



Universidad de las Ciencias Informáticas

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

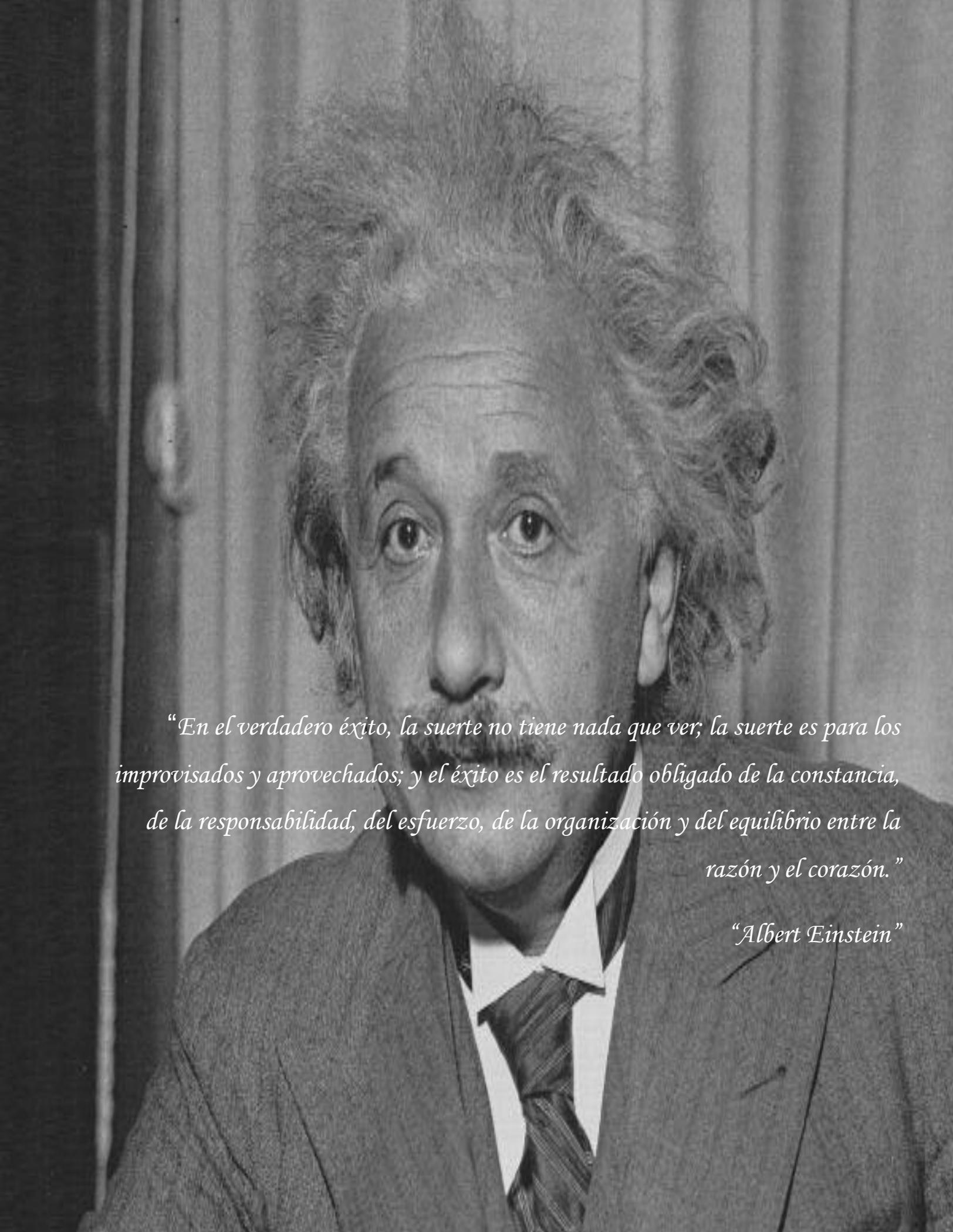
Título: Sistema de Gestión de Lechadas para la Cementación de Pozos
Petroleros.

Autores: Leandris De la Cruz Castillo.
Maikel Sánchez Gutiérrez.

Tutora: Ing. Dailién García Pérez

Ciudad de la Habana, Cuba

Junio, 2009

A black and white portrait of Albert Einstein, showing his characteristic wild, curly hair and mustache. He is wearing a dark suit jacket, a white shirt, and a striped tie. The background is a simple, light-colored wall with a door handle visible on the left.

“En el verdadero éxito, la suerte no tiene nada que ver; la suerte es para los improvisados y aprovechados; y el éxito es el resultado obligado de la constancia, de la responsabilidad, del esfuerzo, de la organización y del equilibrio entre la razón y el corazón.”

“Albert Einstein”

DEDICATORIA

De Leandris

A Mis Padres.

A mis Amigos.

A mi Novia.

En especial a mis abuelos Abigail y Reina.

De Maikel

A mis Padres, hermana y amigos.

AGRADECIMIENTOS GENERALES

El presente trabajo de diploma representa la culminación de un largo trayecto de estudios y sacrificios, la culminación de una etapa muy importante de nuestras vidas y el comienzo de otra etapa, por lo que agradecemos a todas aquellas personas que nos han acompañado durante estos cinco años de carrera y que han tenido que ver de una forma u otra a la contribución de que seamos las personas que ahora somos, a los que nos brindaron un consejo, a los que nos tendieron una mano en los momentos difíciles, a todos los que hicieron posible que usted hoy este leyendo estas palabras.

A esa persona tan especial que hace 7 años atrás tuvo la brillante idea de crear una universidad de futuro, dándonos la oportunidad de superarnos, a ti Fidel gracias por esta oportunidad.

A Raúl Castro que ha sabido mantener el legado de Fidel y hacer que esta revolución siga adelante.

A todos los profesores que a lo largo de la carrera compartieron sus conocimientos con todos nosotros.

A todos aquellos que de una forma u otra contribuyeron a la realización de esta tesis.

De Leandris

A los que hicieron que fuera formándome aquí en esta escuela, los que comenzaron conmigo, el antiguo 9101, que aunque nos separaron siempre hemos estado unidos, a mi maestra Yeleny Zulueta que fue la primera profesora guía que tuve en la UCI y como guía nos condujo por el camino del saber y del buen comportamiento.

A los que en algún momento me sacaron del mundo de los estudios para que pudiese “desconectar” y me invitaron a la recreación. A mis compañeros de apartamento durante estos cinco años.

A mis padres por su apoyo, su amor y cariño y por estar presentes cuando lo necesite, por ser una fuente de inspiración y un ejemplo a seguir para mí.

A mis abuelos Reina y Abigail que me dieron su educación desde pequeño e hicieron que fuera encaminado por el camino de ser una mejor persona y porque siempre confiaron en mí y en mis primos que espero que al igual que yo sean graduados universitarios en algún momento.

A mis abuelos Israel y Neida por su cariño, amor y por aguantarme mis malcriadeces en las vacaciones en su casa.

A mi hermano por estar siempre orgulloso de mí y por confiar en que pudiera ser la persona que soy ahora.

A mis tíos y tías Sonia, Ciria, Sucrenia, Rafael, Alejandro, Juan, Engris, Mario por darme su confianza, cariño y apoyo cuando lo necesité.

A mis primos Yolo, Yosvanis, Pacho, Lisnay, Malian por los buenos y malos momentos que hemos compartido durante toda la vida.

A mi mejor amigo desde la infancia Adrian, por acompañarme durante mis años desde primaria hasta el servicio militar apoyándome en todo lo necesario y dándome su confianza y amistad.

A todas las personas de mi barrio que confiaron en que yo podía llegar a ser quien soy en estos momentos.

Y no por último es menos importante, le agradezco con mucho mi alma a mi novia por el apoyo que me ha brindado durante este tiempo que hemos estado juntos. Por darme su amor y por confiar en que yo podía terminar mi carrera y por ser mi musa en momentos en que la inspiración me tomaba por sorpresa y mi corazón se llenaba de amor solo para hacerle unos poemas.

De Maikel

A mis padres, familiares y hermana por su amor, paciencia, esfuerzo y dedicación.

A mi padre por ser tan especial y hacer todo cuanto pudo para llegar a este momento.

A mi madre por ser la luz que me ilumina el camino, sin su cristalina mirada estaría perdido.

A mi hermana por su cariño y apoyo además de ser mi guía en los años de estudiantes y que siempre lo será.

A mis amigos, que me apoyaron de una forma u otra y fueron muy importantes para mi.

A todo el que falta y debería estar.

A todos gracias.

Agradezco la palmada oportuna, el buen consejo y los argumentos.

RESUMEN

Desde el propio surgimiento de la Universidad de las Ciencias Informáticas, su objetivo principal ha sido la informatización de los centros laborales del país. Este trabajo de diploma surge a raíz de la necesidad existente en el CEINPET (Centro de Investigaciones del Petróleo) de tener un software capaz de automatizar el proceso de diseño de lechadas de cementación para pozos petroleros y que cuente con una BD capaz de guardar y permitir el trabajo con los datos que de estos se manejan diariamente.

Entre las principales herramientas utilizadas en el desarrollo de la aplicación se encuentra Net Beans 6.5 como IDE de programación, usando Java como lenguaje de programación desarrollador del proceso de implementación. Como Sistema Gestor de Base de Datos se usó PostgreSQL, se implementó una arquitectura de 3 capas, usando Hibernate como framework para el trabajo con el acceso a los datos y Swing en las interfaces. Los artefactos de la ingeniería de software propuestos fueron modelados usando la metodología RUP, con UML como lenguaje de modelado y Visual Paradigm como herramienta CASE para el modelado de los artefactos.

En el presente documento se exponen una serie de aspectos relacionados con el tema de Base de Datos, que sirven de punto de partida para la selección de los elementos que conforman la propuesta. Además se muestran los resultados obtenidos en la investigación para el desarrollo del sistema ANACEM, posibilitando lograr el diseño de lechadas de cementación para pozos petroleros, logrando así un ahorro de tiempo y dinero, para las empresas petroleras, principalmente de nuestro país.

Palabras Claves

Gestión, Lechada, Proceso, Sistema, Pozo.

ÍNDICE DE FIGURAS

Figura 1: Diagrama de Flujo para el Sistema SPT de Adquisición de Datos	22
Figura 2: Logo Empresa Petrolera Sanjel.....	23
Figura 3: Infraestructura de Comunicaciones	27
Figura 4: Jerarquía de los diagramas UML 2.0, mostrados como un diagrama de clases	29
Figura 5: PgExplorer	42
Figura 6: PgAdmin3.....	42
Figura 7: Arquitectura usada en Hibernate.....	48
Figura 8: Problemas Que se resuelven con Hibernate.....	49
Figura 9: Diagrama de Clases del Dominio.....	53
Figura 10: Diagrama de Casos de Usos del Sistema.....	93
Figura 11: Diagrama de Clases Persistentes.....	97
Figura 12: Diagrama Entidad-Relación.....	98
Figura 13: Modelo de Despliegue.....	100
Figura 14: Diagrama de Componentes.....	102
Figura 15: Diagrama de CA CU Autenticar Usuario	109
Figura 16: Diagrama de CA CU Gestionar Usuario	109
Figura 17: Diagrama de CA CU Gestionar Pozo	110
Figura 18: Diagrama de CA CU Gestionar Prueba.....	110
Figura 19: Diagrama de CA CU Gestionar Aditivo.....	111
Figura 20: Diagrama de CA CU Visualizar Información	111
Figura 21: Diagrama de CA CU Generar Reporte	112

Figura 22: DC CU Autenticar Usuario	113
Figura 23: DC CU Gestionar Pozo Sección Insertar	113
Figura 24: DC CU Gestionar Pozo Sección Modificar	114
Figura 25: DC CU Gestionar Pozo Sección Eliminar.....	114
Figura 26: DC CU Gestionar Prueba Sección Insertar.....	115
Figura 27: DC CU Gestionar Prueba Sección modificar	115
Figura 28: DC CU Gestionar Prueba Sección eliminar	116
Figura 29: DC CU Gestionar Aditivo Sección Insertar.....	116
Figura 30: DC CU Gestionar Aditivo Sección Modificar	117
Figura 31: DC CU Gestionar Aditivo Sección Eliminar	117
Figura 32: DC CU Gestionar Usuario Sección Insertar	118
Figura 33: DC CU Gestionar Usuario Sección Modificar	118
Figura 34: DC CU Gestionar Usuario Sección eliminar.....	119
Figura 35: DC CU Visualizar Información Sección Buscar Pozo	119
Figura 36: DC CU Visualizar Información Sección Buscar Prueba	120
Figura 37: DC CU Visualizar Información Sección Buscar Aditivo.....	120
Figura 38: DC CU Visualizar Información Sección Buscar servicios	121
Figura 39: DC CU Generar Reporte.....	121
Figura 40: Diagrama de CD Autenticar Usuario.....	122
Figura 41: Diagrama de CD CU Gestionar Pozo	123
Figura 42: Diagrama de CD CU Gestionar Prueba.....	124
Figura 43: Diagrama de CD CU Gestionar Aditivo	125
Figura 44: Diagrama de CD CU Visualizar Información.....	126

Figura 45: Diagrama de CD CU Generar Reporte.....	127
Figura 46: Diagrama de CD CU Gestionar Usuario.....	128

Índice de Tablas

Tabla 1: Descripción de los Actores del Sistema.....	57
Tabla 2: Descripción del Caso de Uso Autenticarse	59
Tabla 3: Descripción del Caso de Uso Gestionar Pozos	65
Tabla 4: Descripción del Caso de Uso Gestionar Pruebas.....	72
Tabla 5: Descripción del Caso de Uso Gestionar Aditivo.....	78
Tabla 6: Descripción del Caso de Uso Visualizar Búsquedas.....	84
Tabla 7: Descripción del Caso de Uso Gestionar Reportes.....	86
Tabla 8: Descripción del Caso de Uso Gestionar Usuario.....	92
Tabla 9: Descripción de la Tabla Tipo_Fluido	129
Tabla 10: Descripción de la Tabla Usuario.....	130
Tabla 11: Descripción de la Tabla Rol	130
Tabla 12: Descripción de la Tabla Lechada	131
Tabla 13: Descripción de la Tabla Cemento	132
Tabla 14: Descripción de la Tabla Fluido	132
Tabla 15: Descripción de la Tabla Pozo	134
Tabla 16: Descripción de la Tabla Aditivo	135
Tabla 17: Descripción de la Tabla Prueba.....	136
Tabla 18: Descripción de la Tabla TiempoConsistencia	137

Tabla 19: Descripción de la Tabla Filtrado	137
Tabla 20: Descripción de la Tabla AguaLibre	138
Tabla 21: Descripción de la Tabla EstabilidadLechada.....	139
Tabla 22: Descripción de la Tabla SedimentaciónCemento	140
Tabla 23: Descripción de la Tabla TiempoFrague	140
Tabla 24: Descripción de la Tabla ResistCompresión.....	141
Tabla 25: Descripción de la Tabla Reología.....	142
Tabla 26: Descripción de la Tabla Lectura	142
Tabla 27: Descripción de la Tabla Gelatinosidad.....	143
Tabla 28: Descripción de la Tabla n_TipoCemento.....	144
Tabla 29: Descripción de la Tabla n_Compañía	144
Tabla 30: Descripción de la Tabla n_Tapón.....	145
Tabla 31: Descripción de la Tabla n_Schedule.....	145
Tabla 32: Descripción de la Tabla Pozo_Prueba.....	145
Tabla 33: Descripción de la Tabla Aditivo_Prueba.....	146
Tabla 34: Descripción de la Tabla n_Tipo_Lechada	146
Tabla 35: Descripción de la tabla Tipo_Revestimiento	147
Tabla 36: Descripción de la Tabla Tipo_Aditivo	147

ÍNDICE

DEDICATORIA..... II

AGRADECIMIENTOS GENERALES III

RESUMEN..... V

ÍNDICE DE FIGURAS..... 1

Índice de Tablas..... 3

ÍNDICE 5

INTRODUCCIÓN 10

CAPÍTULO 1. Fundamentación teórica del tema..... 15

 1.1 Introducción..... 15

 1.2 Conceptos asociados al dominio del problema..... 15

 1.2.1 Sistema..... 15

 1.2.2 Sistema Informático..... 15

 1.2.3 Gestión 16

 1.2.5 Lechadas de Cemento 16

 1.3 Objeto de Estudio 16

 1.3.1 Flujo actual de procesos. 16

 1.3.2 Importancia de una buena gestión de las lechadas para la cementación de pozos petroleros. ... 17

 1.4 Situación Problemática..... 18

 1.5 Estado del arte..... 19

 1.5.1 TEI Sistema SPT de Información Geológica y de Perforación en línea..... 20

 1.5.2 DeepCRETE (Lechadas de cementación para aguas profundas.) 22

 1.5.3 Sanjel..... 23

 1.6 Conclusiones Parciales 25

CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar..... 26

 2.1 Introducción..... 26

 2.2 Las Tecnologías de la Información y las Comunicaciones (TIC) 26

2.3 La Informática aplicada a la industria petrolera.	27
2.4 El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.	28
2.4.1 Ventajas de UML.	29
2.4.2 ¿Por qué se escogió UML como Lenguaje de Modelado para el sistema?	30
2.5 El Proceso Unificado de Software (RUP) como soporte de la modelación de la solución propuesta.	30
2.5.1 Características del Proceso Unificado.....	30
2.5.2 ¿Por qué se usa RUP en este sistema?.....	31
2.6 Análisis de diferentes herramientas utilizadas para la realización del sistema.....	32
2.6.1 Herramientas CASE	32
2.6.1.2 Visual Paradigm.....	32
2.6.1.2.1 Características de Visual Paradigm	32
2.6.2 ¿Por qué se escogió Visual Paradigm para la solución propuesta?	34
2.7 Herramientas existentes para la modelación y gestión de Bases de Datos.	35
2.7.1 Embarcadero ER/studio 6.0.1	35
2.7.1.1 Funcionalidades de ER/studio:.....	35
2.7.2 PHPMyAdmin.....	36
2.7.2.1 Características de PHP My Admin.	36
2.7.3 SQL	36
2.7.3.1 Características de SQL	37
2.7.4 PhpPgAdmin	37
2.7.4.1 Características de PhpPgAdmin.	37
2.7.4.2 Funcionalidades de PhpPgAdmin.	38
2.7.5 PostgreSQL.....	38
2.7.5.1 Características de PostgreSQL.....	38
2.7.5.2 Ventajas de PostgreSQL.....	39
2.7.5.3 Otras herramientas de administración con PostgreSQL.....	41
2.7.6 MySQL.....	43

