Presentación de la solución y validación.

Cambio de Intervalo: El proceso correspondiente al cambio de intervalo es crítico en el sistema, ya que no puede ser revertido por el usuario. Esto se debe a que este paso influye sobre los análisis de tiempo, record de barrenas, reportes, entre otros; si se permite que el usuario cambie a placer los intervalos se puede ver comprometida la integridad de los datos del sistema, uno de los requisitos no funcionales críticos de la aplicación en general. Para garantizar que el usuario no vaya a cometer errores en el momento de cambiar el intervalo, se realiza un análisis del cronograma de perforación del pozo y solamente se puede hacer el cambio de intervalo hacia los intervalos reales anteriores y para el siguiente según el cronograma, el cual es la opción que aparece por defecto en el momento del cambio.

The screenshot shows the 'Registrar Composición de la Herramienta + Tubería de Perforación' window. It includes a menu bar with options like 'Inicio', 'Inventarios', 'Combustible', 'Herramientas', 'Procesos Diarios', 'Fluido', 'Otros Procesos', 'Reportes', 'Gráficas', 'Archivo', and 'Módulos'. The main form contains a title bar, a comment field with the text 'BHA para lograr mayor ROP', and checkboxes for 'No usar', 'Barrena', 'Motor Fondo', and 'Jar'. Below this is a table for drill bits and a summary table.

Barrena						
Número	Tipo	No. Serie	Fabricante	IADC	OD	Longitud (m)
7D	MX09	900625	HUGHES	537	8.5 pulgadas	0.25

Cantidad	Herramienta	Descripción	OD (mm)	ID (mm)	Longitud (m)	Acumulado (m)
1	drill	--	12.3	12	6	11.61
Total						11.61

Figura 5: CU Gestionar Composición de Herramienta. Interfaz crear.

Cambio de Composición de la Herramienta: Este es un proceso asociado al cambio de la composición de la herramienta con la que se perfora el pozo. Es una acción continua, la cual no tiene una causa única de hecho, puede ser: cambio de barrena, avería, cambio de formación, cumplimiento de las horas de trabajo de una de las herramientas que la componen, entre otros. Este proceso involucra a los CU Gestionar Inventario de Barrena, Gestionar Inventario de Jar, Gestionar Inventario de Motor de Fondo, Gestionar Composición de la Herramienta (principal funcionalidad), Gestionar

CAPÍTULO 3: Presentación de la solución y validación.

Parámetros de Uso de la Barrena, Gestionar Horas Acumuladas de Jar, Gestionar Horas Acumuladas del Motor de Fondo y Gestionar Código de Desgaste de la Barrena.

Todas estas funciones están presentes o no según la composición que tenga la herramienta de perforación. Se puede apreciar la complejidad de la combinación de acciones para completar el proceso a través del sistema. Al igual que el proceso de Iniciar Perforación se hace necesario un conocimiento profundo del negocio para poder realizarlo correctamente.



Figura 6: CU Cerrar Día. Interfaz cerrar.

Cierre de Información Diaria: Este proceso está asociado al cierre de información que se realiza en los pozos a las 12 AM, con la información del día que acaba de concluir. Este el momento que da la razón de ser al funcionamiento propuesto por el sistema. A través de esta vista el usuario puede conocer las operaciones que ha realizado en un día, lo cual posibilita que pueda tener un control de la información que ha gestionado. Las operaciones en color rojo son referencia a las que no se han realizado, las de color verde a las que se han realizado y las de color azul las que no se necesita un control diario de su estado, pero se colocan para facilitar el acceso en caso de que se desee modificar su estado. Como se aprecia existen 5 columnas:

- Inventarios: Esta columna está asociada al control de los inventarios.
- Obligatorias: Está referida a las operaciones que deben hacerse diariamente, hasta que todas estas operaciones no estén realizadas el sistema no permite cerrar el día. En este tipo de operaciones se hacen análisis según ciertas variables que permiten colocar a una operación

CAPÍTULO 3: Presentación de la solución y validación.

que en un momento pueda ser obligatorio como opcional. También podemos encontrar el hecho de operaciones que en un momento no aparezcan reflejadas, debido a que no tengan que ser registradas. Por ejemplo registrar los parámetros de uso de la barrena, la cual depende de que la composición de la herramienta (BHA por sus siglas en inglés) tenga barrena incluida como herramienta.

- Opcionales: Estas están ligadas a operaciones que pueden o no haberse realizados y aun no existe forma, a través de la persistencia de los datos obligatorios saber si se hace necesario o no realizarlas, por lo que queda a conocimiento y/o consideración del especialista.
- Geólogo: Está asociado a las operaciones que realiza el geólogo. Esto le permite al supervisor de pozo, desde su espacio de trabajo, saber si el geólogo ha o está realizando las operaciones para poder cerrar el día. Estas operaciones también son consideradas opcionales.
- Otros: Es la agrupación de acciones que están asociadas a procesos que no se pueden colocar en ninguna de las categorías anteriores, por lo cual se creó esta. Presentan un funcionamiento a las operaciones opcionales.

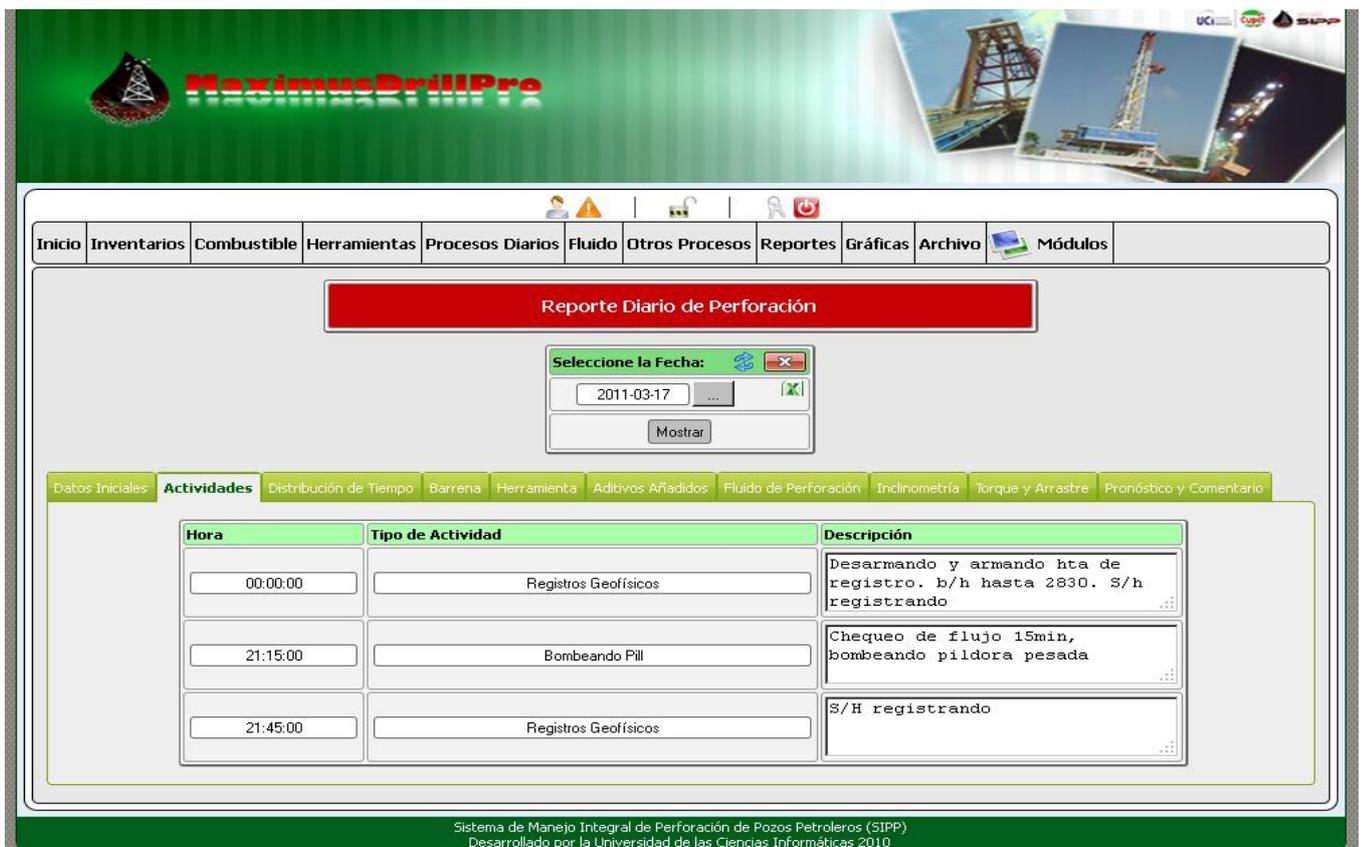
Este realiza además para establecer el control de tiempo diario del pozo, es decir el que sistema conoce en qué fecha se encuentra. A través de la función de cerrar el día el sistema aumenta un día lo cual le permita al supervisor de pozo controlar el momento de cambiar al siguiente día, haciendo que el sistema sea independiente de la fecha y hora del servidor. Esto proporciona un grado de seguridad, ya ante un ataque al reloj del servidor el sistema no se verá afectado directamente.



Figura 7: CU Gestionar Intervalo de Perforación. Interfaz terminar perforación.

CAPÍTULO 3: Presentación de la solución y validación.

Terminar la Perforación: Al igual que el Cambio de Intervalo no puede ser revertido. Una vez terminada la perforación, el sistema detiene la opción de cerrar los días y queda el sistema detenido en el día donde está. Esta operación es totalmente independiente de cualquiera de las variables que maneja el proceso. Esta responsabilidad recae en el especialista, responsable del pozo (Supervisor Jefe de Pozo) Esta acción se pone de manifiesto en dos CU: Mostrar Información del Pozo, donde aparecerá un mensaje informando del hecho y en el Cerrar Día el cual la opción de cerrar día no será visible ni accesible por el usuario.



The screenshot displays the 'MaximusDrillPro' software interface. At the top, there is a navigation menu with options: Inicio, Inventarios, Combustible, Herramientas, Procesos Diarios, Fluido, Otros Procesos, Reportes, Gráficas, Archivo, and Módulos. A red banner in the center reads 'Reporte Diario de Perforación'. Below this, a date selection dialog box is open, showing the date '2011-03-17' and a 'Mostrar' button. The main content area features a sub-menu with options: Datos Iniciales, Actividades, Distribución de Tiempo, Barrena, Herramienta, Aditivos Añadidos, Fluido de Perforación, Inclínometría, Torque y Arrastre, and Pronóstico y Comentario. The 'Actividades' sub-menu is selected, displaying a table with the following data:

Hora	Tipo de Actividad	Descripción
00:00:00	Registros Geofísicos	Desarmando y armando hta de registro. b/h hasta 2830. S/h registrando
21:15:00	Bombeando Pill	Chequeo de flujo 15min, bombeando pildora pesada
21:45:00	Registros Geofísicos	S/H registrando

At the bottom of the interface, a footer reads: 'Sistema de Manejo Integral de Perforación de Pozos Petroleros (SIPP) Desarrollado por la Universidad de las Ciencias Informáticas 2010'.