2.7.6.1 Características de MySQL	43
2.7.7 ¿Por qué se utilizó PostgreSQL como gestor de BD para ANACEM?	43
2.8 Lenguajes de programación para la implementación de aplicaciones de escritorio.	44
2.8.1 C ++	44
2.8.2 C#.....	44
2.8.2.1 Definición de C#.....	44
2.8.2.2 Características de C#	44
2.8.2.4 Desventajas de C#	45
2.8.3 Java.....	45
2.8.3.1 Definición de Java	45
2.8.3.2 Características de Java.....	45
2.8.3.3 Ventajas de Java	46
2.8.3.4 ¿Por qué Java como lenguaje de programación para la implementación del sistema?	46
2.9 Utilización de Framework.....	47
2.9.1 Hibernate	47
2.9.1.1 Definición de Hibernate.....	47
2.9.1.2 Hibernate – Características	47
2.9.1.3 Arquitectura de Hibernate.....	47
2.9.1.4 Ventajas de Hibernate.....	48
2.9.1.5 ¿Por qué usamos Hibernate para la implementación de este sistema?	49
2.9.2 Swing	50
2.10 Conclusiones Parciales.....	50
CAPÍTULO 3 Presentación de la solución propuesta.....	51
3.1 Introducción.....	51
3.2 Modelo Conceptual o Modelo de Dominio.....	51
3.3 Entorno donde trabajará el sistema	52
3.3.1 Conceptos Principales del entorno	52
3.3.2 Eventos Principales del entorno.....	52

3.3.3 Diagrama de clases del Dominio.....	53
3.3.4 Glosario de términos del Dominio	54
3.4 Consultas más importantes.	54
3.5 Requerimientos Funcionales	55
3.6 Requerimientos No Funcionales	56
3.7 Descripción del Sistema Propuesto	57
3.7.1 Descripción de los actores.....	57
3.7.2 Casos de Usos del Sistema.....	58
3.7.2.1 Descripción de los Casos de Usos del Sistema.	58
3.7.2.1.1 Caso de Uso Autenticar Usuario	58
3.7.2.1.2 Caso de Uso Gestionar Pozos	59
3.7.2.1.3 Caso de Uso Gestionar Prueba.....	65
3.7.2.1.4 Caso de Uso Gestionar Aditivo	72
3.7.2.1.5 Caso de Uso Visualizar Búsquedas.....	78
3.7.2.1.6 Caso de Uso Generar Reporte	85
3.7.2.1.2 Caso de Uso Gestionar Usuario.....	86
3.7.3 Diagrama de Casos de Usos del Sistema.....	93
3.8 Conclusiones Parciales.....	93
CAPÍTULO 4 “Construcción de la solución propuesta”	94
4.1 Introducción.....	94
4.2 Diagramas de Clases	94
4.3 Diagramas de Interacción.	95
4.4 Principios de diseño	95
4.4.1 Diagrama de clases del diseño.	95
4.4.2 Patrones de diseño.....	95
4.5 Diseño de la Base de Datos	96
4.5.1 Modelo lógico de datos. Diagrama de clases persistentes.....	96
4.5.2 Modelo físico de datos. Diagrama Entidad-Relación.....	97

4.6 Generalidades de la Implementación 98

 4.6.1 Estrategia de Integración. 98

 4.6.2 Descripción de los principales algoritmos a implementar. 99

4.7 Modelo de Despliegue..... 99

4.8 Modelo de Implementación..... 100

 4.8.1 Diagrama de Componentes 101

4.9 Conclusiones Parciales..... 102

CONCLUSIONES GENERALES..... 103

RECOMENDACIONES 104

REFERENCIAS BIBLIOGRÁFICAS..... 105

BIBLIOGRAFÍA 107

ANEXOS..... 109

GLOSARIO DE TÉRMINOS 148

“Los desarrolladores pueden construir y probar las especificaciones, pero los usuarios son los que en realidad aceptan o rechazan la operatividad actual”

Bernard Boar

INTRODUCCIÓN

En el mundo actual es evidente el auge que ha alcanzado el uso de las tecnologías de la Informática y las Comunicaciones como apoyo a los centros labores, al auto aprendizaje y como medio de comunicación entre las personas.

Las aplicaciones informáticas que gestionan información son cada vez más utilizadas por las instituciones, debido a que permiten sustituir los procedimientos tradicionales de manipulación y control de la información, que se realizan manualmente, por métodos automatizados de almacenamiento que proveen un ambiente de trabajo para la toma de decisiones y el manejo de resultados.

Muchas empresas extranjeras que se dedican al trabajo con lechadas de cementación en la actualidad cuentan con software que pudieran resolver el problema, pero que son software muy caros y de una difícil adquisición para el centro de investigación del petróleo.

La actividad de cementación de los pozos petroleros es fundamental en la durabilidad o vida del pozo. Cuenta con una serie de actividades dentro de la misma de las que depende una correcta selección de dicha lechada, es posible jugar con la velocidad de fraguado del cemento mediante productos retardadores o acelerantes. Por ejemplo, en el caso de que se quiera limitar la penetración del cemento en las formaciones, puede acelerarse el fraguado mediante combinaciones de cemento/silicato o cemento/bentonita/gas.

En los pozos verticales la cementación constituye una operación de singular importancia.

Tiene por objeto obtener la unión de la tubería con la pared del pozo para:

- ✚ Evitar que las aguas superficiales contaminen los acuíferos.

- ✚ Evitar la comunicación de un acuífero utilizable con otro u otros contaminados o que constituyan un horizonte ladrón o que se pretendan utilizar.
- ✚ Aumentar la resistencia mecánica y a la corrosión de las tuberías de revestimiento.
- ✚ En casos especiales proporcionar a un tramo de pozo la hermeticidad necesaria para realizar en él inyecciones a presión, bien sea para hacer un desarrollo con dispersantes o por acidificación, o por fracturación hidráulica.

No obstante, en ocasiones se pueden realizar cementaciones con otros objetivos como formar un tapón de sellado en el fondo del pozo o corregir desviaciones (o a veces para provocarlas) durante la perforación.

El tipo de cemento y los aditivos que se utilicen dependerán de cada caso en concreto. Por ejemplo, para cerrar grandes cavidades durante la perforación se suele emplear cemento al que se le ha añadido productos colmatantes y/o expansivos.

En el caso de cementaciones parciales de la tubería si se intenta aislar una capa "contaminante", la elección del tipo de cemento debe realizarse en función de parámetros físico-químicos tales como la litología del terreno, la composición química del agua, etc., pero también de las propias limitaciones de los equipos de cementación en cuanto a capacidades (volumen y presión) de inyección de la lechada.

Debido a la complejidad de las cementación, los tipos de lechadas varían por cada etapa de revestimiento, más aún en la cementación de liner, ya que las arenas de estas zonas presentan cambios bruscos con respecto a la presión (dientes de sierra) lo que hace difícil mantener un diseño específico tanto para la perforación como la cementación, con el propósito de evitar daños en la formación.

El principal objetivo en una cementación es conseguir un buen aislamiento zonal en el espacio anular, este aislamiento depende en gran medida de la eficiencia de desplazamiento del fluido de perforación por el cemento y por la condición mecánica del hoyo. **(CIED, 2003)**

El Centro de Investigación del Petróleo (CEINPET), es un centro encargado de las investigaciones que por más de 20 años se realizan en la empresa del petróleo, siendo único de su tipo en el país, diversas son

las actividades investigativas que en este se realizan, la mayoría de ellas con suma importancia para el desarrollo de la industrias petrolera en el país.

Como parte de las labores investigativas de este centro se encuentra el análisis del lodo que se usa en cada cementación y en cada perforación que se haga en los pozos, el análisis de las complejidades que pueden tener dichos pozos cuando están siendo perforados y otros temas como es el de cementación el cual es el más importante en este momento.

Actualmente en el CEINPET, la selección de cada lechada que se utiliza para la cementación de los pozos de petróleo, es un aspecto no informatizado y que solo cuenta para la recolección de los datos históricos de investigaciones anteriores con hojas de cálculo Excel para su selección y para el cálculo de los parámetros que son necesarios a la hora de elegir la lechada correcta.

En la empresa Cuba Petróleo (CUPET) en general no existen bases de datos que recojan los resultados de los análisis de laboratorios para las cementaciones de pozos de petróleo. Estos datos actualmente se encuentran en informes de pozos que se guardan en formato duro (hojas) y en tablas Excel en el CEINPET, lo que trabajar con ellos se hace tedioso, además de no estar exentos a errores humanos, y hacer más rápido este trabajo trae consigo un ahorro considerable de tiempo y dinero.

Resolver manualmente un problema de este tipo, requiere de varias horas de trabajo, dependiendo de los recursos involucrados, sin embargo es muy probable que la solución obtenida, no sea la más adecuada y se tenga que negociar el resultado con los clientes.

La realización de un sistema de gestión para las lechadas de cementación tiene una gran importancia a la hora de brindar una mejor organización al acceso a la información que se maneja en dicha empresa, permitiendo agilizar los procesos de realización de formulaciones de cementación para pozos futuros, ya que esta se basa en datos recopilados de perforaciones anteriores. Es por eso que el sistema de Gestión de Lechadas de Cementación para Pozos Petroleros (ANACEM), se implementará con el objetivo de automatizar dichos procesos para este laboratorio, y hacer que se diseñen las lechadas de una manera más eficaz con la que se pueda ganar en tiempo y ahorra dinero a la hora de realizar procesos de este tipo.

Por tanto, el **problema científico a resolver** que se presenta es la necesidad de automatizar los procesos de gestión de las lechadas para la cementación de los pozos petroleros.

El **objeto de estudio** de este trabajo de diploma es el proceso de gestión de lechadas para la cementación de los pozos petroleros, por lo que se enmarca esta investigación en un **campo de acción** como es la gestión de lechadas para la cementación de los pozos petroleros.

Para resolver dicho problema se tiene como **objetivo general** desarrollar un sistema que le permita a la empresa CEINPET automatizar los procesos de gestión de lechadas para la cementación de los pozos petroleros, el cual, para su mejor desarrollo, se encuentra derivado en los **objetivos específicos** siguientes:

- ✚ Realizar un estudio de la actividad de cementación que se realiza en los pozos.
- ✚ Analizar la actividad de gestión de lechadas para la cementación de los pozos petroleros.
- ✚ Realizar el análisis, diseño e implementación del sistema utilizando PostgreSQL como Gestor de Base de Datos.

Para darle cumplimiento a los objetivos específicos, se propone cumplir con las siguientes **tareas investigativas**:

- ✚ Estudio del estado del arte sobre las tecnologías más actuales que existen para el diseño de lechadas de cementación para pozos petroleros.
- ✚ Realizar búsquedas de información en diversas fuentes sobre sistemas gestores de base de datos (SGBD) y lenguajes de programación de escritorio.
- ✚ Establecer los métodos, herramientas y procedimientos más factibles para el desarrollo de la BD, y realizar un estudio de las mismas.
- ✚ Realizar el estudio del negocio para el diseño de lechadas para la cementación.
- ✚ Desarrollar el análisis y diseño utilizando la metodología y el lenguaje de modelado escogido.

✚ Desarrollar la BD para la gestión de los datos necesarios en el sistema.

✚ Implementar el sistema.

Para la realización de esta tesis se hace uso de los **métodos teóricos** y **empíricos** debido a que esta fase de elaboración de una herramienta que permita la gestión de lechadas de cementación en el laboratorio de lodos y cementos del CEINPET, para obtener información que permite a su vez tener los datos necesarios para dar respuesta al problema. Dentro de los métodos teóricos se utiliza el **método histórico-lógico** porque se realizó un estudio del proceso de gestión de lechadas de cementación para conocer sobre el problema a resolver desde sus principales etapas en su trayectoria, posibilitando así saber cuál es la verdadera situación del tema en la actualidad.

Dentro de los métodos empíricos **la observación** siempre va a estar presente, este método permitió conocer la realidad mediante percepción del objeto de estudio, es decir, cuando se comenzó la investigación la observación permitió saber cómo son los problemas existentes, teniendo como parte además de la misma las **entrevistas**.

El presente trabajo de diploma contará con 4 capítulos:

CAPÍTULO 1: Se encuentra la fundamentación teórica del tema que se trata en la investigación, así como los conceptos asociados al objeto de estudio.

CAPÍTULO 2: En este capítulo se registrarán los métodos, herramientas. Además se hace una descripción de las tendencias y tecnologías actuales que se usan en el mundo, las cuales se utilizaron como base para la propuesta que se hace.

CAPÍTULO 3: En este capítulo lo que se hace es un “Análisis del Sistema a desarrollar” donde se hace el diagrama de clases conceptuales, CU del sistema, requerimientos del Sistema, etc.

CAPÍTULO 4: Construcción de la Solución propuesta, aquí se modelan los diagramas de clases y de diseño, se describe como implementar el sistema.

CAPÍTULO 1. Fundamentación teórica del tema

1.1 Introducción

En este capítulo se hará un estudio de los diferentes sistemas software existentes en el mundo de la informática aplicada a la industria petrolera, centrándonos en los que se dedican al estudio de las lechadas de cementación para pozos petroleros.

Se expondrá además los conceptos principales asociados al tema, el objeto de estudio y la situación problemática ya de forma general, ya que se aborda algo de esto en la introducción, así como un breve resumen de la importancia de una buena gestión de las lechadas de cementación.

1.2 Conceptos asociados al dominio del problema

1.2.1 Sistema

Un sistema es una reunión o conjunto de elementos relacionados. Puede estructurarse de conceptos, objetos y sujetos. **(Hernández, 2008)**

Los sistemas se componen de otros sistemas a los que llamamos subsistemas. En la mayoría de los casos, podemos pensar en sistemas más grandes o súper ordinales, los cuales comprenden otros sistemas que llamamos sistema total y sistema integral.

1.2.2 Sistema Informático

Un sistema informático es el conjunto de hardware, software y de un soporte humano. **(Babylon.com)**

Un sistema informático típico emplea una computadora que usa dispositivos programables para capturar, almacenar y procesar datos.

La computadora personal o PC, junto con la persona que lo maneja y los periféricos que los envuelven, resultan de por sí un ejemplo de un sistema informático.

1.2.3 Gestión

El concepto de gestión hace referencia a la acción y al efecto de gestionar o de administrar. Gestionar es realizar diligencias conducentes al logro de un negocio o de un deseo cualquiera. **(Definición.de, 2008)**

1.2.4 Lechada

Mortero de cemento que contiene una gran cantidad de agua que le da la consistencia de un líquido viscoso y permite ser utilizado para rellenar cavidades y juntas entre materiales adyacentes. **(Ideasafines.com, 2008)**

1.2.5 Lechadas de Cemento

Mezcla de cemento y agua. También llamada pasta de cemento. **(Ideasafines.com, 2008)**

1.3 Objeto de Estudio

EL objeto de estudio relacionado a esta tesis es el proceso de gestión de lechadas para la cementación de los pozos petroleros, este es un proceso que su resultado final se ve reflejado en el pozo, pero que comienza desde el laboratorio con el diseño de las lechadas que se utilizan para la cementación de los pozos petroleros.

1.3.1 Flujo actual de procesos.

Actualmente en el laboratorio de CEINPET el proceso comienza cuando el cliente solicita que se le diseñe una lechada para cementar un pozo de petróleo de características particulares, las cuales son entregadas al laboratorio en la planilla de solicitud de análisis del laboratorio de Cemento, como son TD¹, TDV², inclinación del pozo, densidad del lodo, entre otras.

Luego el laboratorio analiza los registros históricos, para buscar cementaciones de pozos con características semejantes.

¹ Profundidad Total Del Pozo

² Profundidad del Pozo Por la Vertical

Teniendo esto, empiezan a modificar esos records realizando las pruebas de Densidad de lechada, Prueba de fluido, Prueba de pérdida de filtrado, Prueba de reología, Prueba de compatibilidad del cemento espaciador-lodo, Prueba de esfuerzo compresivo, y el tiempo de bombeo³. Estas pruebas se realizan hasta alcanzar los resultados que pide el cliente o hasta que se obtengan algunos parecidos los cuales se negocian luego con el mismo.

Una vez terminado el trabajo se realiza un reporte con todo lo ocurrido durante el diseño de dicha lechada ya que este puede tener hasta más de una prueba.