Figura 8: Generar Reporte Diario de Perforación. Interfaz mostrar reporte.

Reporte de la Perforación: Este proceso corresponde a la generación del Reporte Diario de Perforación del Pozo el cual se elabora por el supervisor de pozo y para la confección del mismo se requiere la recopilación de información relacionada con los partes que envía el Direccional del Pozo, el Geólogo del Pozo, el Químico del Pozo, la compañía encargada de la perforación del pozo. En este reporte se recogen todas las actividades realizadas en el pozo en el día. Para ello se consultan las tablas que contienen la información de los principales datos de la perforación las actividades realizadas, como se distribuyó el tiempo de trabajo, que barrena, herramientas fueron utilizadas, etc. Este es el reporte que se le entrega diariamente a Dirección de Perforación de Pozos para que conozcan el estado de cada pozo.

Litotipo	% Inicial	% Final	Descripción
CALIZA	60	55	
DOLOMITA	40	45	

Figura 9: CU Gestionar Descripción Litológica. Interfaz crear.

Descripción de las muestras litológicas: Este es el proceso encargado de gestionar las descripciones de los litotipos encontrados en intervalos de perforación. El sistema muestra una interfaz en la cual se entra el intervalo de profundidad que se va registrar o modificar, además de los litotipos, % inicial, % final y una descripción de cada uno de los litotipos encontrados, este CU tiene que cumplir que la suma de cada uno de los % iniciales y finales deben dar 100 así como que no se pueden repetir los mismo litotipos en un mismo intervalo de profundidad. Todos estos datos se registran para posteriormente ser consultados.

CAPÍTULO 3: Presentación de la solución y validación.

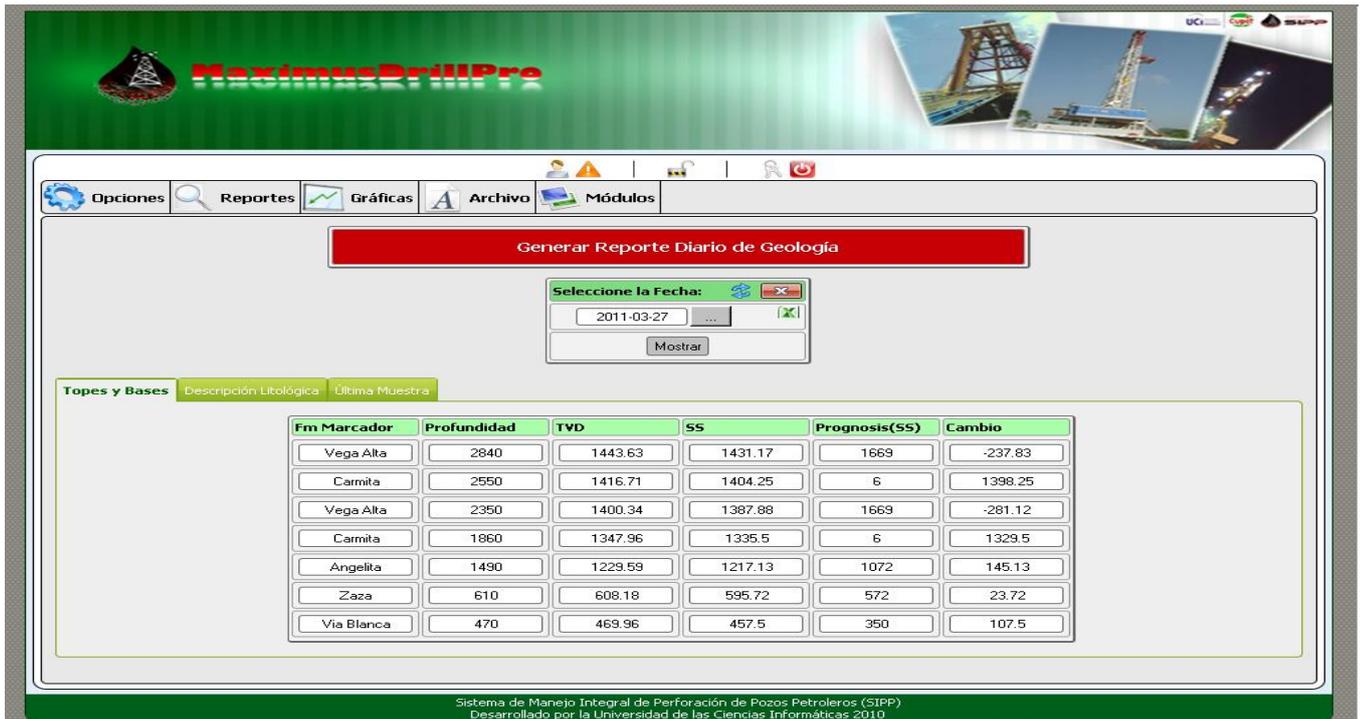


Figura 10: CU Generar Reporte Diario de Geología. Interfaz mostrar reporte.

Reporte Diario de Geología: Este proceso pertenece a la gestión de la información generada por el geólogo del pozo. Importante resaltar que para confeccionar este informe el geólogo del pozo se apoya en los reportes de la compañía direccional y la compañía de servicios de lodo. El geólogo procesa los datos recopilados del análisis de los cortes. Para esto se debe consultar las tablas que contienen toda la información asociada al reporte, es válido aclarar que si aún no se han registrado todos los datos necesarios para la confección del mismo el sistema emitirá un mensaje diciéndole al usuario que lo está consultando que proceso es el que no se ha registrado aun en la aplicación.

Como se aprecia, el subsistema maneja la vida del pozo desde el inicio de la perforación, hasta su fin. Esto se realiza a través de funcionalidades, las cuales juntas y en un orden lógico determinado, conforman procesos que se ven representados en la perforación de pozos en tierra. Estas operaciones son críticas para el funcionamiento correcto del sistema; para poder realizarlos se necesita conocer el proceso a profundidad.

3.3 Modelo de implementación.

El modelo de implementación describe al igual que los elementos del diseño, cómo son las clases, cómo se organizan los componentes de acuerdo al lenguaje de programación utilizado y al entorno de implementación, y muestra las dependencias de los componentes entre sí. Además éste define una jerarquía por su sistema de implementación.

3.3.1 Vista de implementación.

CAPÍTULO 3: Presentación de la solución y validación.

La vista de implementación muestra el empaquetado físico de las partes reutilizables del sistema en unidades sustituibles, llamadas componentes. Una vista de implementación muestra los elementos físicos del sistema mediante componentes, así como sus interfaces y dependencias entre componentes. Los componentes son piezas reutilizables de alto nivel a partir de las cuales se pueden construir los sistemas. (23)

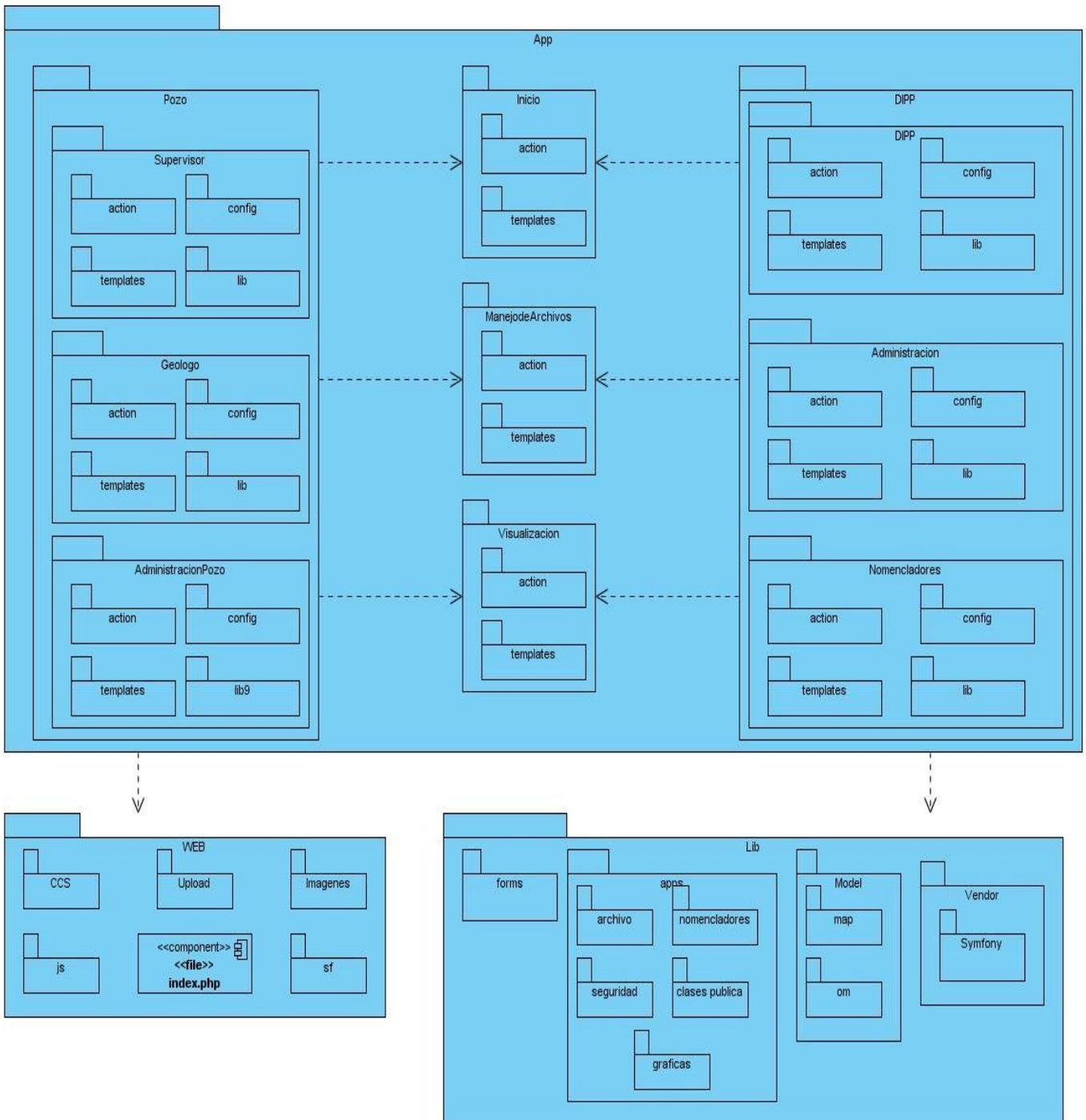


Figura 11: Diagrama Vista de Implementación.

3.3.2 Diagrama de componentes.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Se definen cinco estereotipos estándar que se aplican a los componentes: ejecutables que especifica un componente que se puede ejecutar en un nodo, librería que especifica una biblioteca de objetos estática o dinámica, tabla que especifica un componente que representa una tabla de una base de datos, archivo que especifica un componente que representa un documento que contiene código fuente o datos y documento que especifica un componente que representa un documento. (24)

Los componentes tienen 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.