1.3.2 Importancia de una buena gestión de las lechadas para la cementación de pozos petroleros.

Antes de analizar la importancia de una buena gestión de lechada para la cementación de pozos petroleros, es bueno conocer algunas de sus características y funciones que ayudarán a entender mejor la vitalidad de su función.

Características:

- La cementación es una operación única que tiene que realizarse correctamente desde el primer momento.
- Las cementaciones secundarias o remediales consumen tiempo, son caras y frecuentemente no son satisfactorias.
- La confección de un Programa de Cementación adecuado es fundamental para lograr los objetivos de la cementación.

Funciones:

Las lechadas se diseñan para que respondan a las exigencias durante la operación de cementación, pero sus funciones principales corresponden al cemento fraguado:

- El cemento endurecido evita la migración de fluidos de la formación al espacio anular.

³ Es el tiempo que tarda en ponerse dura la lechada con la que se está trabajando.

- El cemento da fortaleza a la camisa y la protege contra la corrosión por salmueras o gases corrosivos como CO₂ a altas temperaturas.
- Protege a la camisa del aplastamiento por las cargas laterales.
- El cemento fraguado en el anular impermeable y con la dureza requerida es complemento indispensable para que el encamisado del pozo cumpla las funciones ya enunciadas anteriormente.

Una vez analizado esto se llega a la conclusión de que una buena selección de esta lechada trae consigo un ahorro considerable de dinero y tiempo para las empresas petroleras, además de que de esta depende casi totalmente la vitalidad del pozo. Esta se realiza una sola vez, si sale mal, ya habrá una pérdida para la empresa.

De aquí la importancia que este proceso merece y que por muchas décadas se había visto como un mecanismo simple de desplazamiento del lodo por el cemento y asentamiento del tapón, cuya culminación exitosa se convertía en alivio operacional debido a que la mezcla de cemento cumplía su trabajo sin presentar un fraguado prematuro indeseable. **(Villa, 2008)**

1.4 Situación Problemática.

En la actualidad es cada vez más grande el crecimiento del uso de las tecnologías de la información, lo que conlleva a la necesidad de los sistemas automatizados en los centros laborales.

La selección de las lechadas para la cementación de los pozos petroleros es un paso muy importante en lo que a cementación respecta, lo que hace que una buena selección, de cómo resultado un ahorro considerable de dinero, así como el tiempo y la vitalidad del pozo, como se dijera en el argot petrolero “one shoot”, lo que significa que una mala selección pondría en peligro la vida del mismo.

El centro de investigación del petrolero (CEINPET) cuenta con un laboratorio especializado el cual se encarga de los lodos complejidades y diseños de cementaciones para los pozos de petróleo que en Cuba se perforan diariamente, donde al final se debe entregar a la compañía un documento que en el caso de la cementación se denomina Informe de Laboratorios de Cementos, con datos de dicha compañía, de la

composición de las lechadas, los tipos de pruebas y otras informaciones que son de uso exclusivo del laboratorio y de la compañía misma que no se pueden revelar a terceras personas. Este documento aparece recogido en los informes de los diseños de lechadas de cementación para los pozos en formato duro (hojas) y en tablas Excel que se encuentran guardadas en las computadoras de la institución, al igual que la planilla de solicitud de análisis del laboratorio de cemento que entrega la compañía a la hora de solicitar el diseño de la lechada y el reporte de las formulaciones evaluadas para la cementación de camisas, liners y tapones, que este es el documento donde se guardan todos los cambios realizados para lograr el resultado que solicita el cliente al final.

En estos momentos no existe a nivel de la empresa CUPET (Cuba Petróleo), una base de datos que recoja los resultados de los análisis de laboratorios para la cementación.

Actualmente en los laboratorios del CEINPET, el análisis para la selección de las lechadas para la cementación de los pozos petroleros se realiza manualmente, lo que hace tedioso el trabajo debido a las enormes búsquedas que se realizan en archivos que se encuentran en formato duro desde años pasados, además existen dos personas encargadas del diseño de estas lechadas lo que hace que se pueda introducir el error humano, ya que si una de ellas empieza un diseño y debe parar, al retomarlo la otra persona, puede que se equivoque a la hora de buscar los datos en dichos documentos.

Crear un sistema de gestión de lechadas permitiría agilizar el proceso de búsqueda de esta información, ya que los diseños nuevos se realizan tomando como base diseños realizados en anteriores ocasiones. Así como un ahorro considerable de tiempo y dinero para la empresa.

1.5 Estado del arte.

En el mundo son muy pocos los sistemas de gestión de lechadas que se han implementado, que de una forma u otra responden a las exigencias de las empresas, los software están siendo muy costosos, de mucha calidad y rapidez en cuanto a los servicios, de estructuras muy variadas donde cada empresa exige a los informáticos un enfoque que responda a la actividad que necesita informatizar, se han aplicado fundamentalmente las tecnologías de escritorio para la realización de los mismos.

Es muy grande la importancia que han adquirido en la sociedad la informatización de los servicios y fundamentalmente para las empresas petroleras, lo que ha provocado que en el mundo hayan surgido un sin número de sistemas automatizados dirigidos al control eficiente de los procesos que se llevan a cabo en la misma.

A continuación, se hace un estudio crítico sobre los principales sistemas existentes en el ámbito nacional e internacional para la gestión de lechadas de cementación.

En el ámbito nacional existe sólo una empresa que se dedica a la gestión de lechadas para la cementación de los pozos petroleros, la cual no cuenta con un sistema automatizado para esto.

Ámbito Internacional: Analizando el ámbito internacional son muy pocas las empresas que dan a conocer el software que usan en sus trabajos, pero podemos poner como ejemplo el que mostramos a continuación:

1.5.1 TEI Sistema SPT de Información Geológica y de Perforación en línea.

Reúne los datos de diferentes fuentes, realiza los cálculos adecuados según la actividad del equipo de perforación del momento, y transmite los resultados a diferentes periféricos.

Hardware

Para el Monitoreo de registro, Procesamiento y Análisis, la unidad de control geológico está equipada con tres Sistemas Integrados de Computación. Dos de estos Sistemas se instalan empleando una configuración en red para Adquisición, Procesamiento, Almacenamiento y Transmisión de Datos en Línea. Pueden instalarse terminales remotas en lugares estratégicos (oficina del representante del Cliente, oficina del geólogo de pozo, oficina principal, etc.) para permitir la representación alfanumérica y gráfica en tiempo real, en pantallas color. El tercer equipo del Sistema está destinado a los Servicios de Evaluación de Formaciones y al procesamiento de datos fuera de línea.

Software de aplicación

Además del Sistema SPT de Información de Perforación y Geológica en Línea, utilizan varios paquetes de software especializado en prestar asistencia para:

Evaluación Geológica y de Formaciones:

- ✚ Informe Diario Geológico y de Ingeniería.
- ✚ Curva de Avance de Perforación.
- ✚ Perfil de Evaluación de Formación.
- ✚ Columna Estratigráfica.
- ✚ Perfiles Complejos (Relación Cromatografía, Presión, Parámetros de Perforación, Temperatura, etc.)
- ✚ Análisis del Desempeño de la Perforación.
- ✚ Análisis de Relaciones Cromatografías.
- ✚ Informe Final de Pozo.

Optimización de la Perforación:

- ✚ Ingeniería de Perforación Básica.
- ✚ Cálculos Hidráulicos.
- ✚ Evaluación del Fluido de Perforación.
- ✚ Circulación del Fluido de Perforación.
- ✚ Cementación.
- ✚ Aplicaciones para Control de Pozo.
- ✚ Control de Perforación Direccional y Cálculo de Trayectoria para Pozo Direccional.
- ✚ Aplicaciones para Ingeniería de Producción.

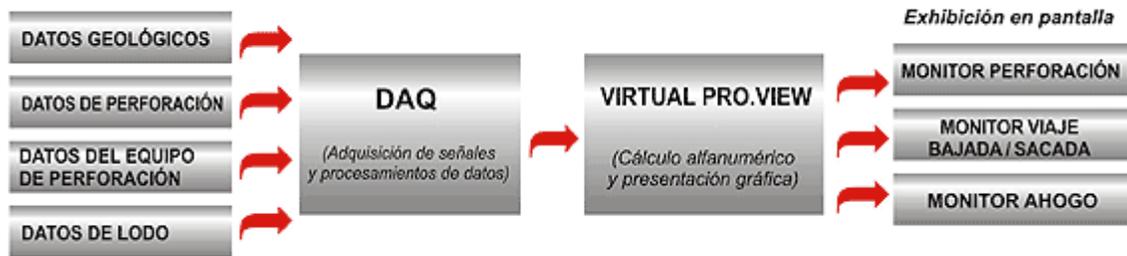


Figura 1: Diagrama de Flujo para el Sistema SPT de Adquisición de Datos

1.5.2 DeepCRETE (Lechadas de cementación para aguas profundas.)

Es un software *creado* por la empresa petrolera Schlumberger que se dedica al diseño de lechadas de cementación en aguas profundas.

Dentro de las funcionalidades que se encuentran en este software están las siguientes:

Aumentar la eficiencia cementación en aguas profunda: Cementación en aguas profundas DeepCRETE es una solución que reduce la espera del tiempo de espesamiento del cemento. El cemento aísla el sistema de formación y desarrolla la resistencia de compresión, haciendo esto de una manera más rápida que los sistemas convencionales de cemento Portland.

Control de flujo superficial con éxito: Cuando se combina con la tecnología de migración de gas, DeepCRETE es la solución adecuada para el control de flujo superficial.

La migración de gas es una tecnología que controla la pérdida de fluidos de baja permeabilidad, cuando se une con la distribución encargada del tamaño de la partícula de la tecnología DeepCRETE, la inhibición de la migración de fluidos se hace más efectiva.

El **DeepCEM** es otro software que forma parte de este que controla los aditivos utilizados para el trabajo de diseño de lechadas utilizando DeepCRETE, esta tecnología ofrecen un rápido desarrollo de la fuerza, incluso al estar frente a las bajas temperaturas encontradas en las formaciones superficiales.

Reducir los costos de perforación: Los operadores de la costa oeste y el sur de África trabajan con taladros que pueden traer consigo malformaciones cuando se trabaja a una profundidad consolidada de más de 6,300 pies en el agua, con una temperatura en el fondo marino de 36 ° F. Las carcasas poseen cuerdas que deben resistir las fuerzas de flexión o pandeo. Uno de sus principales objetivos es evitar la liberación de los hidratos de gas durante la perforación y cementación. El uso de cemento debe ser minimizado.

Los clientes han eliminado en un 7 % el gasto en la cementación de liners de producción mediante el uso de DeepCRETE en camisas 9 5/8 en cubiertas con vistas a la profundidad total. En el resultado del operador guardado cuatro por días el tiempo de perforación, el aislamiento fue superior al sistema de liners.

1.5.3 Sanjel



Figura 2: Logo Empresa Petrolera Sanjel

Sanjel Corporation es una empresa de propiedad privada, con sede en Canadá, una compañía internacional de servicios petroleros con más de dos décadas de experiencia en el sector. Como un importante competidor en el mercado mundial de petróleo y gas del mercado y la empresa de servicios petroleros privada más grande en Canadá, Sanjel ofrece cinco líneas de servicios especializados incluyendo Cementación, Tubo en espiral, de fractura y de nitrógeno. Cada línea de servicios se completa con su propia gama de productos especializados diseñados y fabricados por ellos mismos. Impulsado por el orgullo y reconocido por el rendimiento, Sanjel transforma su tecnología en líder en la industria para soluciones de empleo superior y mejorar la ejecución para los clientes.

Para Sanjel consolidar la vasta base de datos les permite encontrar el más fiable y rentable modo de trabajar las aplicaciones para aumentar al máximo la producción y así cumplir con sus clientes las diferentes necesidades. En un área específica de base, se identifica de manera proactiva los problemas como la pérdida de la circulación y se evita las corrientes de superficie en cubierta, y proporciona además

soluciones personalizadas para evitar la rehabilitación. Una nueva propiedad de las mezclas de cemento y aditivos se están desarrollando en su Centro Técnico, mientras que el perfeccionamiento continuo de las mezclas se lleva a cabo, usando sofisticada tecnología de laboratorio, cada combinación se prueba simulada bajo condiciones del mundo real para garantizar un rendimiento óptimo y control de calidad. Con tal de lograr el éxito de los productos diseñados y la fiabilidad de su propósito. Sanjel tiene por objeto reducir la pérdida de flujo de gas y los costes para los clientes, manteniendo su rendimiento superior sobre el terreno.

Sanjel Corporation mantiene así una ventaja competitiva en el mercado con la consolidación de su capacidad para ejecutar el trabajo en el campo y en el laboratorio. La integración de sus propios productos especializados y un equipo construido para esta finalidad garantiza su capacidad de ofrecer soluciones eficaces y con la excelencia operativa. Los clientes globales van a ellos con una total confianza en sus capacidades técnicas avanzadas, su equipo de críticos es de los mejores, y ha demostrado una gran capacidad para realizar el trabajo.

En estos momentos esta empresa se encuentra trabajando en Cuba.

Existen otras empresas en el mundo como la Halliburton y la Trican que son de gran prestigio a nivel internacional en cuanto a lo que a empresas petroleras respecta, pero que son de muy difícil acceso para nuestro país.

Teniendo en cuenta los software analizados anteriormente, se ve que muchos podrían resolver nuestro problema, pero que son software de empresas privadas y de origen norteamericanos, lo que imposibilita el acceso a cada uno de ellos, contando además del precio que vale cada uno de estos software, el diagrama de Flujo para el Sistema SPT de Adquisición de Datos sirve de ejemplo para el que queremos implementar, sólo cambiando en el procesamiento de los datos que nuestro sistema no haría esa parte, sino adquirir criterios de búsquedas y realizar las consultas necesarias a la BD.

1.6 Conclusiones Parciales

En este capítulo se han dado los principales conceptos, se ha hecho una explicación detallada del objetivo por el cual se lleva a cabo este trabajo, por tal motivo se describió en detalles el objeto de estudio del mismo, determinando el problema existentes que dio origen a la investigación demostrándose así la necesidad de implementar este sistema.

“El comprender no puede ser sustituido por una actividad intensa”

H.H. Williams

CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar

2.1 Introducción

En este capítulo se hará un estudio de las principales herramientas que se usan en el modelo de los diferentes sistemas informáticos, con el objetivo de realizar comparaciones y llegar a conclusiones que permitan definir las herramientas que se utilizarán en el desarrollo del sistema de gestión que se propone.

2.2 Las Tecnologías de la Información y las Comunicaciones (TIC)

Las Tecnologías de la Información y las Comunicaciones (TIC), son el conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética.

Las mismas incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual. Las nuevas formas de trabajo y la globalización de la economía imponen la necesidad del acceso instantáneo a la información y por tanto, de interconectar las distintas redes que se han ido creando, diseñándose nuevas arquitecturas de sistemas, en las que la función de comunicación es de igual importancia o superior por lo estratégico de la disponibilidad instantánea de la información.

A esto se añade, la existencia de unas infraestructuras de comunicación muy extendidas y fiables y un abaratamiento de los costes de comunicación lo que estimuló la aparición de nuevos servicios adecuados a las estrategias de las corporaciones. La comunicación instantánea es vital para la competitividad de una empresa, en un mundo en que la información se convierte en una salida más del sistema de producción.

Una amplia gama de soluciones como aplicaciones específicas para adquisición digital de datos geofísicos (*hardware* y *software*), de interpretación, de visualización de datos incluyendo animación digital, procesamiento de señales e imágenes, simulación de reservorios, evaluación de riesgos, pasando por herramientas de administración de datos corporativos, hasta aquellas que permiten compartir el conocimiento.

La gestión de la información es un proceso vital para las empresas de las industrias petroleras, las cuales acumulan, en bases de datos, mucha información, producto de la investigación, desarrollo, exploración y producción que hacen sobre pozos y yacimientos. En un mercado que funciona de manera globalizada, el manejo de los datos se convierte en un punto focal, tomar mejores decisiones que conlleven a realizar perforaciones más exitosas y minimizar gastos. La información en sí, dentro de la industria, es un activo. Gracias a sistemas informáticos avanzados, es posible reunir toda esa información y ponerla a funcionar de manera armónica.

La industria petrolera, particularmente en Latinoamérica, es una industria dispersa que ha crecido desordenadamente según ha ido evolucionando, creando a su paso varios silos de información. Oracle tiene una presencia firme en ese mercado, al que hoy le ofrece sus soluciones. En PDVSA, por ejemplo, los sistemas críticos de control de la producción residen sobre plataforma Oracle.

Pero no se trata sólo de gestionar la información: se trata de integrar los procesos, porque sin integración no hay eficiencia. La integración de datos y procesos permite tomar decisiones de manera correcta y tener una visión del negocio de 360 grados. Esto se logra al disponer de datos técnicos, financieros y económicos, entre otros, en una fuente única y veraz de información.

2.4 El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.

UML es un lenguaje de modelado, que permite analizar y diseñar sistemas orientados a objetos. **(Girón, 2007)**

Además es una de las herramientas más utilizadas a nivel mundial en el modelado de sistemas, está especializado en construir, especificar, visualizar y documentar los elementos que se producen en el

proceso de desarrollo de software de Sistemas de Software Orientados a Objetos. Está compuesto por diversos elementos gráficos que se combinan para conformar diagramas y debido a que es un lenguaje, cuenta con reglas para combinar tales elementos.

UML tiene un vocabulario en el que se identifican:

- ✚ Elementos: Abstracciones que constituyen los bloques básicos de construcción.
- ✚ Relaciones: Ligan los elementos.
- ✚ Diagramas: Es la representación gráfica de un conjunto de elementos y sus relaciones.

2.4.1 Ventajas de UML.

- ✚ UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas.
- ✚ UML es una consolidación de muchas de las notaciones y conceptos más usados orientados a objetos.

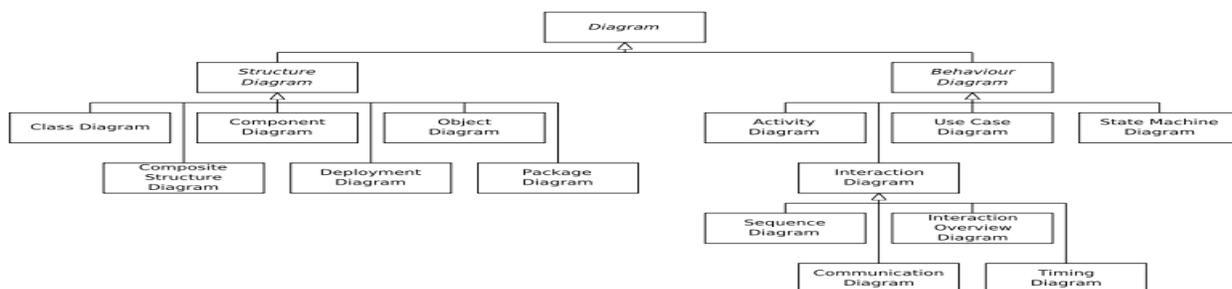


Figura 4: Jerarquía de los diagramas UML 2.0, mostrados como un diagrama de clases

En este sistema la versión de UML que se usa es la 2.0 debido a que Visual Paradigm trabaja con la 2.0, y es la herramienta CASE que se está usando en este momento en la construcción de este sistema.

Rational Rose trabaja con la versión 1.0 que se diferencia de esta en la agregación del diagrama de tiempo contenido en la versión 2.0.

2.4.2 ¿Por qué se escogió UML como Lenguaje de Modelado para el sistema?

Se escogió como lenguaje de modelado UML ("Unified Modeling Language") porque está consolidado como el lenguaje estándar en el análisis y diseño de sistemas informáticos. Además de que mediante UML es posible establecer una serie de requerimientos y estructuras necesarias para modelar un sistema de software previo al proceso intensivo de escribir código.

UML es un lenguaje que posee más características visuales que programáticas, las que facilitan a integrantes de un equipo participar e intercomunicarse fácilmente, estos integrantes siendo los analistas, diseñadores, especialistas de área y desde luego los programadores.

Una de las características más importantes que hacen que se escoja UML como lenguaje de modelado es que está diseñado para uso con software orientado a objetos, y tiene un uso limitado en otro tipo de cuestiones de programación.

2.5 El Proceso Unificado de Software (RUP) como soporte de la modelación de la solución propuesta.

El Proceso Unificado es un proceso de desarrollo de software que contiene un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de actitud y tamaños de proyecto. Está basado en componentes, lo cual quiere decir que el sistema software en construcción está formado por componentes software interconectados a través de interfaces bien definidas. **(Jacobson y otros, 2000)**

2.5.1 Características del Proceso Unificado.

Los verdaderos aspectos definitorios del Proceso Unificado, y que lo convierten en único, se resumen en tres frases clave - dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental.

Dirigido por los casos de uso:

Teniendo en cuenta que la razón de ser de un sistema es brindar servicios a los usuarios, RUP define caso de uso como el conjunto de acciones que debe realizar un sistema para dar un resultado de valor a un determinado usuario y los utiliza tanto para especificar los requisitos funcionales del sistema, como para guiar todos los demás pasos de su desarrollo, dígame diseño, implementación y prueba. **(Jacobson y otros , 2000)**

Estar centrado en la arquitectura:

La arquitectura es una vista del diseño completo con las características más importantes, dejando a un lado los detalles. Esta no solo incluye las necesidades de los usuarios e inversores, sino también otros aspectos técnicos como el hardware, sistema operativo, sistema de gestión de base de datos, protocolos de red; con los que debe coexistir el sistema. En otras palabras, la arquitectura representa la forma del sistema, la cual va madurando en su interacción con los casos de uso hasta llegar a un equilibrio entre funcionalidad y características técnicas.

Ser iterativo e incremental:

La alta complejidad de los sistemas actuales hace que sea factible dividir el proceso de desarrollo en varios mini-proyectos. Cada uno de estos mini-proyecto se les denomina iteración y pueden o no representar un incremento en el grado de terminación del producto completo. En cada iteración los desarrolladores seleccionan un grupo de casos de uso, los cuales se diseñan, implementan y prueban. La planificación de iteraciones hace que se reduzcan los riesgos de los costes de un incremento, no sacar al mercado un producto en el tiempo previsto, mantener la motivación del equipo pues puede ver avances claros a corto plazo y que el desarrollo pueda adaptarse a los cambios en los requisitos.

2.5.2 ¿Por qué se usa RUP en este sistema?

Para la implementación de este sistema se usa RUP debido a que este incorpora las mejores prácticas de desarrollo de software validadas comercialmente: desarrollo incremental guiado por casos de uso y centrado en la arquitectura.

Además de que existen varios pasos para la implementación de RUP que permiten que se adapte a proyectos pequeños.

Dentro de la configuración del mismo se puede adaptar el producto a la organización, así como una serie de pasos como son la evaluación del estado actual, establecer los objetivos, identificar los riesgos, planear la implementación y otras, por las cuales se decide usar RUP en este proyecto.

Se puede decir también que existe una versión (RUP Ágil) que está diseñada para que se use en proyectos de corta duración como el sistema que se va a implementar, donde no se generan menor cantidad de documentos, con menor cantidad de roles.

2.6 Análisis de diferentes herramientas utilizadas para la realización del sistema.

2.6.1 Herramientas CASE

Las **herramientas CASE** (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

2.6.1.2 Visual Paradigm

Visual Paradigm para UML es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. **(VP.com, 2007)**

2.6.1.2.1 Características de Visual Paradigm

- ✚ Soporte de UML versión 2.1.
- ✚ Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- ✚ Ingeniería inversa - Código a modelo, código a diagrama.

- ✚ Ingeniería inversa Java, C++.
- ✚ Generación de código - Modelo a código, diagrama a código.
- ✚ Editor de Detalles de Casos de Uso - Entorno todo-en-uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- ✚ Generación de objetos Java desde la base de datos.
- ✚ Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- ✚ Ingeniería inversa de bases de datos - Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- ✚ Generador de informes para generación de documentación.
- ✚ Editor de figuras.
- ✚ Alta velocidad a la hora de cargar y salvar los proyectos.
- ✚ Soporte multilenguaje. **(VP.com, 2007)**

2.6.2 Rational Rose

Rational ofrece las funciones necesarias para que las pequeñas y medianas empresas agilicen los procesos de desarrollo y distribución de los sistemas y el software más importantes en las siguientes categorías:

- ✚ Gestión de calidad
- ✚ Gestión de cambios y releases
- ✚ Gestión de arquitectura

✚ Gestión de procesos y productos

Todos los productos de Rational Rose dan soporte a Unified Modeling Language (UML), pero no son compatibles con las mismas tecnologías de implementación. Rational Rose Enterprise es un entorno de modelado que permite generar código a partir de modelos C++, Java/J2EE, Visual C++ y Visual Basic. **(ibm.com, 2006)**

Al igual que todos los productos de Rational Rose, ofrece un lenguaje de modelado común que agiliza la creación del software.

Incluye también estas funciones:

- ✚ Los componentes del modelo se pueden controlar independientemente, lo que permite una gestión y un uso de modelos más granular.
- ✚ Generación de código en lenguaje C++, Java y Visual Basic, con funciones configurables de sincronización entre los modelos y el código.
- ✚ Modelado en UML para diseñar bases de datos, que integra los requisitos de datos y aplicaciones mediante diseños lógicos y analíticos.
- ✚ Posibilidad de publicar en las web modelos e informes para mejorar la comunicación entre los miembros del equipo.

2.6.2 ¿Por qué se escogió Visual Paradigm para la solución propuesta?

Se escogió Visual Paradigm porque soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue, Visual Paradigm presenta una serie de características como es la generación de objetos Java desde la base de datos y que permite la ingeniería inversa de la misma, o sea desde Sistemas Gestores de Bases de Datos (DBMS) existentes, se puede transformar a diagramas de Entidad-Relación, haciendo así más fácil el trabajo con la Base de Datos.

Además brinda soporte multilinguaje, soporta la importación de proyectos realizados en Rational Rose, y tiene una gran compatibilidad con plataformas Java en cualquier sistema operativo. (Windows/Linux/Mac OS X).

A diferencia de Rational Rose, Visual Paradigm trabaja con la versión 2.0 de UML que incluye varios diagramas, como el diagrama de tiempo que no está presente en Rational.

Todo esto unido a la alta velocidad a la hora de cargar y salvar los proyectos, hacen que haya sido Visual Paradigm la herramienta CASE elegida para el desarrollo de este software.

2.7 Herramientas existentes para la modelación y gestión de Bases de Datos.

2.7.1 Embarcadero ER/studio 6.0.1

Es una herramienta de modelado de datos fácil de usar y multinivel, para el diseño y construcción de bases de datos a nivel físico y lógico. Direcciona las necesidades diarias de los administradores de bases de datos, desarrolladores y arquitectos de datos que construyen y mantienen aplicaciones de bases de datos grandes y complejos.

2.7.1.1 Funcionalidades de ER/studio:

- ✚ Capacidad fuerte en el diseño lógico.
- ✚ Sincronización bidireccional de los diseños lógico y *físico.
- ✚ Construcción automática de Base de Datos.
- ✚ Reingeniería inversa de Base de Datos.
- ✚ Documentación basada en HTML.
- ✚ Un Repositorio para el modelado.

2.7.2 PHPMyAdmin

PhpMyAdmin, fue creada por la misma persona que creó **phpAds**, Tobías Ratschiller.

PhpMyAdmin, no sólo es un programa OpenSource, sino que es “el programa” con mayúsculas para la administración de bases de datos MySQL en forma remota. No solamente por las funcionalidades que ofrece y que van mejorando con cada nueva versión, sino también porque a lo largo de los años ha sabido ganarse su lugar, y estar presente en casi todos los proveedores de hosting a nivel mundial, además de haber ganado numerosos premios. **(Reduser.com, 2005)**

PhpMyAdmin está disponible en 47 lenguajes diferentes, y se puede cambiar entre los distintos lenguajes disponibles, simplemente seleccionando el lenguaje de nuestro interés desde una lista desplegable, lo cual la hace sumamente adaptable a muchos usuarios y países.

2.7.2.1 Características de PHP My Admin.

- ✚ Administración completa de las Bases de Datos.
- ✚ Administración completa de las Tablas.
- ✚ Ejecuta sentencias SQL.
- ✚ Exporta datos a diferentes formatos.
- ✚ Administra usuarios y privilegios de MySQL.
- ✚ Es un Administrador Multiplataforma.
- ✚ Esta herramienta se encuentra bajo la Licencia GNU/GPL (GNU General Public License).

2.7.3 SQL

SQL (*Structured Query Language*), es el lenguaje que se utiliza para realizar una consulta a la base de datos. Por ejemplo cuando utilizamos un buscador, basado en el texto que hayamos introducido, se arma la consulta, que es enviada a la base de datos, y ejecutada. **(Reduser.com, 2005)**

2.7.3.1 Características de SQL

- ✚ Facilidad de instalación, distribución y utilización.
- ✚ SQL Server posee una gran variedad de herramientas administrativas y de desarrollo que permite mejorar la capacidad de instalar, distribuir, administrar y utilizar SQL Server.
- ✚ Administrativas (Administrador Corporativo).
- ✚ Desarrollo (Analizador de Consultas).
- ✚ Puede utilizarse el mismo motor de base de datos a través de plataformas que van desde equipos portátiles que ejecutan Microsoft Windows® 95 ò 98 hasta grandes servidores con varios procesadores que ejecutan Microsoft Windows NT®, Enterprise Edition.
- ✚ Almacenamiento de datos.
- ✚ SQL Server incluye herramientas para extraer y analizar datos resumidos para el proceso analítico en línea (OLAP, Online Analytical Processing). SQL Server incluye también herramientas para diseñar gráficamente las bases de datos y analizar los datos mediante preguntas en lenguaje normal.
- ✚ SQL Server se integra con el correo electrónico, internet y Windows, permitiendo una comunicación local.

2.7.4 PhpPgAdmin

PhpPgAdmin: Es un sistema para administrar un servidor PostgreSQL, programado en PHP. (Sentidowbe.com, 2004)

2.7.4.1 Características de PhpPgAdmin.

Entre las características que ofrece destaca la posibilidad de administrar varios servidores, soporte a múltiples versiones de PostgreSQL, administración de usuarios, grupos, bases de datos, esquemas, etc.,

manipulación sencilla de datos, exportar los datos a diferentes formatos, importar sentencias SQL y mucho más.

2.7.4.2 Funcionalidades de PhpPgAdmin.

Tiene todas las funcionalidades necesarias para realizar todo tipo de acciones sobre la base de datos: múltiples servidores PostgreSQL, administración de usuarios, bases de datos, esquemas, tablas, vistas, funciones, triggers, manipulación de datos, ejecutar sentencias SQL, backups... Es multi-idioma, fácil de instalar y configurar.

2.7.5 PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde la década de 1980. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. PostgreSQL es ampliamente considerado como una de las alternativas de sistema de bases de datos de código abierto. **(Ezpinosa, 2005)**

2.7.5.1 Características de PostgreSQL

Soporte SQL92/SQL99

PostgreSQL implementa un subconjunto extendido de los estándares SQL92 y SQL99.

Transacciones

Permiten el paso entre dos estados consistentes manteniendo la integridad de los datos.

Integridad referencial

PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos PK y FK.

Bloqueos de tabla y filas

PostgreSQL ofrece varios modos de bloqueo para controlar el acceso concurrente a los datos en tablas. Algunos de estos modos de bloqueo los adquiere PostgreSQL automáticamente antes de la ejecución de una declaración, mientras que otros son proporcionados para ser usados por las aplicaciones.

Múltiples tipos de datos predefinidos

Como todos los manejadores de bases de datos, PostgreSQL implementa los tipos de datos definidos para el estándar SQL3 y aumenta algunos otros.

Soporte de tipos y funciones de usuario

PostgreSQL soporta operadores, funciones métodos de acceso y tipos de datos definidos por el usuario.

Incorpora una estructura de datos Array.

Conectividad TCP/IP, JDBC y ODBC

Interfaz con diversos lenguajes

2.7.5.2 Ventajas de PostgreSQL

PostgreSQL ofrece muchas ventajas para su compañía o negocio respecto a otros sistemas de bases de datos:

Instalación ilimitada

Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Algunos proveedores comerciales consideran a esto la principal fuente de incumplimiento de licencia. Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software.

Esto tiene varias ventajas adicionales:

Modelos de negocios más rentables con instalaciones a gran escala.

- ✚ No existe la posibilidad de ser auditado para verificar cumplimiento de licencia en ningún momento.
- ✚ Flexibilidad para hacer investigación y desarrollo sin necesidad de incurrir en costos adicionales de licenciamiento.

Mejor soporte que los proveedores comerciales

Además de nuestras ofertas de soporte, tenemos una importante comunidad de profesionales y entusiastas de PostgreSQL de los que su compañía puede obtener beneficios y contribuir.

Ahorros considerables en costos de operación

Este software ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que los productos de los proveedores comerciales, conservando todas las características, estabilidad y rendimiento.

Además de esto, estos programas de entrenamiento son reconocidamente mucho más costo-efectivos, manejables y prácticos en el mundo real que aquellos de los principales proveedores comerciales.

Estabilidad y confiabilidad legendarias

En contraste a muchos sistemas de bases de datos comerciales, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. Ni una sola vez. Simplemente funciona.

Extensible

El código fuente está disponible para todos sin costo. Si su equipo necesita extender o personalizar PostgreSQL de alguna manera, pueden hacerlo con un mínimo esfuerzo, sin costos adicionales. Esto es complementado por la comunidad de profesionales y entusiastas de PostgreSQL alrededor del mundo que también extienden PostgreSQL todos los días.

Multiplataforma

PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable).

Diseñado para ambientes de alto volumen

PostgreSQL usa una estrategia de almacenamiento para conseguir una mucha mejor respuesta en ambientes de grandes volúmenes. Los principales proveedores de sistemas de bases de datos comerciales usan también esta tecnología, por las mismas razones.

2.7.5.3 Otras herramientas de administración con PostgreSQL

PSQL

Es la herramienta canónica para la ejecución de sentencias SQL a través del shell del SO. Es una herramienta de tipo frontend que permite describir sentencias SQL, ejecutarlas y visualizar sus resultados. **(Ezpinosa, 2005)**

El método de ingreso puede ser mediante la inserción directa del código en la consola, o la ejecución de sentencias dentro de un archivo de texto. Provee de diversas metas comando para la ejecución de las sentencias, así como diversas opciones tipo shell propias de la herramienta.