3.3.2.1 Descripción general de los componentes.



Java Script: Es el fichero donde se implementan las validaciones del lado del cliente, además de las funciones para la creación de campos dinámicos.



CSS: En estos ficheros es donde le da formato al diseño de las interfaces. Los css son importantes para lograr una vista agradable y una uniformidad en la aplicación.



Action Class: Son las clases controladoras donde está contenida toda la lógica de la aplicación. Estas son las encargadas de interactuar con el modelo y a la vez actualizar las vistas (25).

CAPÍTULO 3: Presentación de la solución y validación.



Success: Son las vistas de la aplicación que están encargadas de interactuar con el usuario, en los Success se muestra el resultado lógico de la aplicación de una forma comprensible por usuario (25).



Model Class: Son las clases generadas por el propel de Symfony. Estas contienen la lógica de negocio y el acceso a los datos (Peer), lo cual posibilita que el trabajo de consultas a la base de datos se haga de forma mucho más rápida y sencilla.



Propel: Symfony utiliza PROPEL como ORM (Object-Relational-Mapping) y PROPEL utiliza Creole como capa de abstracción de Base de Datos (25).



Security Class: Son las clases que utiliza symfony para restringir el acceso a una acción determinada. En esta clase se especifican los requerimientos de seguridad que el usuario debe cumplir para acceder a una acción o para todas.



View: En este fichero se establece la estructura de la vista por defecto, el nombre del layout, el título de la página, las hojas de estilo y los archivos Java Script que se incluyen.



Layout: Contiene los elementos que son comunes para toda la aplicación o para parte de ella como es la cabecera y pie de página, normalmente son globales en toda la aplicación o en un gran número de páginas.

3.3.2.2 Diagramas de Componentes del Subsistema Pozo.

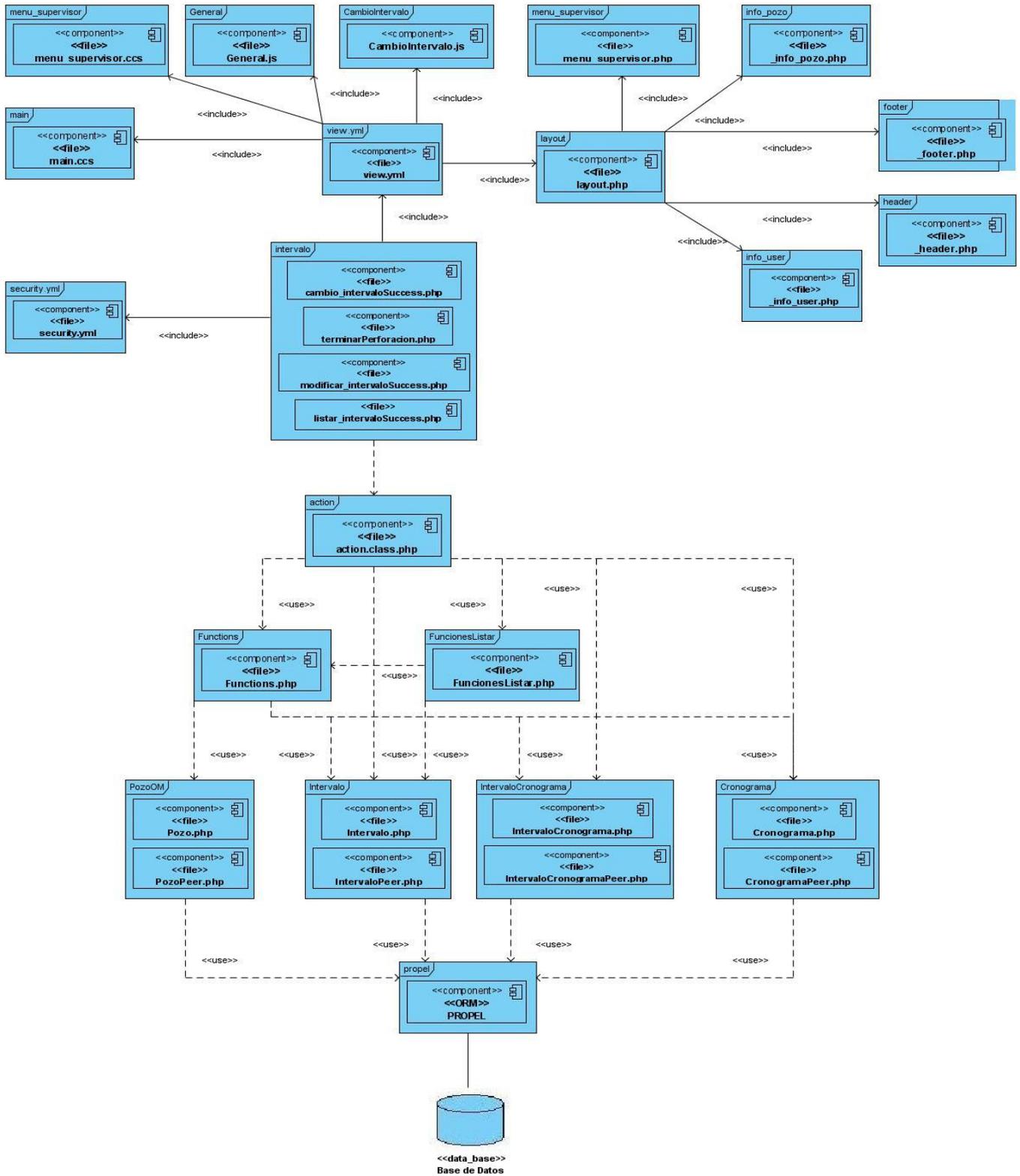


Figura 12: Diagrama de Componentes Gestionar Intervalo.

CAPÍTULO 3: Presentación de la solución y validación.

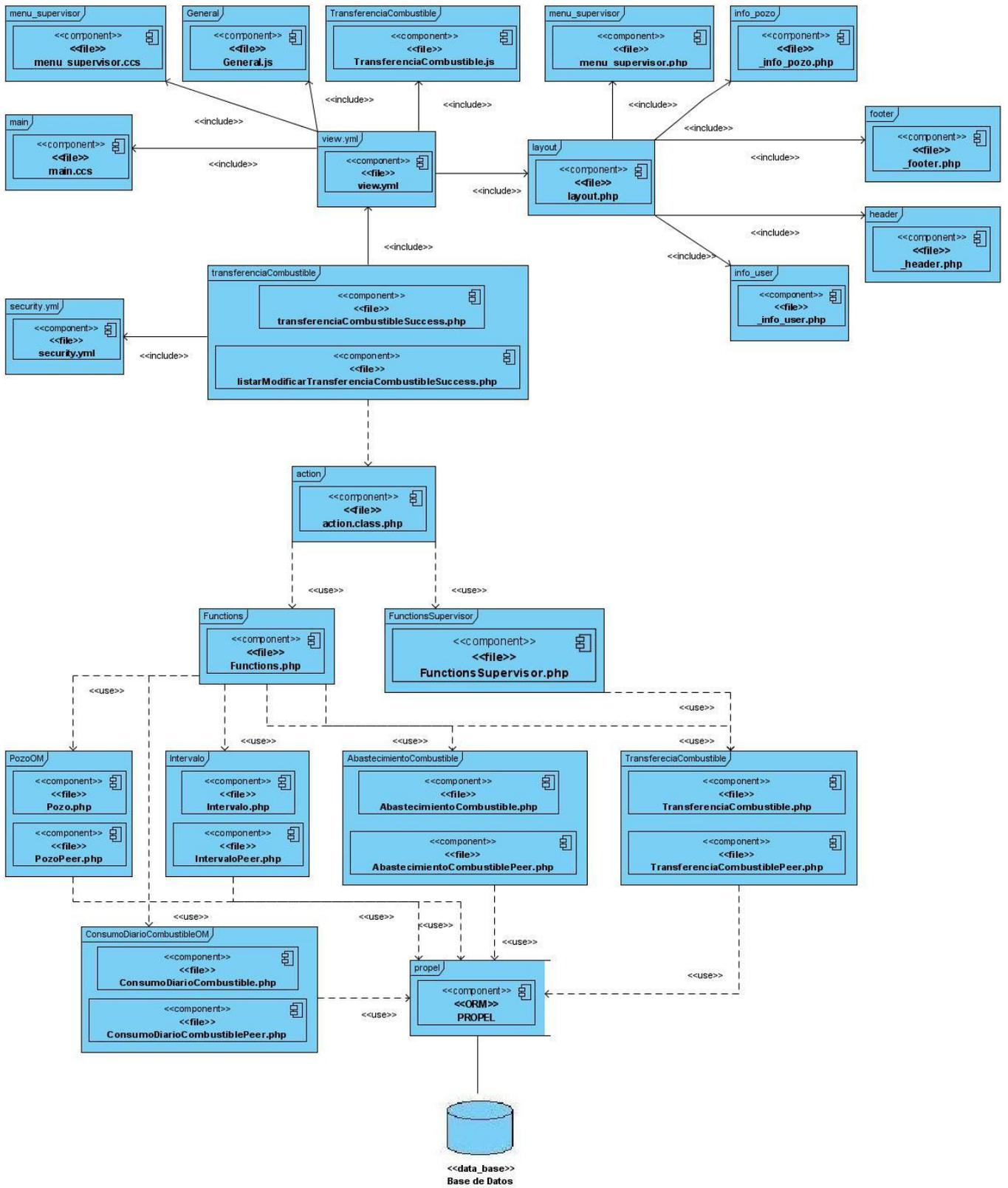


Figura 13: Diagrama de Componentes Gestionar Transferencia de Combustible.

CAPÍTULO 3: Presentación de la solución y validación.