PgExplorer

PgExplorer es una herramienta de desarrollo para PostgreSQL con una amplia interfaz gráfica. **(Ezpinosa, 2005)**

Características de PgExplorer

Entre sus características se incluye un vista en árbol de las bases de datos y sus respectivos objetos. Se puede realizar ingeniería inversa a través de sentencias SQL o scripts personalizados. Hay una amplia gama de asistentes que guían a través del proceso necesario para generar comandos SQL para varios objetos y sentencias.

También incluye un Diseñador de Consultas Gráfico

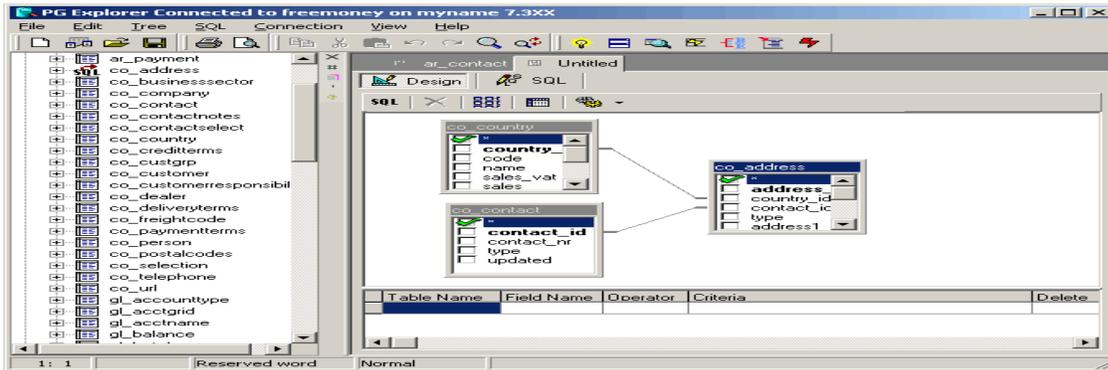


Figura 5: PgExplorer

PgAdmin3

Es una interfaz comprensible para el diseño y administración de una base de datos PostgreSQL, diseñada para ejecutarse en la mayoría de los Sistemas Operativos. (Ezpinosa, 2005)

La aplicación corre bajo GNU/Linux, Windows, 2000/XP. La interfaz gráfica soporta todas las características de PostgreSQL y facilita la administración.

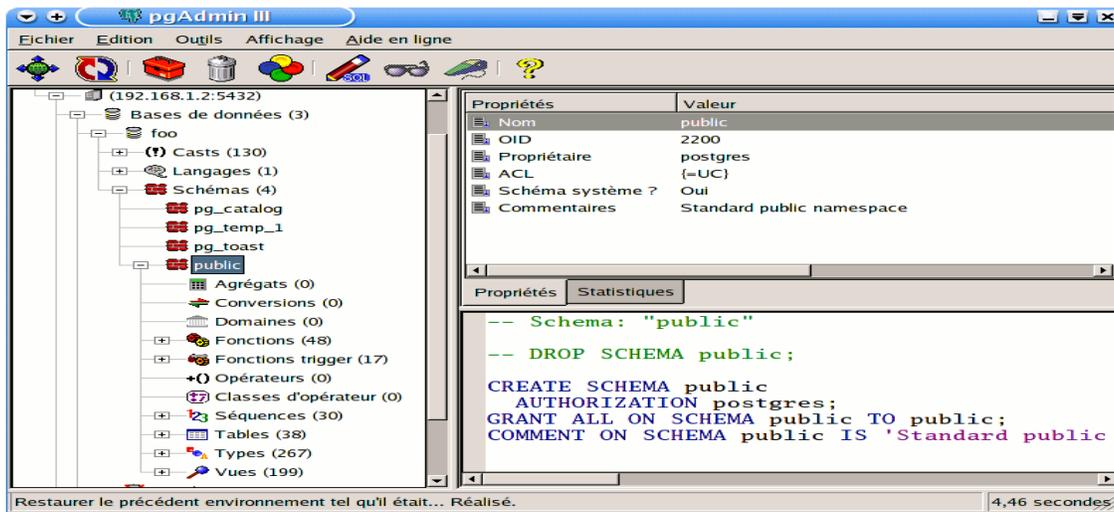


Figura 6: PgAdmin3

2.7.6 MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multi-hilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca. **(Netpecos.org, 2007)**

2.7.6.1 Características de MySQL

Las principales características de este gestor de bases de datos son las siguientes:

- ✚ Aprovecha la potencia de sistemas.
- ✚ Soporta gran cantidad de tipos de datos para las columnas.
- ✚ Gran portabilidad entre sistemas.
- ✚ Soporta hasta 32 índices por tabla.
- ✚ Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

2.7.7 ¿Por qué se utilizó PostgreSQL como gestor de BD para ANACEM?

PostgreSQL es ampliamente considerado como una de las alternativas de sistema de bases de datos de código abierto, que si se une esta característica a que es multiplataforma y que soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos PK y FK, hace que este sea una fuerte competencia como software a la hora de realizar una BD.

Además, PostgreSQL incorpora una estructura de datos Array que permite que se guarde la información de una forma más organizada y que a la hora de obtenerla se pueda acceder más rápido a esta.

En contraste a muchos sistemas de bases de datos comerciales, PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad, uno de los criterios principales a la hora de seleccionar PostgreSQL como gestor de BD en la implementación del sistema.

2.8 Lenguajes de programación para la implementación de aplicaciones de escritorio.

2.8.1 C ++

C++ es un lenguaje que abarca tres paradigmas de la programación: la programación estructurada, la programación genérica y la programación orientada a objetos. **(Bjarne, 2006)**

Visual C ++ 6.0 incluye un sofisticado editor de código fuente que soporta muchas características tales como auto-tabulación. En Visual C ++ 6.0, una nueva y emocionante característica ha sido añadido, el autocompletar, todo lo que hay que hacer es teclear el comienzo de una declaración de programación y el editor proporciona una lista de posibles complementos a elegir. Esta característica es muy útil cuando se trabaja con objetos de C ++.

2.8.2 C#

2.8.2.1 Definición de C#

Lenguaje de programación diseñado por Microsoft en 2001 como parte de su plataforma .NET. Combina el lenguaje de bajo nivel de C y la velocidad de la programación de alto nivel de Visual Basic. **(Hardh2o.com, 2007)**

2.8.2.2 Características de C#

- ✚ Facilidad de uso.
- ✚ Programación orientada a objetos.
- ✚ Administración de memoria.
- ✚ Seguridad en el manejo de datos.
- ✚ Sistema de tipos unificado.
- ✚ Uso de operadores.

2.8.2.4 Desventajas de C#

Las desventajas que se derivan del uso de este lenguaje de programación son que en primer lugar se tiene que conseguir una versión reciente de Visual Studio .NET, por otra parte se tiene que tener algunos requerimientos mínimos del sistema para poder trabajar adecuadamente tales como contar con Windows NT 4 o superior, tener alrededor de 4 gigas de espacio libre para la pura instalación, etc.

2.8.3 Java

2.8.3.1 Definición de Java

Java es un lenguaje de programación orientado a objetos desarrollado por la Sun Microsystems, una compañía famosa sobre todo por sus estaciones de faja alta. Inspirado al C++, Java fue proyectado con la finalidad de obtener un producto de pequeñas dimensiones, simple y portátil sobre diferentes plataformas y sistemas operativos, sea a nivel de código fuente que a nivel de código binario; lo que significa que los programas Java pueden ser ejecutados sobre cualquier computadora en la cual sea instalada la máquina virtual. **(Babylon.com, 2007)**

2.8.3.2 Características de Java

- ✚ Es un lenguaje de fácil uso, orientado a objetos.
- ✚ Es un lenguaje que facilita un entorno interpretado, pues tiene gran velocidad de desarrollo, y es de gran portabilidad.
- ✚ Es un lenguaje con ejecución multitarea.
- ✚ Permite identificadores.
- ✚ Tiene reglas sobre los tipos de datos.

2.8.3.3 Ventajas de Java

Primero: No se debe volver a escribir el código si se quiere ejecutar el programa en otra máquina (o sea otra arquitectura). Un sólo código funciona para todos los browsers compatibles con Java o donde se tenga una Máquina Virtual de Java (Mac's, PC's, Sun's, etc).

Segundo: Java es un lenguaje de programación orientado a objetos, y tiene todos los beneficios que ofrece esta metodología de programación.

Tercero: Un browser compatible con Java deberá ejecutar cualquier programa hecho en Java, esto ahorra a los usuarios tener que estar insertando "plug-ins" y demás programas que a veces quitan tiempo y espacio en disco.

Cuarto: Java es un lenguaje y por lo tanto puede hacer todas las cosas que puede hacer un lenguaje de programación: Cálculos matemáticos, procesadores de palabras, bases de datos, aplicaciones gráficas, animaciones, sonido, hojas de cálculo, etc.

Quinto: Si lo que interesa son las páginas de Web, ya no tienen que ser estáticas, se le pueden poner toda clase de elementos multimedia y permiten un alto nivel de interactividad, sin tener que gastar en paquetes carísimos de multimedia.

2.8.3.4 ¿Por qué Java como lenguaje de programación para la implementación del sistema?

El lenguaje de programación escogido fu Java porque además de que está inspirado en el C++, Java fue proyectado con la finalidad de obtener un producto de pequeñas dimensiones, simple y portátil sobre diferentes plataformas y sistemas operativos, ya sea a nivel de código fuente o a nivel de código binario; lo que significa que los programas Java pueden ser ejecutados sobre cualquier computadora en la cual sea instalada la máquina virtual.

En el lenguaje Java, un solo código funciona para todos los browsers compatibles con Java o donde se tenga una Máquina Virtual de Java.

Este lenguaje puede hacer todas las cosas que puede hacer un lenguaje de programación: Cálculos matemáticos, procesadores de palabras, bases de datos, aplicaciones gráficas, animaciones, sonido, hojas de cálculo, además de tener una lata compatibilidad con el Visual Paradigm y PostgreSQL que son la herramienta CASE y el gestor de BD que se seleccionó para el diseño e implementación de este software.

2.9 Utilización de Framework

2.9.1 Hibernate

2.9.1.1 Definición de Hibernate.

Es una herramienta para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones.

2.9.1.2 Hibernate – Características

- ✚ No intrusivo (estilo POJO)
- ✚ Muy buena documentación (fóruns para ayuda, libro)
- ✚ Comunidad activa con muchos usuarios
- ✚ Transacciones, caché, asociaciones, polimorfismo, herencia, persistencia transitiva, estrategias de fetching.

2.9.1.3 Arquitectura de Hibernate.

El API de Hibernate es una arquitectura de dos capas (Capa de persistencia y Capa de Negocio).

En la siguiente Figura se muestran los roles de las interfaces Hibernate más importantes en las capas de persistencia y de negocio de una aplicación J2EE. La capa de negocio está situada sobre la capa de persistencia, debido a que actúa como un cliente de la capa de persistencia.

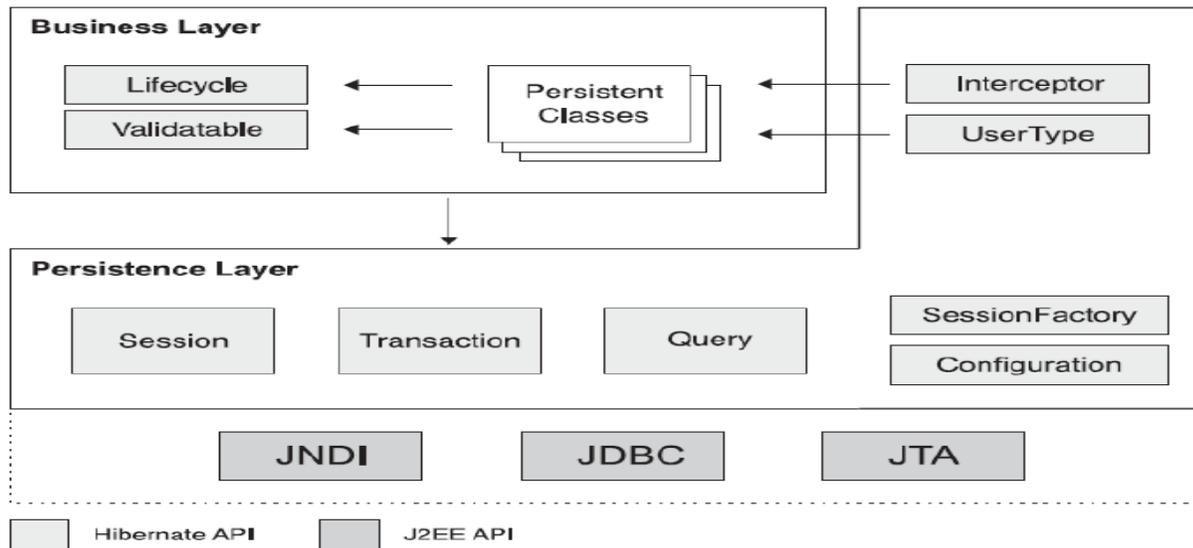


Figura 7: Arquitectura usada en Hibernate

2.9.1.4 Ventajas de Hibernate.

Prácticamente, no hay lugar donde el uso de este framework no sea útil, las aplicaciones Java que requieran una constante manipulación de base de datos requieren por su facilidad de manejo y versatilidad para los desarrolladores optar por Hibernate.

Si bien es cierto existe una brecha grande entre los elementos que intervienen en el análisis y desarrollo de las aplicaciones de los proyectos que actualmente están utilizando el paradigma orientados a objetos, esto aun se ve aplazado por un ancla el cual es la base de datos de tipo relacional, sin embargo en la actualidad este trabajo de conexión y manipulación de información de la base de datos ya no es tan tediosa gracias a la aparición de Hibernate, de la cual toda aplicación Web hecha en Java que necesite interactuar con la base de datos puede hacer uso, lo cual simplifica enormemente la programación de lógica.

2.9.1.5 ¿Por qué usamos Hibernate para la implementación de este sistema?

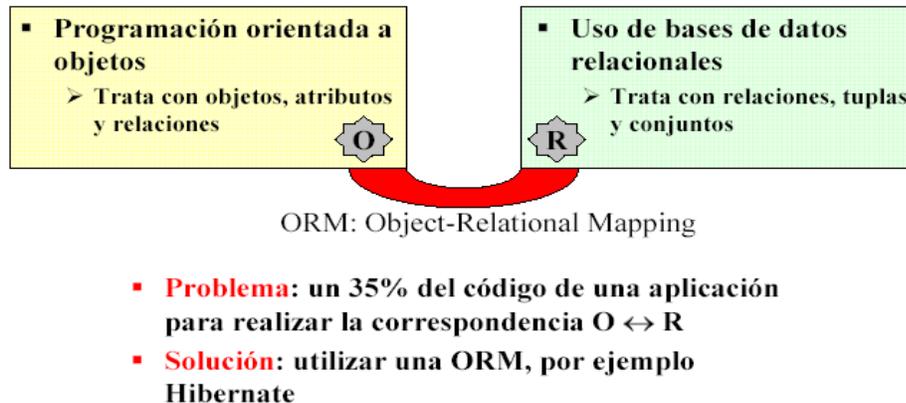


Figura 8: Problemas Que se resuelven con Hibernate

Se escogió Hibernate para el desarrollo del sistema porque cuando se quiere hacer que los objetos sean persistentes utilizando para ello una base de datos relacional, uno se da cuenta de que hay una desavenencia entre estos dos paradigmas, la también llamada diferencia objeto-relacional (object – relational gap”) e Hibernate como mapeador objeto-relacional (ORM) ayuda a evitar esta diferencia.

Además, son muchas las características y ventajas que presenta Hibernate que facilitan el trabajo en los sistemas que para el acceso a datos usan BD relacionales. Hibernate proporciona además un lenguaje para el manejo de consultas a la base de datos. Es una muy buena herramienta en lo que se refiere a mapeo clases en una base de datos relacional. Además Hibernate es menos invasivo que otros marcos de trabajo de mapeo O/R. Se utilizan Reflection y la generación de bytecodes en tiempo de ejecución, y la generación del SQL ocurre en el momento de la arrancada. Esto permite desarrollar objetos persistentes siguiendo el lenguaje común de Java: incluyendo asociación, herencia, polimorfismo, composición y el marco de trabajo Collections de Java.

2.9.2 Swing

Este framework es utilizado para el diseño de interfaces de usuario. Brinda los componentes gráficos y las ventanas independientes de las librerías usadas. La arquitectura que posee es particionada ya que es independiente de la plataforma en la que se esté desarrollando el software.

2.10 Conclusiones Parciales

En este capítulo se hizo un estudio de las diferentes herramientas existentes para el diseño e implementación de sistemas informáticos, llegando a conclusiones que permitieron que se escogieran las que se utilizarán en el sistema software que se implementará.

Se hizo además un estudio de las TIC y de cómo se comportan los sistemas informáticos en las industrias petroleras, que es el tema sobre el cual se está trabajando.

“Un computador hará lo que digas, pero ello puede ser muy diferente de lo que tengas en mente”

Joseph Weizenbaum

CAPÍTULO 3 Presentación de la solución propuesta

3.1 Introducción

En este capítulo se hará una presentación de la solución que se propone, en base de una descripción del dominio del problema que se debe resolver, con la descripción del sus actores, casos de usos y se presentarán las clases de dicho dominio.