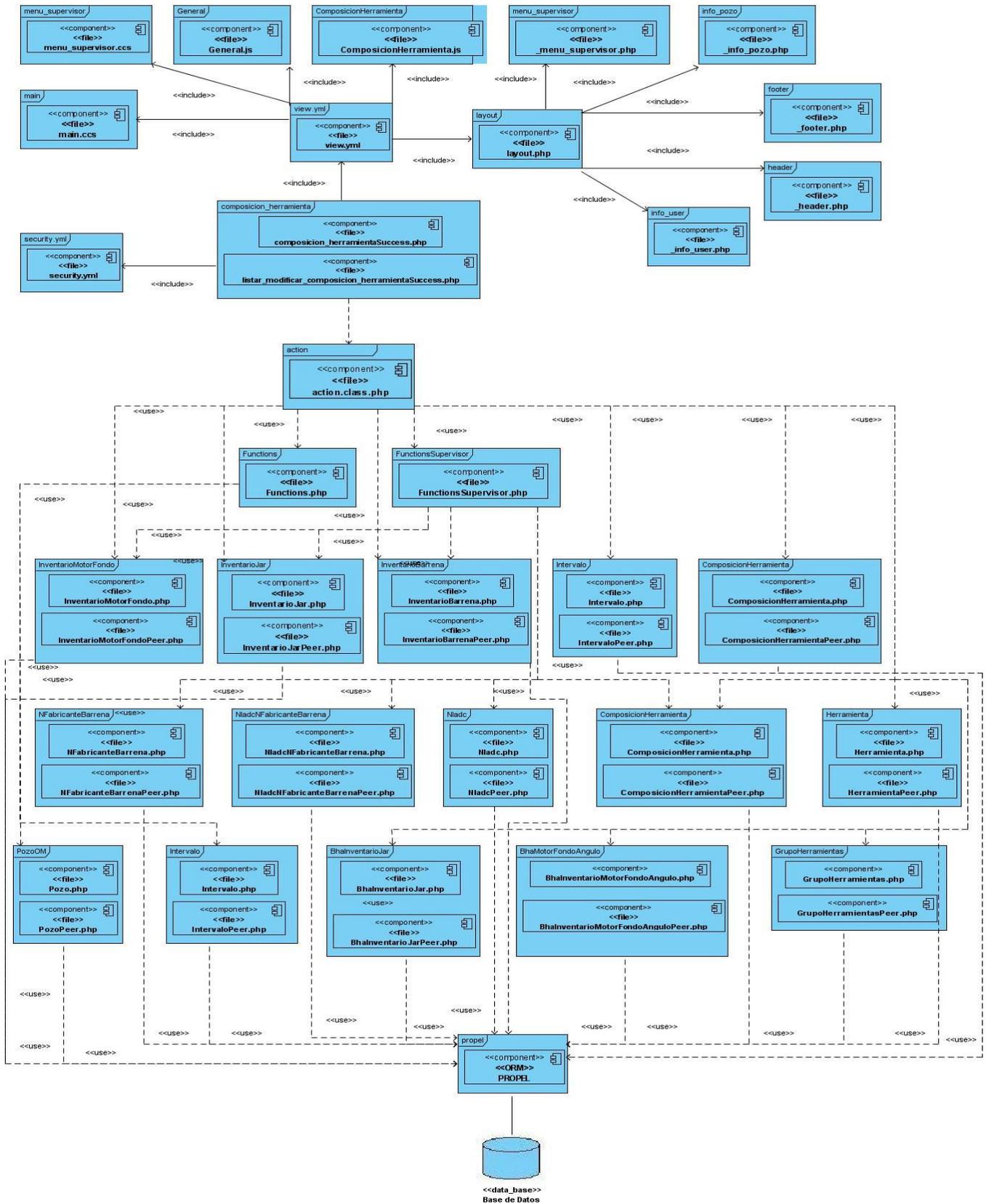


Figura 14: Diagrama de Componentes Gestionar Composición de Herramientas.

CAPÍTULO 3: Presentación de la solución y validación.

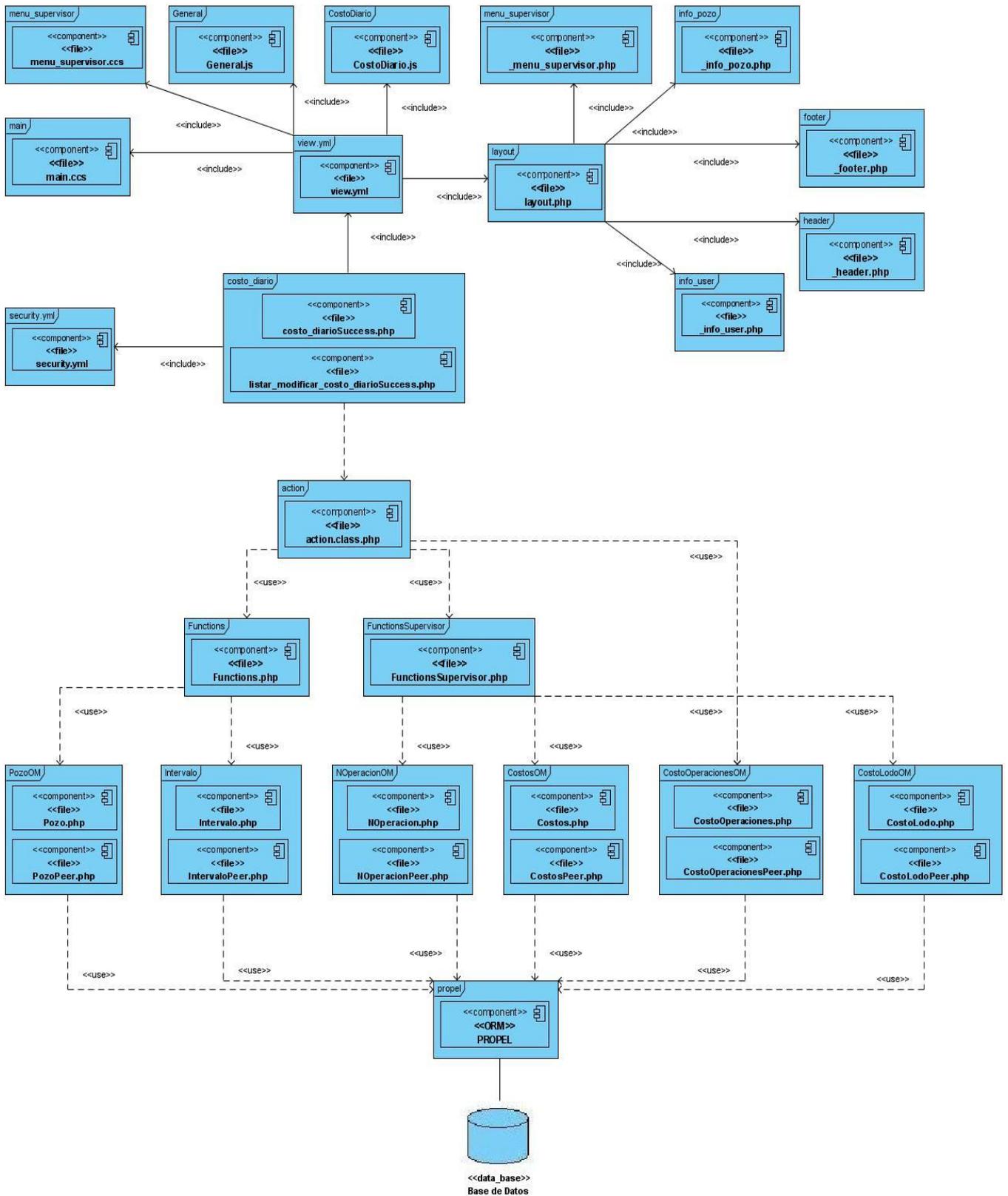


Figura 15: Diagrama de Componentes Gestionar Costos Diarios.

CAPÍTULO 3: Presentación de la solución y validación.

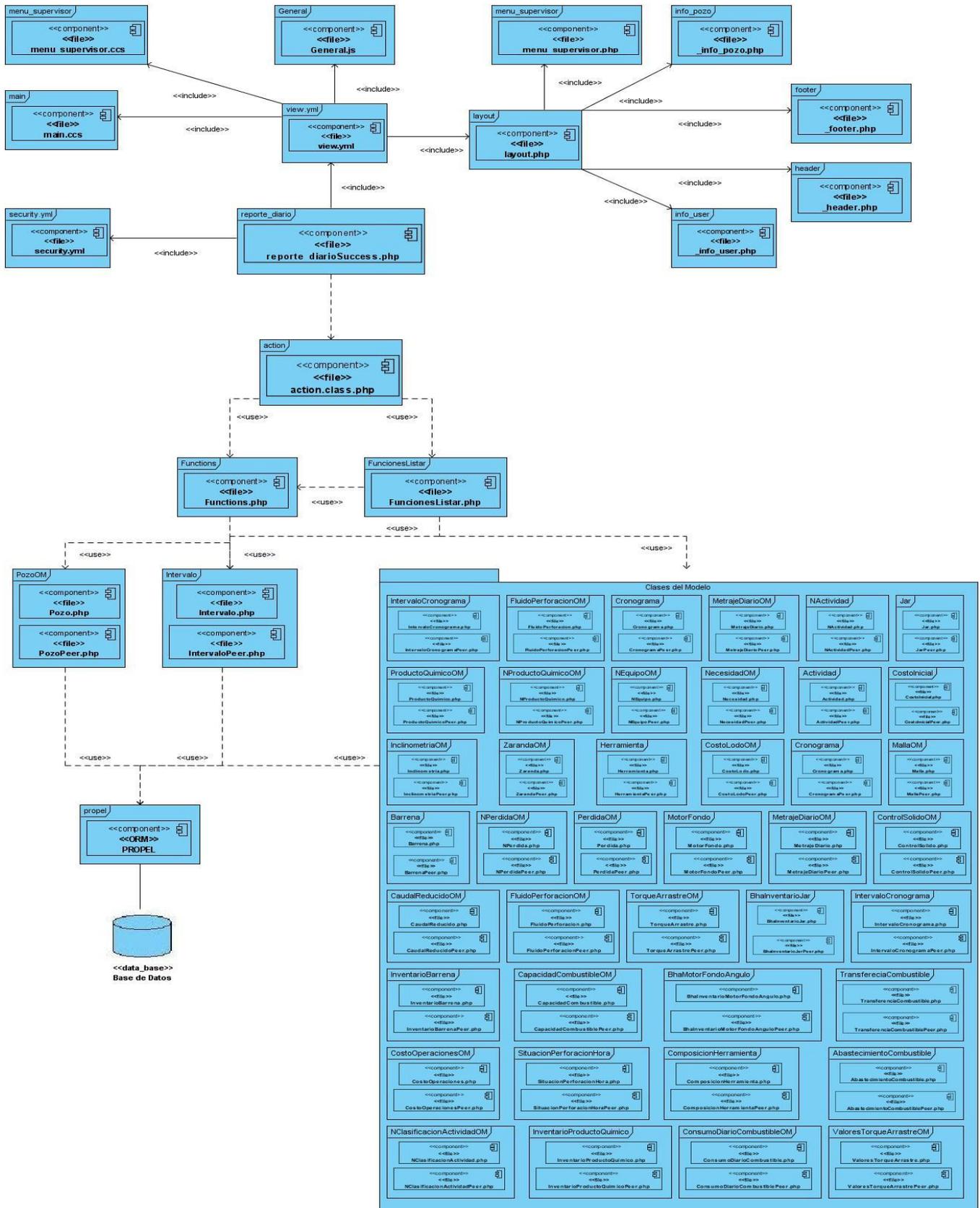


Figura 16: Diagrama de Componentes Generar Reporte Diario de Perforación.

CAPÍTULO 3: Presentación de la solución y validación.

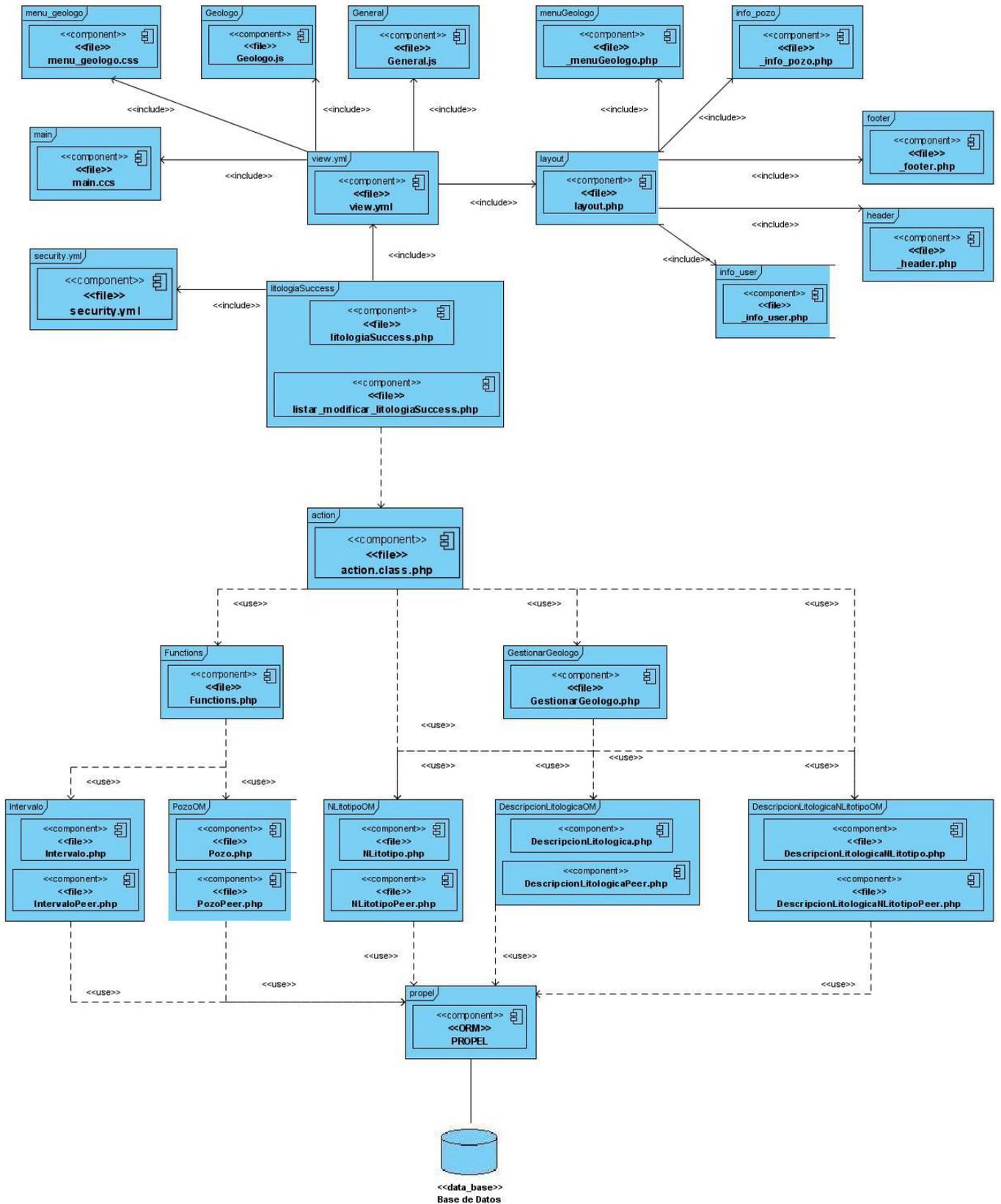


Figura 17: Diagrama de Componentes Gestionar Descripción Litológica.

CAPÍTULO 3: Presentación de la solución y validación.

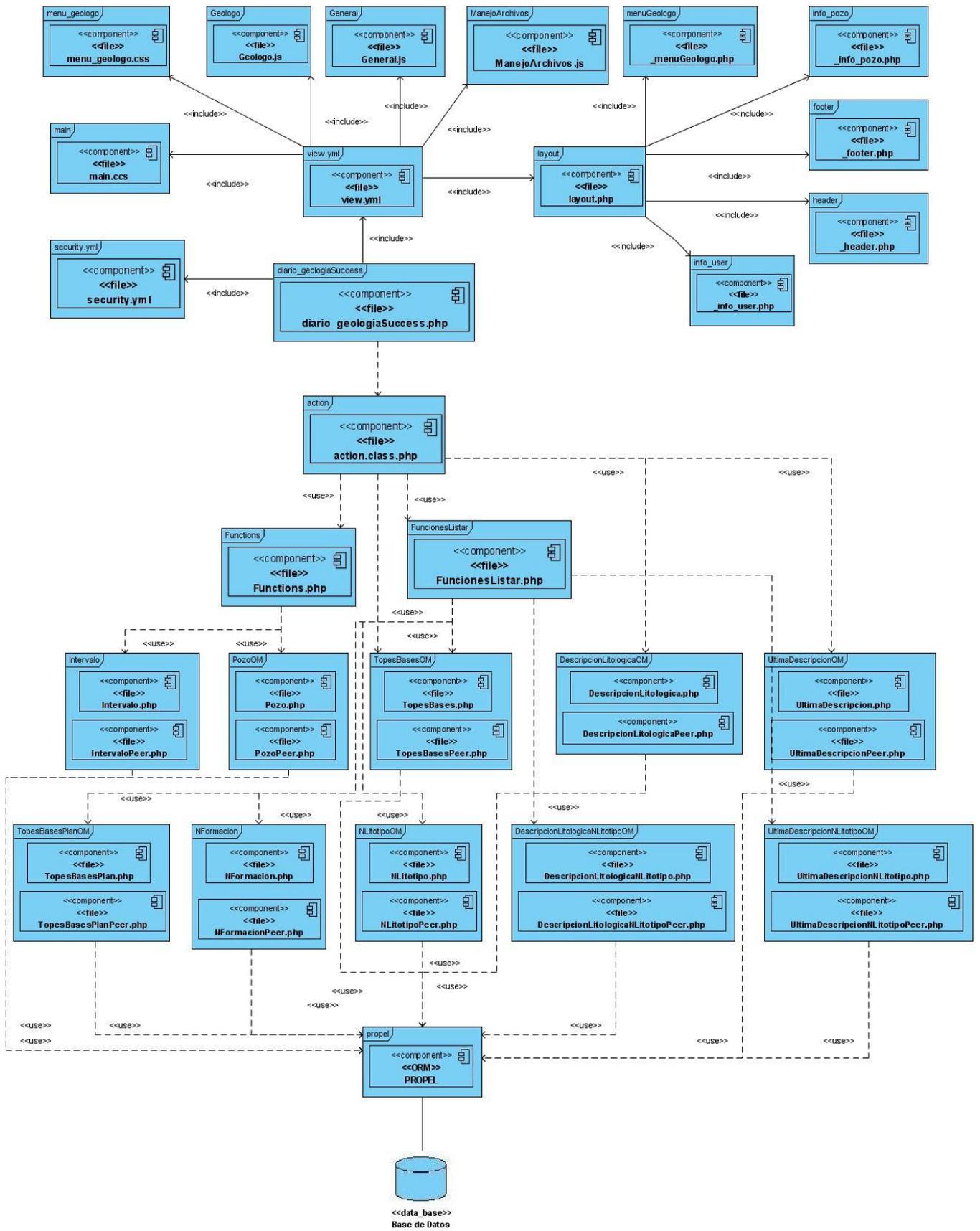


Figura 18: Diagrama de Componentes Generar Reporte Diario de Geología.

3.4 Elementos arquitectónicos.

3.4.1 Patrón Arquitectónico.

En el desarrollo de la solución propuesta se utilizó el patrón Modelo-Vista-Controlador (MVC), el cual separa la aplicación en tres partes: la vista, el modelo y el controlador. La vista es quien muestra la información del modelo al usuario. El controlador es quien recibe las solicitudes del usuario, encargándose de comunicar la vista con el modelo y el modelo es el que contiene el acceso a los datos y las reglas del negocio (26).

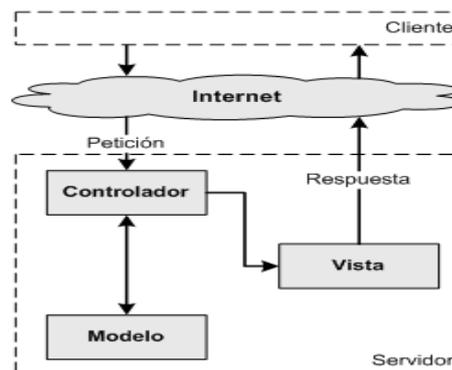


Figura 19: Patrón Arquitectónico MVC.

3.4.2 Patrones de Diseño.

Experto: Es un principio básico que suele utilizarse en el diseño orientado a objetos. Este propone la asignación de responsabilidad a la clase que cuenta con la información necesaria para crear una instancia o implementar un método (27).

Controlador: Asigna la responsabilidad del manejo de mensajes de los eventos de un sistema a una clase. Esta será la encargada de recibir y controlar el intercambio de información entre el usuario y el sistema, proporcionando la existencia de un solo punto de contacto entre estos. Lo que sugiere que la lógica de negocio debe estar separada de la capa de presentación (27).

Alta Cohesión: Es asignar una responsabilidad a una clase de modo que la cohesión siga siendo alta. Colaborando con otras clases para llevar a cabo las tareas, evitando precisamente los problemas causados por la baja cohesión (27).

Bajo Acoplamiento: Es la asignación de una responsabilidad a una o varias clases de manera que se mantenga un bajo acoplamiento entre estas, es la medida en una clase no dependa de muchas otras para realizar una responsabilidad, lo que permite reducir el impacto de los cambios y favorecer la reutilización (27).

CAPÍTULO 3: Presentación de la solución y validación.

Para un mayor conocimiento de la implementación del subsistema se presenta una tabla que muestra las principales clases y los patrones de diseño que en ellas se evidencian.

Tabla 1: Relación entre las principales clases y los patrones de diseño.

Experto en Información	Modelo:- PozoPeer		
Controlador		SupervisorActions	GeologoActions
Alta Cohesión	Modelo:- PozoPeer	SupervisorActions	GeologoActions
Bajo Acoplamiento	Modelo - PozoPeer	SupervisorActions	GeologoActions

La utilización de estos patrones favoreció el diseño clases realizado para desarrollar las funcionalidades del subsistema. A continuación se presentan las principales ventajas que esto proporciono:

- En las clases del modelo generadas por la ORM (Propel) se manifiestan los patrones: experto, alta cohesión y bajo acoplamiento. Esto permitió reutilizar partes funcionales del sistema. Favoreció la creación de código más fácil de entender y mantener. La baja dependencias entre las clases permitió que las afectaciones de una de ellas no impactara el resto las clases; lo cual fue importante debido a los altos cambios a los que fue sometido el modelo de base de dato durante la implementación.
- En las clases Actions de los dos módulos implementados se manifiestan los patrones: controlador, alta cohesión y bajo acoplamiento. Separar la lógica del negocio de la presentación. Tener un punto de intercambio entras las solicitudes realizadas por el cliente y la gestión de la lógica del negocio. Delegar responsabilidades a otras clases que contengan la información necesaria para realizar una operación. Por ejemplo el Actions del Módulo Supervisor delega la responsabilidad a la clase FunctionsListar.php de confeccionar los reportes de la perforación.