3.2 Modelo Conceptual o Modelo de Dominio

El análisis orientado a objetos tiene por finalidad estipular una especificación del dominio del problema y los requerimientos desde la perspectiva de la clasificación por objetos y desde el punto de vista de entender los términos empleados en el dominio. Para descomponer el dominio del problema hay que identificar los conceptos, los atributos y las asociaciones del dominio que se juzgan importantes. El resultado puede expresarse en un modelo conceptual, el cual se muestra gráficamente en un grupo de diagramas que describen los conceptos (objetos).

El modelo conceptual no es una descripción de los componentes del software; representa los conceptos en el dominio del problema en el mundo real. Este constituye el artefacto más importante a crear durante el análisis orientado a objetos. **(Larman, 1999)**

Utilizando UML, un modelo de dominio se representa con un conjunto de diagramas de clases en los que no se define ninguna operación. Los modelos de dominio pueden mostrar:

Objetos del dominio o clases conceptuales.

Asociaciones entre las clases conceptuales.

Atributos de las clases conceptuales. **(Sánchez, 2007)**

Las clases del dominio aparecen en tres formas típicas:

Objetos del negocio: Representan cosas que se manipulan en el negocio.

Objetos del mundo real y conceptos de los que el sistema debe hacer un seguimiento.

Sucesos que ocurrirán o han ocurrido. **(Jacobson, et al., 2000)**

3.3 Entorno donde trabajará el sistema

3.3.1 Conceptos Principales del entorno

Modelos de solicitud: Es el modelo que emite el cliente donde solicita el diseño de una lechada determinada al laboratorio de lodos y cementos del CEINPET.

Partes diarios de perforación: Es la parte que emite diariamente el equipo de perforación donde se ponen los datos de esta, pudiendo así tener un control del proceso de perforación.

Consultas: Son las búsquedas que se realizan en los Schedule de los pozos. Esta se realiza sobre el libro de la norma internacional de la API.

Schedule: Son las tablas donde se registran los diseños de lechadas realizados a los pozos de petróleo. En este caso se refiere al libro de la norma internacional de la API.

3.3.2 Eventos Principales del entorno

Confección de tablas: Se realiza después de cada diseño o cuando se desee. Consiste en pasar a tablas Excel los datos que se encuentran en el registro diario de diseños de las lechadas de cementación.

Selección de datos: Consiste en realizar las consultas necesarias para obtener los datos para comenzar el diseño de una nueva lechada de cementación. Esta se realiza sobre pozos que tengan características semejantes al que se le está diseñando la lechada.

Obtención de formulaciones: Es el proceso que consiste en realizar todas las pruebas que el laboratorio considere hasta obtener una lechada con los parámetros de salidas que solicitó el cliente o con parámetros semejantes.

3.3.3 Diagrama de clases del Dominio

El objetivo del diagrama de clases del dominio es contribuir a la comprensión del contexto del sistema, y por lo tanto también contribuir a la comprensión de los requisitos del sistema que se desprenden de este contexto. El diagrama de clases del dominio, debe contribuir a una comprensión del problema que se supone que el sistema resuelve en relación a su contexto. A continuación la figura muestra el diagrama de clases del dominio propuesto.

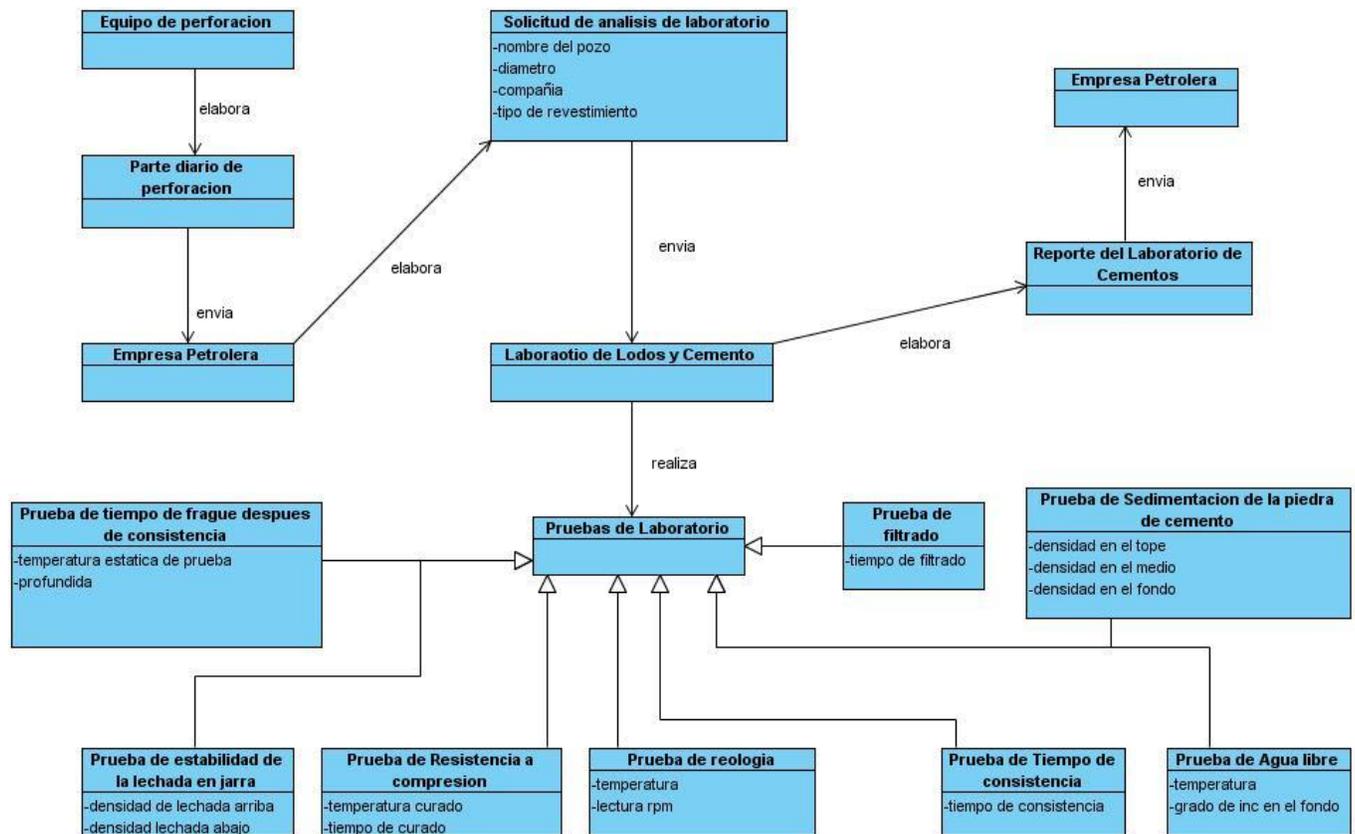


Figura 9: Diagrama de Clases del Dominio

3.3.4 Glosario de términos del Dominio

El glosario de términos junto al Modelo de Dominio ayuda a los usuarios, clientes, desarrolladores y otros interesados a utilizar un vocabulario común. La terminología común es necesaria para compartir el conocimiento con los otros. El glosario es muy útil para alcanzar un consenso entre los desarrolladores, relativo a la definición de los diversos conceptos y nociones, y para reducir en general, el riesgo de confusiones. Para construir un sistema software de cualquier tamaño, los ingenieros de hoy en día deben “fundir” el lenguaje de todos los participantes en uno sólo consistente.

En el Glosario de Términos se definen términos comunes importantes que utilizan los analistas al describir el sistema. Es un artefacto muy intuitivo para usarlo con terceras personas externas, como usuarios y clientes. **(Jacobson, et al., 2000)**

API: Abreviaturas de American Petroleum Institute, localizada en Washington, D.C. Este es el centro de la asociación para las industrias petroleras.

MD: Measure Depth. Es la profundidad total medida en el pozo.

TVD: Total Vertical Depth. Es la profundidad del pozo medido por la vertical.

BC: Unidades Bearden de consistencia.

BHCT: Bottom Hole Circulation Temperature. Es la temperatura de circulación en la boca del pozo.

BHST: Bottom Hole Static Temperature. Es la temperatura estática que tiene el pozo en ese momento.

3.4 Consultas más importantes.

- ✚ Solicitar información de los pozos evaluados en un rango de profundidad por la vertical (TVD).
- ✚ Solicitar información de los pozos evaluados en un rango de profundidad por la vertical (TVD) y un rango de inclinación seleccionado.
- ✚ Solicitar información de los pozos evaluados en un rango de temperatura BHCT seleccionado. Ejemplo entre 5 y 10 °C, 11 y 15, 16 y 20, 21 y 25, etc.

- ✚ Solicitar información de los pozos evaluados en un rango de temperatura BHCT seleccionado, tipo de lechada y rango de densidad
- ✚ Solicitar información de los pozos evaluados en un rango de temperatura BHCT seleccionado y rango de fecha de recepción del cemento por el cliente.
- ✚ Solicitar información de todos los trabajos realizados con cemento de las fechas de recepción del cliente seleccionadas Ej. mes Enero 2009.
- ✚ Solicitar información de los trabajos realizados por pozo.
- ✚ Solicitar información de los trabajos realizados por compañía.

3.5 Requerimientos Funcionales

Los requisitos funcionales son capacidades o funciones que el sistema deba cumplir, es decir, los requisitos funcionales muestran que es lo que debe hacer el sistema. **(Jacobson y otros, 2000)**

R1. El sistema debe permitir que los usuarios puedan autenticarse y darles un nivel previo de permiso, en dependencia a este, serán las funcionalidades que el usuario pueda realizar sobre el sistema.

R2. El sistema debe permitir al usuario Gestionar los pozos de petróleos que se encuentran en la BD. La gestión de los pozos incluye que se inserte, modifique o elimine un pozo a petición del usuario.

R3. El sistema debe permitir que se inserte, elimine o modifique una prueba realizada a cualquier pozo petrolero. El usuario podrá seleccionar la prueba y sobre esta realizar las operaciones que se definieron anteriormente.

R4. El sistema permitirá que se gestionen los aditivos para una compañía dada. Esto consiste en insertar, modificar o eliminar dichos aditivos para cualquier empresa que el usuario seleccione.

R5. El sistema permitirá al usuario visualizar la información que se encuentra en la base de Datos, ya sea de los pozos, pruebas, aditivos, o los servicios que se les ha prestado a cualquier compañía que el usuario seleccione para ver dicha información.

R6. El sistema permitirá al usuario generar el reporte de los trabajos realizados.

R7. El sistema permitirá al administrador que pueda gestionar usuarios, donde podrá eliminar, modificar o insertar el usuario que vaya a trabajar con el mismo.

3.6 Requerimientos No Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, conforman un grupo de categorías y representan las características de la aplicación siendo el complemento de los requisitos funcionales. **(Jacobson y otros, 2000)**

Requerimientos de Usabilidad: El sistema debe tener una interfaz de uso sencillo y de buen entendimiento por el usuario, complementada con un manual de usuario, fácil de utilizar, para facilitar así la interacción de estos usuarios con la misma (los diseñadores de las lechadas de cementación pueden ser personas con poca experiencia en el uso de aplicaciones informáticas).

Requerimientos de Seguridad: El sistema estará protegido de acceso no autorizado, sólo los usuarios autorizados tendrán acceso al mismo, garantizando a estos sólo acceder a la información que les concierne.

Requerimientos de Portabilidad: La aplicación podrá ser usada bajo cualquier familia del sistema operativo Windows, Linux u otro, es por ello que para al implementarla se hizo uso de herramientas de Programación y Gestión de Bases de Datos que son multiplataforma.

Requerimientos de Software: Debido a que para la realización del software se utilizaron herramientas multiplataforma, las PC cliente y servidor de BD podrá tener cualquier sistema operativo.

Requerimientos de Hardware: Se requiere que todas las computadoras tengan Procesador Intel Pentium IV, 512 MB de RAM y 2 Gb de espacio libre en disco duro como mínimo, pero teniendo en cuenta que el servidor de BD estará situado en la misma PC cliente se requiere que la PC tenga 80 Gb de espacio en disco.

Requerimientos de Hardware para dispositivos: Para el trabajo con la aplicación en la impresión de los reportes, podrá usarse impresoras de cualquier tipo y marca.

3.7 Descripción del Sistema Propuesto

3.7.1 Descripción de los actores

Los actores del sistema son personas u otros sistemas que son trabajadores en el negocio, así como actores del negocio que interactúan directamente con el sistema, en caso que estos últimos existan.

Actores	Justificación
Diseñador	Es la persona encargada de realizar todo lo referente al diseño de las lechadas de cementación, en su interacción con el sistema podrá realizar todas las funcionalidades que este brinde, excepto gestionar usuarios.
Consultor	Se refiere a la persona que tendrá en su interacción con el sistema un nivel menor de acceso, sólo podrá realizar búsquedas y generar el reporte de la misma.
Administrador	Es la persona que se encargara de gestionar los usuarios que trabajaran con el sistema, podrá realizar todas las demás operaciones..

Tabla 1: Descripción de los Actores del Sistema

3.7.2 Casos de Usos del Sistema

Los casos de uso del sistema son contratos que se establecen entre los actores y el sistema en el cual se le da respuesta a las funcionalidades definidas en los requerimientos funcionales.

3.7.2.1 Descripción de los Casos de Usos del Sistema.

3.7.2.1.1 Caso de Uso Autenticar Usuario

Caso de Uso:	Autenticar Usuario	
Actores:	Consultor	
Resumen:	El Caso de Uso se inicia cuando el consultor introduce su nombre de usuario y su contraseña para acceder a la aplicación, estos se verifican y finaliza dándole los permisos y habilitándole la entrada, el mismo termina cuando el usuario oprime el botón para la entrada de los datos.	
Precondiciones:	El usuario debe de estar registrado en la BD y la contraseña debe ser la correspondiente al mismo.	
Referencias	R1	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El consultor introduce el nombre de usuario y su contraseña.	1.1 El sistema verifica los datos del usuario. 1.2. En caso de ser correcto le permite realizar las funciones definidas para él y	

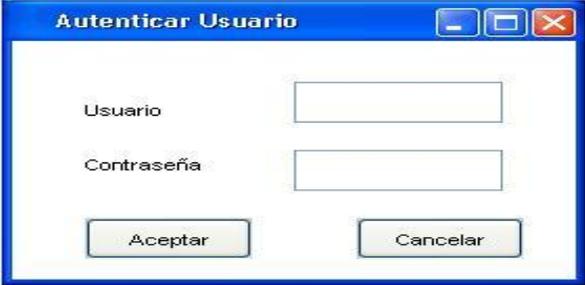
	termina el CU.
Prototipo de Interfaz	
	
Flujo Alternativo 1	
Acción del Actor	Respuesta del Sistema
	1.2 En caso de que el nombre de usuario o contraseña sean incorrectos se envía un mensaje de notificación.
Prototipo de Interfaz	
Poscondiciones	Se habilitan las funcionalidades según lo privilegios.

Tabla 2: Descripción del Caso de Uso Autenticarse

3.7.2.1.2 Caso de Uso Gestionar Pozos

Caso de Uso:	Gestionar Pozos
Actores:	Diseñador
Resumen:	El CU comienza cuando el diseñador selecciona la opción de gestionar

	un pozo, donde el mismo puede seleccionar la operación que desea realizar sobre el pozo, ya sea insertar, eliminar o modificar, y termina cuando acepta la opción seleccionada.
Precondiciones:	Que el usuario se haya autenticado como diseñador.
Referencias	R2
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
<p>1. El diseñador selecciona la opción gestionar pozos, para insertar, eliminar o modificar un pozo.</p>	<p>1.1 El sistema busca todos los datos del pozo existentes.</p> <p>1.2 El sistema muestra un formulario con los campos bloqueados, además de una tabla con los pozos que se encuentran en la Base de Datos. El diseñador podrá:</p> <ul style="list-style-type: none">  Insertar Pozo  Modificar Pozo  Eliminar Pozo
<p>2. El diseñador selecciona la opción que desea realizar.</p>	<p>2.1 El sistema ejecuta la opción que desea el usuario.</p> <ul style="list-style-type: none">  Si selecciona Insertar Pozos, ir a la

sección “Insertar pozo”.

✚ Si selecciona Modificar, ir a la sección “Modificar Pozo”.

✚ Si selecciona eliminar, ir a la sección “Eliminar Pozo”.

Prototipo de Interfaz

The screenshot shows a software window titled "Gestionar Pozo" with three tabs: "Datos Pozo", "Cemento", and "Lechada". The "Datos Pozo" tab is active and contains a form with the following fields:

- Compañía: dropdown menu
- Tapón: dropdown menu
- Nombre: text input
- Díámetro: text input
- BHST: text input
- BHCT: text input
- Presión: text input
- Revestimiento: dropdown menu
- Schedule: dropdown menu
- MD: text input
- TVD: text input
- Inclinación: text input
- Gradiente temp: text input

Below the form are "Aceptar" and "Cancelar" buttons. At the bottom of the window is a table with columns: Nombre, Compañía, Díámetro, MD, BHCT, TVD, BHST. Below the table are "Adicionar", "Modificar", and "Eliminar" buttons.