3.5 Pruebas del Subsistema Pozo.

Las pruebas de software, son los procesos que permiten verificar y revelar la calidad de un producto. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad de una aplicación. Básicamente es una fase en el desarrollo de software consistente en probar las aplicaciones construidas. Las pruebas de software se integran dentro de las diferentes fases del ciclo del software dentro de la Ingeniería de Software. Así se ejecuta un programa y mediante técnicas experimentales se trata de descubrir que errores tiene. Para determinar el nivel de calidad se deben efectuar unas medidas o pruebas que permitan comprobar el grado de cumplimiento respecto de las especificaciones iniciales del sistema.

CAPÍTULO 3: Presentación de la solución y validación.

A continuación se presentan las pruebas que se pusieron en práctica para aprobar el correcto funcionamiento del Subsistema Pozo en su versión 2.0. Las cuales fueron las pruebas de caja negra realizadas por el CEDIN en nuestra universidad y las pruebas de aceptación por parte del cliente en la zona de Varadero, Matanzas donde se realizó el despliegue de la aplicación.

3.5.1 Pruebas de Caja negra.

Las pruebas de caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software. O sea, la prueba de caja negra permite al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. La prueba de caja negra no es una alternativa a las técnicas de prueba de caja blanca. Más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que los métodos de caja blanca. La prueba de caja negra intenta encontrar errores de las siguientes categorías: funciones incorrectas o ausentes, errores de interfaz, errores en estructuras de datos o en accesos a bases de datos externas, errores de rendimiento y errores de inicialización y de terminación.

Mediante las técnicas de prueba de caja negra se obtiene un conjunto de casos de prueba que satisfacen los siguientes criterios: casos de prueba que reducen, en un coeficiente que es mayor que uno, el número de casos de prueba adicionales que se deben diseñar para alcanzar una prueba razonable y casos de prueba que nos dicen algo sobre la presencia o ausencia de clases de errores en lugar de errores asociados solamente con la prueba que estamos realizando (28).

3.5.2 Pruebas de Aceptación.

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, pues sería impresentable al cliente; sino que se realizan sobre el producto terminado e integrado o pudiera ser una versión del producto o una iteración funcional pactada previamente con el cliente. La experiencia muestra que aún después del más cuidadoso proceso de pruebas por parte del desarrollador, quedan una serie de errores que sólo aparecen cuando el cliente comienza a usarlo.

Sea como sea, el cliente siempre tiene razón. Decir que los requisitos no estaban claros, o que el manual es ambiguo puede salvar la cara; pero ciertamente no deja satisfecho al cliente. Una prueba de aceptación puede ir desde un informal caso de prueba hasta la ejecución sistemática de una serie de pruebas bien planificadas. De hecho, las pruebas de aceptación pueden tener lugar a lo largo de semanas o meses, descubriendo así errores latentes o escondidos que pueden ir degradando el

CAPÍTULO 3: Presentación de la solución y validación.

funcionamiento del sistema. Estas pruebas son muy importantes, ya que definen el paso nuevas fases del proyecto como el despliegue y mantenimiento (29).

Se emplean dos técnicas para las pruebas de aceptación:

La 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. Las pruebas alfa se llevan a cabo en un entorno controlado. Para que tengan validez, se debe primero crear un ambiente con las mismas condiciones que se encontrarán en las instalaciones del cliente. Una vez logrado esto, se procede a realizar las pruebas y a documentar los resultados (29).

La 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. Como resultado de los problemas informados durante la prueba beta, el desarrollador del software lleva a cabo modificaciones y así prepara una versión del producto de software para toda la clase de clientes (29).

Durante el proceso de desarrollo del subsistema, se le realizaron pruebas pilotos por parte de los miembros del equipo de desarrollo. La última de estas fue realizada durante el despliegue de un pozo en perforación en Matanzas, específicamente en la zona de Varadero, las cuales permitieron comprobar el correcto funcionamiento de la aplicación bajo circunstancias reales. Esto permitió que los supervisores y geólogos de pozo pudieran interactuar con él validando su correcto funcionamiento. Gracias a estas pruebas se logró una versión que cumple con todos los requisitos pactados, haciéndose constancia de ello con la firma de aceptación del cliente.

3.5.3 Descripción de los Casos de Pruebas.

Los casos de prueba son un conjunto de entradas de pruebas, condiciones de ejecución y resultados esperados desarrollados para verificar una función determinada. El principal objetivo del proceso de prueba es validar el correcto funcionamiento del sistema, aumentando su calidad con la detección de errores o funcionamientos incorrectos. Las pruebas aplicadas al subsistema fueron desarrolladas por el grupo de pruebas de Centro de Desarrollo de Informático Industrial, las cuales fueron aplicadas en tres iteraciones donde se iban probando todos los CU del subsistema y realizando reportes de las no conformidades encontradas por el grupo de prueba, en la primera iteración fueron detectadas once no conformidades, las cuales fueron entregadas en un documento y resueltas. Así se fue realizando el

CAPÍTULO 3: Presentación de la solución y validación.

proceso de pruebas hasta la tercera iteración donde se logro obtener un subsistema con la calidad esperada del producto.

Tabla 2: CP Gestionar Transferencia de Combustible. Crear Transferencia (30).

ID del escenario	Escenario	Variable 1 Existencia: Equipo, campamento		Variable 2 Transferencia Equipo, campamento		Respuesta del sistema	Resultado de la prueba
EC1.1	Introducir correctamente los datos en los respectivos campos.	50206	16601	50204	16600	Los datos se han insertado correctamente	satisfactorio
EC1.2	Introducir letras en los datos de los respectivos campos.	2	1	7	a	Debe entrar solo números	satisfactorio
EC1.3	Dejar campos vacíos en el formulario que se muestra.	2	1	7		Debe llenar el campo "transferencia del campamento"	satisfactorio
		2	1		5	Debe llenar el campo "transferencia del equipo"	satisfactorio
EC1.4	Introducir correctamente los datos y no confirmar la operación.	2	1	1	0	Permanece en la misma interfaz	satisfactorio
EC1.5	Introducir en el campo Transferencia Equipo un valor mayor que el de Existencia Equipo.	1	1	3	1	El campo Transferencia Equipo debe ser menor que la Existencia Equipo	satisfactorio

Tabla 3: CP para sección Crear Descripción Litológica (30).

ID del escenario	Escenario	Variable 1 Profundidad Inicial	Variable 2 Profundidad final	Variable 3 Litotipos	Variable 4 %inicial	Variable 5 %final	Variable 6 Descripción	Respuesta del sistema	Resultado de la prueba
------------------	-----------	-----------------------------------	---------------------------------	-------------------------	------------------------	----------------------	---------------------------	-----------------------	------------------------

CAPÍTULO 3: Presentación de la solución y validación.

1.1	Crear Descripción Litológica correctamente.	V 3	V 5	V Anhidrita	V 100	V 100	V OK	Muestra un mensaje de éxito de la operación: "Los datos se han insertado correctamente." Y mantiene la interfaz principal del caso de uso.	<i>satisfactorio</i>
1.2	Crear Descripción Litológica, dejando campos vacíos (Profundidad Inicial, Profundidad Final y descripción).	I Vacio	I Vacio	V Anhidrita	I 100	I 100	I Vacio	El Sistema muestra una ventana emergente indicando el error: "No puede dejar campos vacíos". y muestra nuevamente el formulario para poder insertar los datos	<i>satisfactorio</i>
1.3	Crear Descripción Litológica, introduciendo caracteres extraños (Profundidad Inicial, Profundidad Final).	I M()^	I ++	V Anhidrita	I 100	I 100	V comentar	El Sistema muestra una ventana emergente indicando el error: "Debe entrar solo números".	<i>satisfactorio</i>

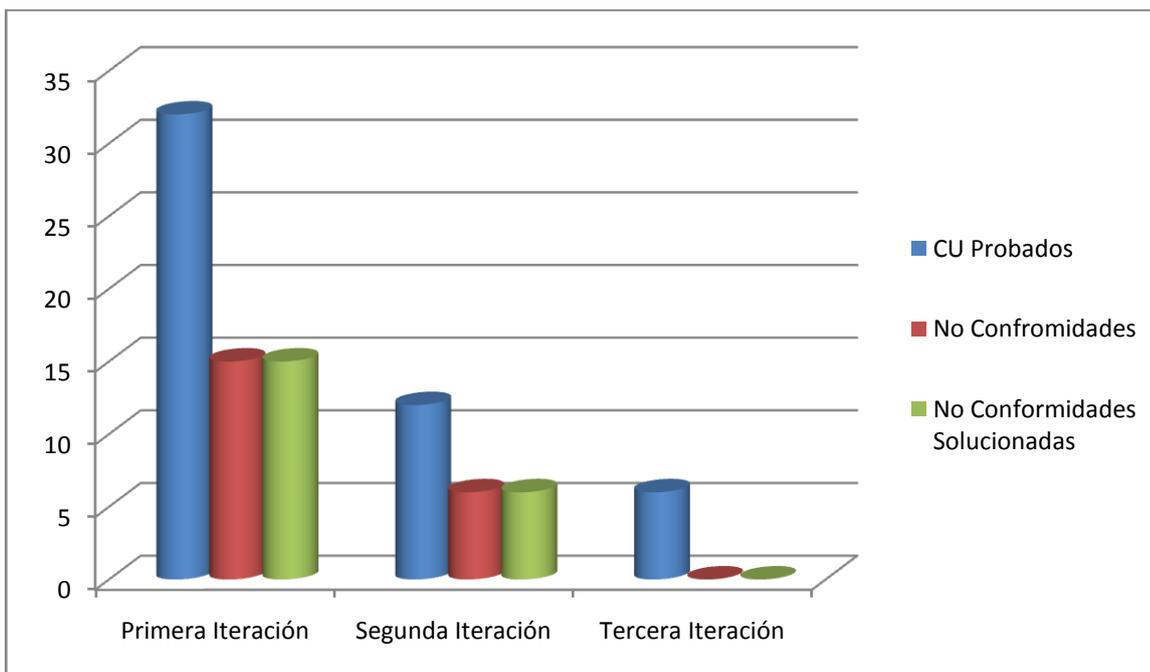
CAPÍTULO 3: Presentación de la solución y validación.

1.4	Agregar una fila para datos de litotipo	N/P	N/P	N/P	N/P	N/P	N/P	El sistema inserta una nueva fila, vacía, para introducir datos de Litotipo, % Inicial, % Final, Descripción.	<i>satisfactorio</i>
1.5	Crear Descripción Litológica, sin que los valores de % sumen 100.	V 3	V 5	V Anhidrita	V 10	V 10	V OK	El Sistema muestra una ventana emergente con cartel "La suma de los por cientos iniciales (o finales) tiene que dar 100" y muestra nuevamente el formulario para poder insertar los valores correctamente.	<i>satisfactorio</i>
				Diabasa	30	30			
1.6	Crear Descripción Litológica, introduciendo el mismo valor en el campo Litotipo.	V 3	V 15	V Diabasa	V 10	V 10	V OK	El sistema muestra un mensaje "No puede repetir los mismo litotipos".	<i>satisfactorio</i>
				Diabasa	90	90			

CAPÍTULO 3: Presentación de la solución y validación.