Flujo Normal de Eventos

Sección 2.1 “Insertar Pozo”

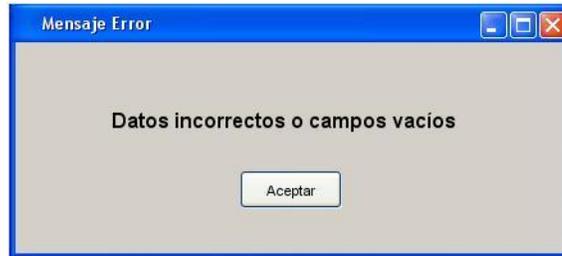
Acción del Actor

Respuesta del Sistema

	<p>1.1 El sistema desbloquea el panel que contiene los datos del pozo.</p>
<p>1. El diseñador introduce los datos y oprime el botón aceptar. Ver flujo alternativo 2</p>	<p>1.1 El sistema verifica que los datos estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 1 sección 1.1</p> <p>1.2 El sistema verifica que el pozo no exista. Ver flujo alternativo 1 sección 1.2</p> <p>1.2 El sistema envía un mensaje que la operación se realizó con éxito y termina el CU.</p>
<p>Prototipo de Interfaz</p>	
<p>Flujo Alterno 1</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>1.1 Si los datos no son correctos o existen campos vacíos se le notifica al usuario.</p>
	<p>1.2 Si ya existe dicho pozo el sistema</p>

notifica al usuario y termina el CU.

Prototipo Interfaz

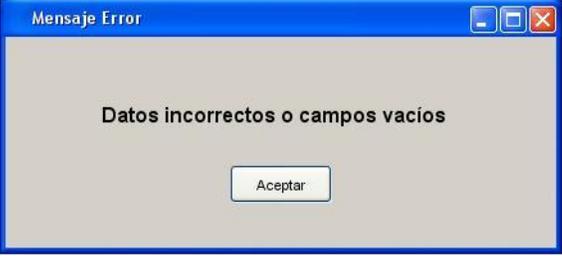


Flujo Alterno 2

Acción del Actor	Respuesta del Sistema
1.1 El diseñador selecciona la opción Cancelar.	1.1 El sistema elimina los datos generales del pozo que se encuentren en los campos y los bloquea nuevamente.

Sección 2.1 “Modificar pozos”

Acción del Actor	Respuesta del Sistema
	1. El sistema desbloquea los campos y los llena con los datos del pozo seleccionado.
2. El diseñador introduce los nuevos datos y acepta. Ver Flujo Alterno 1	2.1 El sistema verifica los datos, y que no existan campos vacíos. Ver Flujo

	Alternativo 2. 2.2 Envía un mensaje confirmando que la acción se realizó con éxito y termina el CU.
Prototipo de Interfaz	
Flujo Alternativo 1	
Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción cancelar	1.1 El sistema elimina los datos del pozo que se encuentren en los campos y los bloquea nuevamente.
Flujo Alternativo 2	
	2.1 Si los datos no son correctos o existen campos vacíos, se le notifica al usuario.
Prototipo Interfaz	
	

Sección “Eliminar Pozos”	
Acción del Actor	Respuesta del Sistema
	1.1 El sistema muestra un mensaje para que el diseñador confirme que desea eliminar los datos del pozo.
2. El diseñador acepta. Ver flujo alternativo 1	2.1 El sistema notifica que se ha realizado la acción y termina el CU.
<i>Prototipo de Interfaz</i>	
Flujo Alterno 1	
Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción cancelar.	1.1 El sistema cancela la eliminación de los datos del pozo y termina el caso de uso.
<i>Prototipo de Interfaz</i>	
Poscondiciones	Se obtiene la realización de cada funcionalidad seleccionada.

Tabla 3: Descripción del Caso de Uso Gestionar Pozos

3.7.2.1.3 Caso de Uso Gestionar Prueba

Caso de Uso:	Gestionar Pruebas
---------------------	--------------------------

Actores:	Diseñador
Resumen:	El CU comienza cuando el diseñador selecciona en la interfaz principal gestionar pruebas para realizar las operaciones que desea realizar, ya sea insertar, eliminar o modificar, y termina cuando el diseñador acepta la operación.
Precondiciones:	Que el usuario se haya autenticado como diseñador
Referencias	R3

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
<p>1. El diseñador selecciona la opción gestionar pruebas, para insertar, eliminar o modificar una prueba.</p>	<p>1.1 El sistema busca todos los datos de las pruebas existentes.</p> <p>1.2 El sistema muestra un formulario con los campos bloqueados y muestra una tabla con los resultados de la búsqueda de las pruebas que se encuentran en la BD. El diseñador podrá en este caso:</p> <ul style="list-style-type: none">  Insertar Prueba  Modificar Prueba  Eliminar Prueba

2. El diseñador selecciona la opción que desea realizar.

1.1 El sistema ejecuta la opción que desea el usuario.

- ✚ Si selecciona Insertar Prueba, ir a la sección “Insertar Prueba”.
- ✚ Si selecciona Modificar Prueba, ir a la sección “Modificar prueba”.
- ✚ Si selecciona Eliminar Prueba, ir a la sección “Eliminar Prueba”.

Prototipo Interfaz

El prototipo de interfaz de usuario muestra una ventana titulada "Frame" con una barra de pestañas que incluye: Datos Prueba, Agua Libre, Reología, Filtrado, Sedimentación, Imepo Frague, Consistencia, Compresión, Estabilidad, Lectura y Gelatinosidad. El formulario principal contiene los siguientes campos:

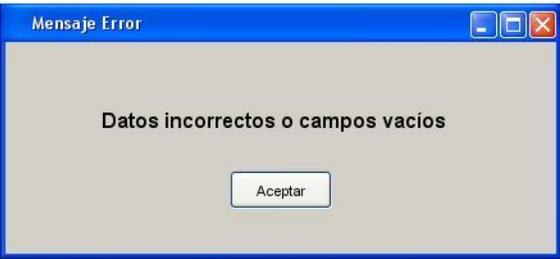
- Pozo: campo de selección con una flecha hacia abajo.
- Nombre: campo de selección con una flecha hacia abajo.
- Etapas: campo de texto.
- Número: campo de texto.
- Realizado Por: campo de texto.
- Comentario: campo de texto grande.

Debajo del formulario hay dos botones: "Aceptar" y "Cancelar".

En la parte inferior, hay una tabla con las siguientes columnas: Nombre, Pozo, Etapas y Realizado Por. Debajo de la tabla hay tres botones: "Adicionar", "Modificar" y "Eliminar".

Flujo Normal de Eventos

Sección 2.1 “Insertar Prueba”	
Acción del Actor	Respuesta del Sistema
	1.1 El sistema desbloquea el panel que contiene los datos de la prueba.
2. El diseñador introduce los datos y oprime el botón aceptar. Ver flujo alternativo 2	<p>2.1 El sistema verifica que los datos estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 1 sección 1.1</p> <p>2.2 El sistema activa el panel del tipo de prueba seleccionada.</p>
3. El diseñador introduce los datos de la prueba y acepta. Ver flujo alternativo 2	<p>3.1 El sistema verifica que los datos estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 1 sección 1.1</p> <p>3.2 El sistema verifica que el pozo no exista. Ver flujo alternativo 1 sección 1.2</p> <p>3.3 El sistema envía un mensaje que la operación se realizó con éxito y termina el CU.</p>
Prototipo de Interfaz	

Flujo Alternativo 1	
Acción del Actor	Respuesta del Sistema
	1.1 Si los datos no son correctos o existen campos vacíos, el sistema notifica al usuario.
	1.2 En caso de que la prueba exista el sistema envía un mensaje de notificación y termina el caso de uso.
Prototipo Interfaz	
	
Flujo Alternativo 2	
1. El diseñador selecciona la opción cancelar	1.1 El sistema elimina los datos generales de la prueba que se encuentren en los campos y los bloquea nuevamente.
Prototipo de Interfaz	

Sección 2.1 “Modificar prueba”	
Acción del Actor	Respuesta del Sistema
	1.1 El sistema desbloquea los campos y los llena con los datos de la prueba seleccionada.
2. El diseñador introduce los nuevos datos y acepta. Ver flujo alternativo 1	2.1 El sistema verifica que los datos estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 2 2.2 El sistema envía un mensaje que la operación se realizó con éxito y termina el CU.
<i>Prototipo de Interfaz</i>	
Flujo Alternativo 1	
Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción cancelar	1.1 El sistema elimina los datos generales de la prueba que se encuentren en los campos y los bloquea nuevamente.
Flujo Alternativo 2	
	1.1 Si los datos no son correctos o

existen campos vacíos, se le notifica al usuario.

Prototipo Interfaz



Sección “Eliminar Prueba”

Acción del Actor	Respuesta del Sistema
	1.1 El sistema muestra un mensaje para que el diseñador confirme que desea eliminar los datos de la prueba
2. El diseñador acepta. Ver flujo alternativo 1	2.1 El sistema notifica que se ha realizado la acción y termina el CU

Prototipo de Interfaz

Flujo Alterno 1

Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción cancelar.	1.1 El sistema cancela la eliminación de los datos de la prueba y termina el

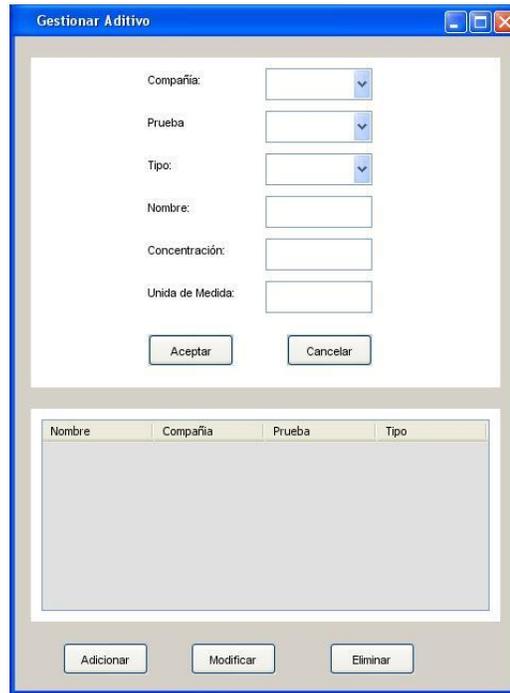
	caso de uso.
Prototipo de Interfaz	
Poscondiciones	Se obtiene la realización de cada funcionalidad seleccionada.

Tabla 4: Descripción del Caso de Uso Gestionar Pruebas

3.7.2.1.4 Caso de Uso Gestionar Aditivo

Caso de Uso:	Gestionar Aditivo
Actores:	Diseñador
Resumen:	El caso de uso inicia cuando el actor solicita gestionar un aditivo, el sistema permite que se inserte el mismo, se modifique o elimine y termina cuando el diseñador acepta y se notifica que se ha realizado la acción.
Precondiciones:	Que el diseñador se haya autenticado.
Referencias	R4
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción gestionar aditivos, para insertar, eliminar o modificar un aditivo.	1.1 El sistema busca todos los datos de los aditivos existentes. 1.2 El sistema muestra un formulario con los campos del

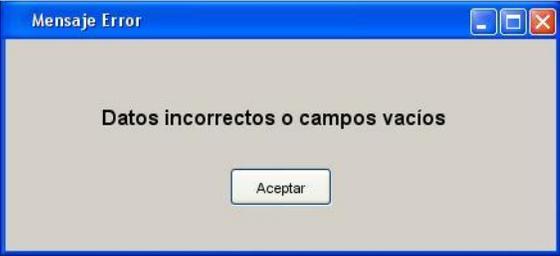
	<p>aditivo bloqueados, además de una tabla con los resultados de la búsqueda realizada. El diseñador podrá en este caso:</p> <ul style="list-style-type: none"> Insertar Aditivo Modificar Aditivo Eliminar Aditivo
<p>2. El diseñador selecciona la opción que desea realizar.</p>	<p>1.1 El sistema ejecuta la opción que desea el usuario.</p> <ul style="list-style-type: none"> Si selecciona Insertar Aditivo, ir a la sección “Insertar Aditivo”. Si selecciona Modificar Aditivo, ir a la sección “Modificar Aditivo”. Si selecciona Eliminar Aditivo, ir a la sección “Eliminar Aditivo”.
<p>Prototipo Interfaz</p>	



Flujo Normal de Eventos

Sección “Insertar Aditivo”

Acción del Actor	Respuesta del Sistema
	1.1 El sistema desbloquea el panel que contiene los datos del aditivo
2. El diseñador introduce los datos y oprime el botón aceptar. Ver flujo alternativo 2	2.1 El sistema verifica que los datos estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 1 2.2 El sistema envía un mensaje que la

	operación se realizó con éxito y termina el CU.
Prototipo de Interfaz	
Flujo Alternativo 1	
Acción del Actor	Respuesta del Sistema
	1.1 Si los datos entrados no son correctos o existen campos vacíos, el sistema envía un mensaje notificando al usuario.
Prototipo Interfaz	
	
Flujo Alternativo 2	
1.1 El diseñador selecciona la opción Cancelar.	1.1 El sistema elimina los datos generales del aditivo que se encuentren en los campos y los bloquea nuevamente.
Prototipo de Interfaz	

Sección “Eliminar Aditivo”	
Acción del Actor	Respuesta del Sistema
	1.1 El sistema muestra un mensaje para que el diseñador confirme que desea eliminar los datos del aditivo.
2. El diseñador acepta. Ver flujo alternativo 1	2.1 El sistema notifica que se ha realizado la acción y termina el CU
<i>Prototipo de Interfaz</i>	
Flujo Alterno 1	
Acción del Actor	Respuesta del Sistema
1. El diseñador selecciona la opción cancelar.	1.1 El sistema cancela la eliminación de los datos del pozo y termina el caso de uso.
<i>Prototipo de Interfaz</i>	
Sección 2.1 “Modificar aditivo”	
Acción del Actor	Respuesta del Sistema
	1. El sistema desbloquea los campos de los datos y los llena con los datos del aditivo seleccionado.

<p>2. El diseñador introduce los nuevos datos y acepta. Ver Flujo Alterno 1</p>	<p>2.1 El sistema verifica los datos, y que no existan campos vacíos. Ver Flujo Alterno 2.</p> <p>2.2 Envía un mensaje confirmando que la acción se realizó con éxito y termina el CU.</p>
<p>Prototipo de Interfaz</p>	
<p>Flujo Alterno 1</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>2. El diseñador selecciona la opción cancelar</p>	<p>1.1 El sistema elimina los datos generales del Proyecto que se encuentren en los campos y los bloquea nuevamente.</p>
<p>Flujo Alterno 2</p>	
	<p>2.1 Si los datos no son correctos o existen campos vacíos, se le notifica al usuario.</p>
<p>Prototipo Interfaz</p>	



Prototipo de Interfaz

Poscondiciones	Se obtiene la realización de cada funcionalidad seleccionada.
-----------------------	--

Tabla 5: Descripción del Caso de Uso Gestionar Aditivo

3.7.2.1.5 Caso de Uso Visualizar Búsquedas

Caso de Uso:	Visualizar Búsquedas
Actores:	Consultor
Resumen:	<p>Este caso de uso inicia cuando el consultor selecciona la opción de ver la información, para esto el sistema muestra las operaciones de:</p> <ul style="list-style-type: none">  Buscar Pozos  Buscar Pruebas  Buscar Aditivo  Buscar Servicios realizados a una compañía. <p>Y finaliza cuando el sistema muestra la información solicitada y el usuario acepta.</p>

Precondiciones:	Que el usuario se autentique.
Referencias	R5
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. El consultor selecciona la opción Gestionar Búsquedas, para buscar un pozo, una prueba, un aditivo o un servicio brindado a una compañía determinada.</p>	<p>1.1 El sistema muestra en un menú las siguientes opciones:</p> <ul style="list-style-type: none">  Buscar Pozos  Buscar Pruebas.  Buscar Aditivos  Buscar servicios realizados a una compañía
<p>2. El consultor selecciona la opción que desea realizar.</p>	<p>1.1 El sistema ejecuta la opción que desea el usuario.</p> <ul style="list-style-type: none">  Si selecciona Buscar Pozos, ir a la sección “Buscar Pozos”.  Si selecciona Buscar Aditivo, ir a la sección “Buscar Aditivo”.  Si selecciona Buscar Prueba, ir a la sección “Buscar Prueba”.

	<p>✚ Si selecciona Buscar Servicios, ir a la sección “Buscar Servicios”.</p>
<p>Sección “Buscar Pozos”</p>	
Acción del Actor	Respuesta del Sistema
	<p>1.1 El sistema muestra un formulario con el o los criterios de búsquedas por los cuales se va a regir el usuario.</p>
<p>2. El consultor selecciona el o los criterios de búsqueda que necesita que cumpla el pozo.</p>	<p>2.1 El sistema verifica que se haya seleccionado al menos un criterio de búsqueda.</p> <p>2.2 El sistema muestra una lista de todos los pozos que cumplen estas condiciones y termina el CU.</p>
<p>Prototipo de Interfaz</p>	



Flujo Alterno

Acción del Actor	Respuesta del Sistema
	2.1 En caso de que el usuario no seleccione ningún criterio de búsqueda, el sistema envía un mensaje de notificación.

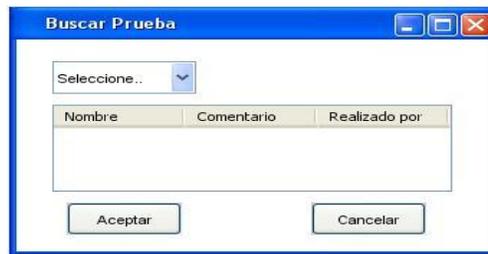
Prototipo de Interfaz

Sección “Buscar Prueba”

Acción del Actor	Respuesta del Sistema
	1.1 El sistema muestra un formulario para que el usuario seleccione el nombre de la prueba.
2. El consultor selecciona la prueba.	2.1 El sistema verifica que se haya seleccionado.