1.7	Crear Descripción Litológica, No confirmar operación	V 3	V 5	V Anhidrita	V 100	V 100	V OK	El Sistema no envía los datos por tanto no se guardan correctamente e los datos entrados manteniéndose en el mismo formulario.	<i>satisfactorio</i>
-----	------------------------------------------------------	--------	--------	----------------	----------	----------	---------	--------------------------------------------------------------------------------------------------------------------------------	----------------------

Se realizaron las pruebas de caja negra que brindan al cliente conformidad y seguridad ante las funcionalidades del subsistema. Se realizó un caso de prueba por cada caso de uso arrojando los resultados que se presentan en la siguiente gráfica:



Gráfica 1: Resultado del proceso de pruebas.

3.6 Conclusiones Parciales.

En este capítulo se expuso la solución propuesta, presentando los requisitos funcionales así como los diagramas de componentes de los CU más significativos del subsistema, y de esta forma se arriban a las siguientes conclusiones:

- El subsistema implementado soluciona el problema planteado en la investigación.

CAPÍTULO 3: Presentación de la solución y validación.

- La realización de las pruebas de aceptación permitió validar los requisitos pactados con el cliente.
- Las pruebas de caja negra realizadas a la solución, permitieron aumentar la calidad del producto final.

Conclusiones Generales

CONCLUSIONES GENERALES

A término del desarrollo de la investigación se logró cumplir el objetivo propuesto; la implementación del Subsistema Pozo en su versión 2.0 del Sistema para el Manejo Integral de la perforación de Pozos de Petróleo y Gas, concluyendo lo siguiente:

- El estudio de los conceptos asociados al dominio del problema permitió que el desarrollo del subsistema cumpliera con las reglas y restricciones del negocio.
- El uso de herramientas, tecnologías libres y multiplataforma permitió construir una solución con grandes capacidades de portabilidad.
- La realización de las pruebas de caja negra al subsistema se pudo validar la solución que se propone en este trabajo de diploma, asegurando que el mismo cumple con los requerimientos establecidos.
- Al aplicar las pruebas de aceptación permitió validar el cumplimiento de los requisitos pactados con el cliente.

RECOMENDACIONES

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

- Implementar las funcionalidades asociadas a los procesos de prueba, encamisado y cementación, y terminación y ensayo.
- Utilizar un framework para el desarrollo de las interfaces.
- Optimizar el código del subsistema.

REFERENCIAS BIBLIOGRÁFICAS

1. **bsiGroup**. bsigroup.com. [Online] 2011. [Cited: 06 07, 2011.] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.
2. **Cuevas, David Tavares**. BibliotecaUci. [Online] 2009. [Cited: Octubre 22, 2010.] http://bibliodoc.uci.cu/TD/TD_2728_09.pdf.
3. **SIPP**. *Glosario de Términos*. 2009.
4. **Paneque, Jorge Roberto Allen**. *Módulo Pozo del Sistema de Manejo Integral de Perforación de Pozos*. Universidad de las Ciencias Informáticas. 2010. Tesis de grado.
5. Peloton. *Drilling & Well Data Software Solutions*. [Online] [Cited: enero 20, 2011.] <http://www.peloton.com/es/>.
6. Infoil. [Online] [Cited: enero 14, 2011.] http://www.infoil.com.ar/solutions_perforation.html.
7. **Halliburton**. *Software WELLSIGHTMR v4.5*. 2007.
8. Lenguajes de Programación. [Online] [Cited: diciembre 6, 2010.] <http://www.lenguajes-de-programacion.com/programacion-web.shtml>.
9. pisuerga. *Introducción a la Programación Orientada a Objeto*. [Online] [Cited: Diciembre 11, 2010.] http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_1.htm.
10. Symfony. *HowToContributeToSymfony*. [Online] [Cited: enero 17, 2011.] <http://trac.symfony-project.org/wiki/HowToContributeToSymfony#CodingStandards>.
11. **Gutiérrez, Javier J**. [Online] [Cited: enero 20, 2011.] http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
12. **Mehdi Achour, Friedhelm Betz, Antony Dovgal**. PHP. [Online] 2001. [Cited: 01 20, 2011.] <http://php.net/manual/es/index.php>.
13. Software libre para todos Liberado PHP 5.0.0 . *Software Libre* . [Online] [Cited: marzo 05, 2011.] <http://www.somoslibres.org/modules.php?name=News&file=article&sid=178>.
14. Hooping. [Online] [Cited: marzo 7, 2011.] <http://www.hooping.net/faq-caracteristicas.aspx>.
15. Universidad de la Rioja. [Online] [Cited: marzo 6, 2011.] <https://belenus.unirioja.es/~guprado/pagweb/caraccss.html>.
16. Javascríp. [Online] [Cited: marzo 5, 2011.] <http://javascriptexperts.blogspot.com/2009/05/javascript-objetos-dinamicos-closures.html>.
17. desarrolloweb.com. [Online] [Cited: marzo 07, 2011.] <http://www.desarrolloweb.com/articulos/1178.php>.
18. **Fabien Potencier, Francois Zaninotto**. *Symfony la guía definitiva*. 2007.

Referencias Bibliográficas

19. **Aldo Cristiá Alvarez, Yeni Morgado Sánchez, Ing. Marcos Luis Ortíz Valmaseda, Ing. Yudisney Vazquez Ortíz.** [Online] junio 2010. [Cited: marzo 20, 2011.] <http://postgresql.uci.cu/system/files/Gu%C3%ADa%20para%20la%20personalizaci%C3%B3n%20de%20PostgreSQL%208.4.pdf>.
20. Scribd. [Online] [Cited: marzo 23, 2011.] <http://es.scribd.com/doc/52208534/29/CARACTERISTICAS-Y-VENTAJAS-DEL-APACHE>.
21. **Hernández, Cinolkis Cobas.** *Desarrollo del Componente de Seguridad para el subsistema Mondrian perteneciente al sistema Pentaho.* Universidad de las Ciencias Informáticas. La Habana : s.n., 2010. Tesis de grado.
22. **Fontela, Carlos.** Técnicas de aseguramiento de la calidad del producto. CyS Ingeniería de Software. [Online] junio 19, 2009. [Cited: marzo 24, 2011.] <http://cysingsoft.wordpress.com/2008/06/24/tecnicas-de-aseguramiento-de-la-calidad-del-producto>.
23. El modelo dinámico y de implementación. [Online] [Cited: 5 5, 2011.] <http://www.elai.upm.es:8009/spain/Asignaturas/InfoInd/apuntesAOOD/cap5UMLDinamicoImpl.pdf>.
24. Modelo de Implementación: Diagramas de Componentes y Despliegue. [Online] [Cited: mayo 8, 2011.] <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>.
25. **Fabien Potencier, Francois Zaninotto.** *Symfony la guía definitiva.* 2010.
26. Patrón "Modelo-Vista-Controlador". [Online] [Cited: Mayo 27, 2011.] <http://www.proactiva-calidad.com/java/patrones/mvc.html>.
27. **Craig Larman.** *UML y Patrones.* La Habana : Felix Varela, 2004.
28. **Pressman, Roger S.** *Ingeniería de Software un enfoque practico.*
29. Universidad de la República - Uruguay. [Online] [Cited: mayo 3, 2011.] <http://www.fing.edu.uy/~dvallesp/Tesis/webActividades/todo/actividades/VyV/testing/tda.htm>.
30. **SIPP.** *Diseño de Casos de Prueba del Subsistema Pozo.* 2011.

ANEXOS

Anexo 1: SIPP

Usted está en el sistema como: **Nombre: Olimac** **Usuario: kmilo** [Cerrar Sesión](#)

Nombre	Operador	Equipo	Yacimiento	Formación actual	Interv. Perforación	Pronóstico 24hrs.	Fecha	Día actual de perf.	Profundidad actual	Operación actual
VDW-1001	CUPET	CUPET 4	Varadero Oeste	Cojimar	Intermedio	Realizar LOT y Bajar con nueva herramienta para continuar perforando.	2010-04-05	10	487	Perforando

Información Geológica

Topes y bases de formaciones y marcadores:

Formación	Profundidad por Muestras
Cojimar	

Sistema de Manejo Integral de Perforación de Pozos Petroleros (SIPP)
Desarrollado por la Universidad de las Ciencias Informáticas 2010

Anexo 2: MasterView

MasterView™
integrated well lifecycle™

Adquisición de Datos, Análisis y Visualización

- Pronostico Geológico
- Planificación de Pozos
- Permisología Regulatoria
- Cronograma de Taladro y Proyectos de Pozo
- Construcción de la locación
- Operaciones de Perforación
- Operaciones Geológicas
- Completaciones
- Instalaciones, construcción y gasoductos
- Intervenciones y operaciones del pozo
- Recuperación y remediación

WellView™

RigView™

SiteView™

Anexo 3: WellView

WellView

Quickly find wells using query templates.

Group your wells by criteria that you specify.

Customize the well list.

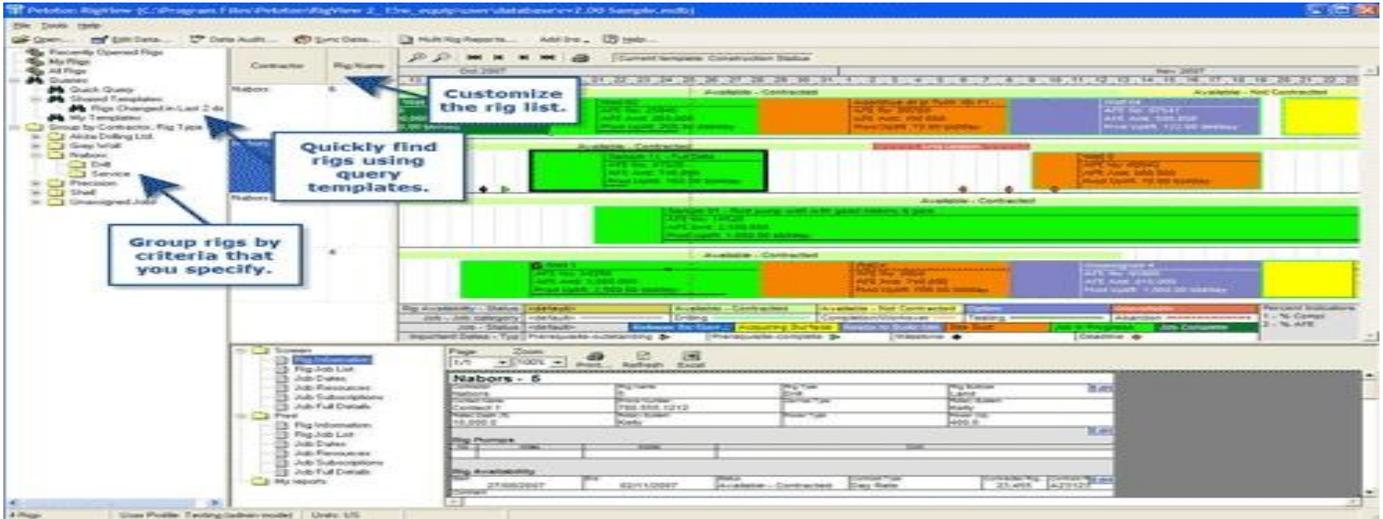
Well Name	APU/UW	License #	Ground Elevation	Original Fill Elevation
Sample 13 - Phase and Prod	100/04-36-050-21w/5/00	1234568	1,026.10	1,030.30
Sample 14 - Phase and Prod	100/02-02-050-20w/5/00	1234570		80
Sample 15 - Phase and Prod	100/05-36-050-21w/5/00	1234572		70
Sample 16 - Phase and Prod	100/01-36-050-21w/5/00	1234573		70
Sample 17 - Phase and Prod	100/15-03-052-21w/5/00	1234574		70
Sample 18 - Phase and Prod	100/07-12-011-20w/5/00	1234575		70
Sample 21 - Prod	100/01-27-051-21w/5/00	1234577	990.40	1,002.00
Sample 22 - Prod	100/02-08-051-22w/5/00	1234576	1,044.20	1,049.60
Sample 23 - Prod	100/04-36-051-21w/5/00	1234571	990.00	991.00
Sample 24 - Prod	100/06-22-051-20w/5/00	1234566	1,047.10	1,051.40
Sample 25 - Prod	100/04-35-055-04w/5/00	1234569	1,205.90	1,311.20
Sample 26 - Prod	100/02-16-051-20w/5/00	1234568	1,079.50	1,082.20
Sample 39 - Proposal vs Actual	100/05-37-067-21w/5/00	1234623	1,104.00	1,109.00

13 Well(s)

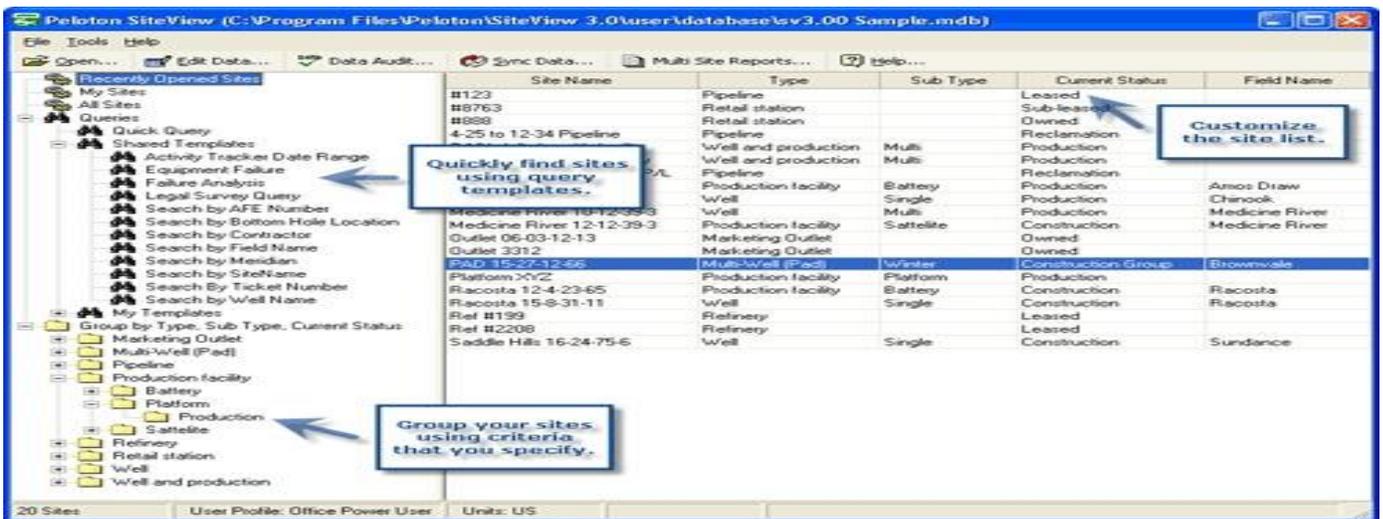
Profile: All Data Units: Metric Reference Datum: KB

Anexos

Anexo 4: RigView

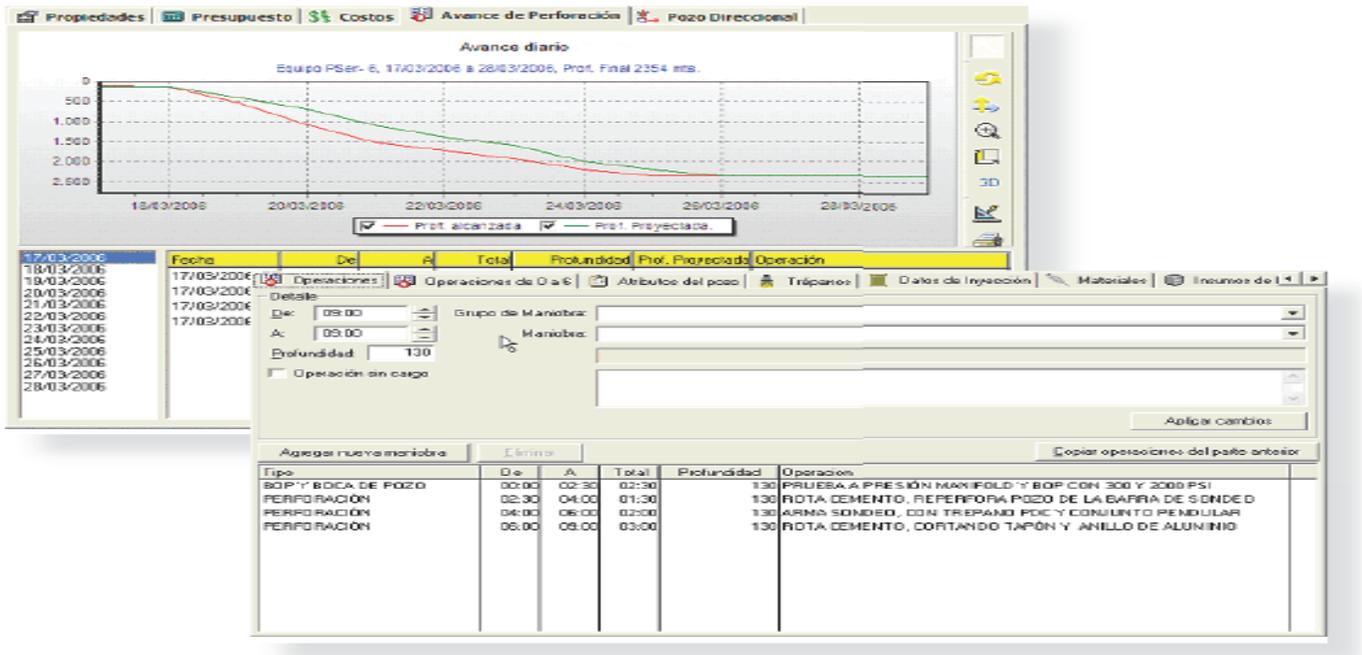


Anexo 5: SiteView



Anexos

Anexo 6: InfoPerf



Anexo 7: WellSight v4.5

The screenshot shows the 'WELLSIGHT 2000 Report Manager' window with the 'Report Selection' dialog box open. The dialog is titled 'Report Selection' and features the HALLIBURTON logo. It contains a list of reports to be selected, with 'Concentration Daily Report' currently selected. To the right, there are fields for 'Daily Reports', 'Recap Interval', and 'Summary Reports', each with 'From' and 'To' dropdown menus. The 'Daily Reports' and 'Summary Reports' fields are set to '001A', and the 'Recap Interval' field is set to '01'. At the bottom, there are buttons for 'Help', 'Cancel', '< Back', 'Next >', and 'Finish'.

GLOSARIO DE TERMINOS

A

B

- **Backreaming:** Es una operación que se realiza durante la perforación, en la cual se sube la tubería de perforación con rotación en el top drive y se repasa una sección del hoyo.
- **Bajando con y sin rotación:** Es una operación que se realiza durante la perforación, en la cual se baja la tubería de perforación con o sin rotación por una parte del hoyo que ya ha sido perforada. Normalmente se rota cuando se encuentra un punto de resistencia.
- **Baroid Fluid Services:** Es una empresa que presta servicios de fluidos para la perforación de pozos.
- **Bit Sub:** Herramienta de perforación que se coloca después de la barrena
- **Bugs:** Errores o mal funcionamientos que se encuentran en un software.

C

- **CEDIN:** Centro de Desarrollo Informático Industrial.
- **CEINPET:** Centro de Investigaciones de Petróleo.
- **CSS:** Hojas de estilo en cascada.
- **CU:** Caso de Uso.
- **CUPET:** Unión Cuba-Petróleo.

D

- **Deslizando:** Es la operación donde se orienta el pozo, solamente hay rotación en el motor de fondo, no así en el top drive.
- **DIPP:** Dirección de Intervención y Perforación de Pozos.

E

- **EPEPC:** Empresa de Producción y Extracción de Petróleo del Centro.
- **Estabilizadores:** Herramienta que se utiliza en la perforación la cual tiene como objetivo estabilizar el peso de la composición de la herramienta.

F

- **Framework:** Marco de trabajo, solución reutilizable y extensible.

Glosario de Términos

G

H

- **Harware:** Parte física de un computador y más ampliamente de cualquier dispositivo electrónico
IEEE: Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos).
- **HTML:** “HyperText Mark-Up Language”, que viene a ser, en nuestro idioma, “Lenguaje para el Formato de Documentos de Hipertexto”.
- **HTTP:** Protocolo de Transferencia de Hipertexto.

I

- **Interfaz:** Zona de contacto o conexión entre dos componentes de “harware”, entre dos aplicaciones, o entre un usuario y una aplicación. Apariencia externa de una aplicación informática.

J

K

L

- **Levantando con y sin rotación:** Es una operación que se realiza durante la perforación, en la cual se levanta la tubería de perforación con o sin rotación por una parte del hoyo que ya ha sido perforada. Normalmente se rota cuando se encuentra un punto de resistencia.

M

N

Ñ

O

- **ORM:** Mapeo Relacional de Objetos, es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

P

- **Perforando:** Es la operación principal durante el proceso de perforación. Siempre se realiza bajando y con rotación en el top drive.
- **Power Drive:** Herramienta que se utiliza para orientar la herramienta. Esta tiene la peculiaridad de que esta puede orientarse automáticamente a partir de los parámetros planificados.

Glosario de Términos

Q

R

S

- **SGBD:** Sistema gestor de Base de Datos.
- **SideStrack:** Acción que se realiza en el pozo cuando se considera que el hoyo perforado se perdió por complejidades técnicas o geológicas.
- **SIPP:** Sistema para el Manejo Integral de la Perforación de Pozos de Petróleo y Gas.

T

U

- **UCI:** Universidad de las Ciencias Informáticas.
- **URL:** Localizadores uniformes de recursos.

V

W

X

Y

Z