2.2 El sistema muestra una lista de todos los pruebas de ese tipo que se encuentran en la BD y termina el CU.

Prototipo de Interfaz



Flujo Alterno

Acción del Actor

Respuesta del Sistema

2.1 En caso de que el usuario no seleccione ninguna prueba, el sistema envía un mensaje de notificación.

Prototipo de Interfaz

Sección “Buscar Aditivo”

Acción del Actor

Respuesta del Sistema

1.1 El sistema muestra una interfaz para que se seleccione la compañía.

<p>2. El consultor selecciona la compañía</p>	<p>2.1 El sistema verifica los datos. 2.2 El sistema muestra una interfaz con los aditivos que encuentre para dicha compañía y termina el caso de uso.</p>
---	--

Prototipo de Interfaz



Flujo Alternativo

Acción del Actor	Respuesta del Sistema
	<p>2.1 En caso de que los datos no sean correctos, envía un mensaje "No selecciono la compañía"</p>

Prototipo de Interfaz

Sección "Buscar Servicios realizados a una compañía"

Acción del Actor	Respuesta del Sistema
	<p>1.1 El sistema muestra un formulario para que se seleccione la compañía.</p>

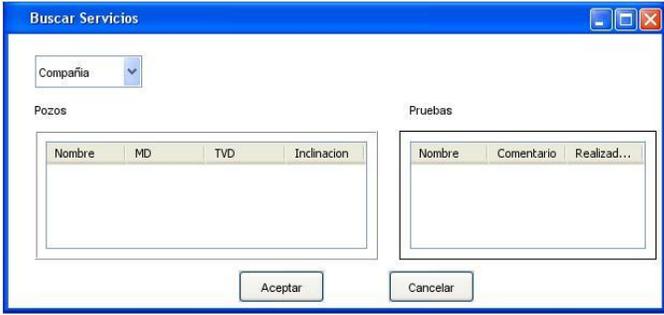
<p>2. El consultor selecciona compañía a la cual se le quiere buscar los trabajos.</p>	<p>2.1 El sistema verifica que se haya seleccionado la compañía.</p> <p>2.2 El sistema muestra una lista de todos los pozos y las pruebas que de esta compañía que se encuentran en la base de datos y termina el CU.</p>
<p>Prototipo de Interfaz</p> 	
<p>Flujo Alternativo</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>2.1 En caso de que el usuario no seleccione ningún criterio de búsqueda, el sistema envía un mensaje de notificación al usuario.</p>
<p>Prototipo de Interfaz</p>	
<p>Poscondiciones</p>	

Tabla 6: Descripción del Caso de Uso Visualizar Búsquedas

3.7.2.1.6 Caso de Uso Generar Reporte

Caso de Uso:	Generar Reportes
Actores:	Consultor
Resumen:	El CU inicia cuando el consultor solicita generar un reporte de la lechada que acaba de diseñar, luego el sistema muestra los datos y permite imprimirlo o no en dependencia de lo que el usuario quiera.
Precondiciones:	Que se haya insertado un nuevo pozo al cual se le haya diseñado una lechada
Referencias	R6
Prioridad	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El consultor selecciona generar reporte	1.1 El sistema muestra una interfaz para que se seleccione el pozo al que se le quiere realizar el reporte.
2. El consultor selecciona el pozo.	2.1 El sistema muestra un formulario con los datos que debe llevar el reporte.
3. El consultor selecciona la opción del exportar a pdf.	3.1 El sistema exporta los datos del reporte a pdf.

Prototipo de Interfaz



Flujos Alternos

Acción del Actor

Respuesta del Sistema

Prototipo de Interfaz

Poscondiciones

Permite imprimir dicho reporte.

Tabla 7: Descripción del Caso de Uso Gestionar Reportes

3.7.2.1.2 Caso de Uso Gestionar Usuario

Caso de Uso:	Gestionar Usuario
Actores:	Administrador
Resumen:	El CU comienza cuando el diseñador selecciona la opción de gestionar un usuario, donde podrá insertar, eliminar o modificar un usuario, y termina cuando acepta la opción seleccionada.

Precondiciones:	Que el usuario se haya autenticado como administrador.
Referencias	R7
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
<p>1 El administrador selecciona la opción gestionar usuario, para insertar, eliminar o modificarlo.</p>	<p>1.1 El sistema busca todos los datos de los usuarios existentes.</p> <p>1.2 El sistema muestra un formulario con los campos bloqueados, además de una tabla con los datos de los pozos existentes. El administrador podrá en este caso:</p> <ul style="list-style-type: none">  Insertar Usuario  Modificar Usuario  Eliminar Usuario
<p>2 El administrador selecciona la opción que desea realizar.</p>	<p>2.1 El sistema ejecuta la opción que desea el usuario.</p> <ul style="list-style-type: none">  Si selecciona Insertar Usuario, ir a la sección “Insertar usuario”.  Si selecciona Modificar, ir a la sección “Modificar usuario”.

Si selecciona eliminar, ir a la sección “Eliminar usuario”.

Prototipo Interfaz



Flujo Normal de Eventos

Sección 2.1 “Insertar Usuario”

Acción del Actor	Respuesta del Sistema
	1.1 El sistema desbloquea el panel que contiene los datos del usuario.
2. El administrador introduce los datos y oprime el	2.1 El sistema verifica que los datos

<p>botón aceptar. Ver flujo alternativo 2</p>	<p>estén correctos o que no hayan campos vacíos. Ver Flujo alternativo 1 sección 1.1</p> <p>1.2 El sistema verifica que el usuario no exista. Ver flujo alternativo 1 sección 1.2</p> <p>2.2 El sistema envía un mensaje que la operación se realizó con éxito y termina el CU.</p>
<p>Prototipo de Interfaz</p>	
<p>Flujo Alterno 1</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
	<p>1.1 Si los datos no son correctos o existen campos vacíos, se le notifica al usuario.</p>
	<p>1.2 En caso que ya existe dicho usuario, el sistema envía un mensaje de notificación.</p>
<p>Prototipo Interfaz</p>	



Flujo Alterno 2

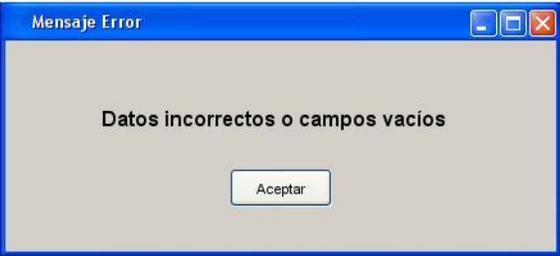
1.1 El administrador selecciona la opción Cancelar.

1.1 El sistema elimina los datos generales del usuario que se encuentren en los campos y los bloquea nuevamente.

Prototipo de Interfaz

Sección 2.1 “Modificar usuario”

Acción del Actor	Respuesta del Sistema
	1.1 El sistema desbloquea los campos y los llena con los datos del usuario seleccionado
2. El administrador introduce los nuevos datos y acepta. Ver Flujo Alterno 1	2.1 El sistema verifica los datos, y que no existan campos vacíos. Ver Flujo Alterno 2. 2.2 Envía un mensaje confirmando que la acción se realizó con éxito y termina

		el CU.
Prototipo de Interfaz		
Flujo Alternativo 1		
Acción del Actor	Respuesta del Sistema	
1. El administrador selecciona la opción cancelar	1.1 El sistema elimina los datos generales del aditivo que se encuentren en los campos y los bloquea nuevamente.	
Flujo Alternativo 2		
	2.1 Si los datos no son correctos o existen campos vacíos, se le notifica al usuario.	
Prototipo Interfaz		
		
Sección "Eliminar Usuario"		
Acción del Actor	Respuesta del Sistema	

	1.1 El sistema muestra un mensaje para que el diseñador confirme que desea eliminar los datos del usuario.
2. El administrador acepta. Ver flujo alternativo 1	2.1 El sistema notifica que se ha realizado la acción y termina el CU
<i>Prototipo de Interfaz</i>	
Flujo Alterno	
Acción del Actor	Respuesta del Sistema
1. El administrador selecciona la opción cancelar.	1.1 El sistema cancela la eliminación de los datos del usuario y termina el caso de uso.
<i>Prototipo de Interfaz</i>	
Poscondiciones	Se obtiene la realización de cada funcionalidad seleccionada.

Tabla 8: Descripción del Caso de Uso Gestionar Usuario

3.7.3 Diagrama de Casos de Usos del Sistema

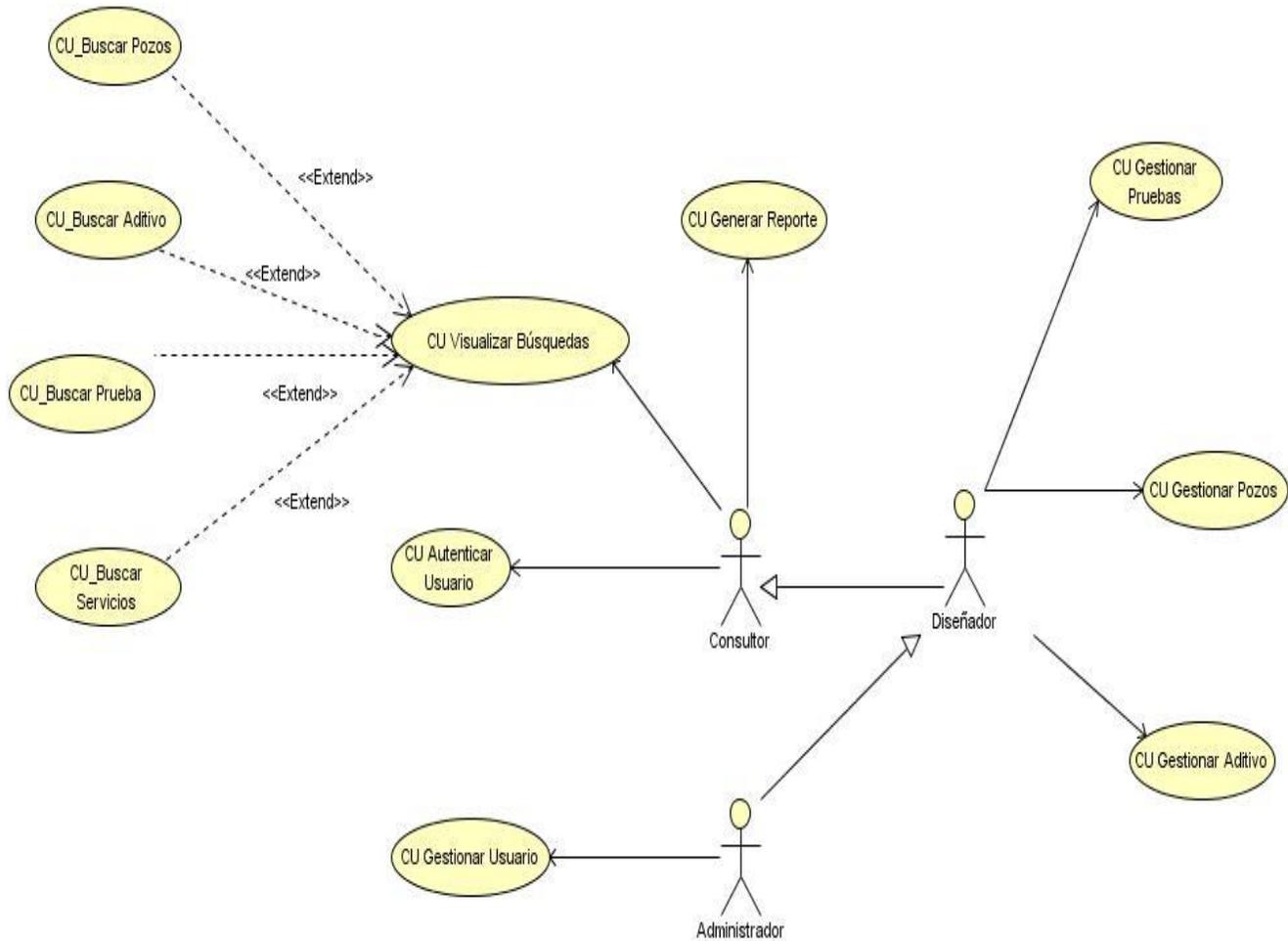


Figura 10: Diagrama de Casos de Usos del Sistema

3.8 Conclusiones Parciales

En este capítulo se hizo una presentación del dominio del problema, por lo que se identificaron las clases del dominio, dando una idea de cómo se comportan estos procesos en la actualidad en el laboratorio de lodos y cemento del Centro de Investigaciones del Petróleo, se expusieron los requerimientos funcionales y no funcionales del prototipo funcional a desarrollar y se modeló este en términos de casos de uso de sistema.

“Nunca diseño un edificio sin haber visto primero el sitio y haber conocido quien lo va a utilizar”

Frank Lloyd Wright

CAPÍTULO 4 “Construcción de la solución propuesta”

4.1 Introducción

En este capítulo se expondrán los elementos más importantes del análisis y el diseño del sistema que se presenta, en términos de clases del análisis, de diseño y diagramas que representarán de una mejor forma lo que se verá en el capítulo. Además del diseño de la base de datos y algunas generalidades de la implementación de dicho sistema.

4.2 Diagramas de Clases

Las clases del análisis representan una abstracción de una o varias clases y subsistemas del diseño del sistema que posee las siguientes características:

- ✚ Se centra en el tratamiento de los requisitos funcionales.
- ✚ Define atributos que son conceptuales y reconocibles en el dominio del problema.
- ✚ Esto hace que una clase del análisis sea más evidente en el contexto del dominio del problema.
- ✚ La clase del análisis encaja en una de estos tres estereotipos: interfaz, control, entidad.

Una clase de análisis participa en varias realizaciones de CU y algunas responsabilidades, atributos y asociaciones suelen ser sólo relevantes para una única realización de CU, por tanto el Diagrama de clases del análisis es un artefacto en el que se coordinan todos los requisitos sobre una clase y sus objetos, los cuales pueden tener diferentes casos de usos. Representa las cosas del mundo real, no de la implementación automatizada de estas cosas. **(Robles, 2007)**

[Ver Anexo1](#)

4.3 Diagramas de Interacción.

Los diagramas de interacción son diagramas que modelan aspectos dinámicos del sistema.

Un diagrama de Colaboración es una forma alternativa al diagrama de secuencia de mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas en torno a los objetos y los enlaces entre ellos. **(Rumbaugh, et al., 2000)**

[Ver Anexo 2](#)

4.4 Principios de diseño

El diseño se modela precisamente de forma tal que soporte los requisitos funcionales del sistema, se crean los diagramas de diseño basados en las generalidades de los diagramas de análisis realizados previamente y es en estos diagramas justamente donde se incorporan los elementos propios del lenguaje de programación utilizado, la arquitectura y los patrones a utilizar.

4.4.1 Diagrama de clases del diseño.

Un diagrama de clases del diseño es una representación más concreta que el diagrama de clases del análisis. Representa la parte estática del sistema así como las clases y sus relaciones.

[Ver Anexo 3](#)

4.4.2 Patrones de diseño.

Un patrón de diseño es una abstracción de una solución en un nivel alto. Los patrones solucionan problemas que existen en muchos niveles de abstracción.

Los patrones de diseño constituyen soluciones exitosas a problemas comunes que se presenten durante el diseño de una aplicación. Evitan cometer errores, son soluciones concretas a problemas frecuentes que se presenten, están basados en la experiencia acumulada de expertos de desarrollo de software. Entre los patrones de diseño se pueden encontrar los patrones surgidos por la banda de los GoF y los patrones de

asignación de responsabilidades Grasp. A continuación una breve descripción de los utilizados en el desarrollo del sistema:

Dentro de los patrones de diseño utilizados en la modelación de este sistema, destacan algunos patrones de diseño GoF (Gans of Four, Grupo de los Cuatro), como es el Patrón Factoría Abstracta y el Singleton. También se usó Patrones Estructurales como el Patrón Fachada. Dentro de los Patrones de diseño GRASP (patrones generales de software para asignar responsabilidades) que se usaron están el experto, creador, alta cohesión, bajo acoplamiento y controlador. Hibernate como framework ORM para el trabajo con los datos contiene patrones propios como es el patrón Dao (Data Access Object).

4.5 Diseño de la Base de Datos

El diseño de la Base de Datos es de vital importancia, en ella se almacena toda la información de los datos que son necesarios para darle solución al problema existente. Permite obtener toda la información que se quiera recuperar del sistema.

El diseño de la base de Datos se obtiene de las clases del diseño y de las realizaciones de los casos de uso. El principal artefacto de esta actividad es el Modelo de Datos que describe la representación lógica y física de los datos persistentes.

4.5.1 Modelo lógico de datos. Diagrama de clases persistentes.

Las clases persistentes con aquellas que mantienen su valor a través del tiempo. El diagrama de clases persistentes expresa los aspectos relacionados con el almacenamiento de datos del sistema. La información con la que el sistema trabaja es necesario guardarla en un medio permanente, es por eso que utilizamos las clases persistentes ya que estas proporcionan un almacenamiento físico seguro de la información, protegiendo la misma en caso de fallo del sistema, evitando la pérdida de la información. Mediante este diagrama se establece el intercambio de información entre el usuario, el sistema y el almacenamiento de los datos.

A continuación se muestra el Diagrama de Clases Persistentes:

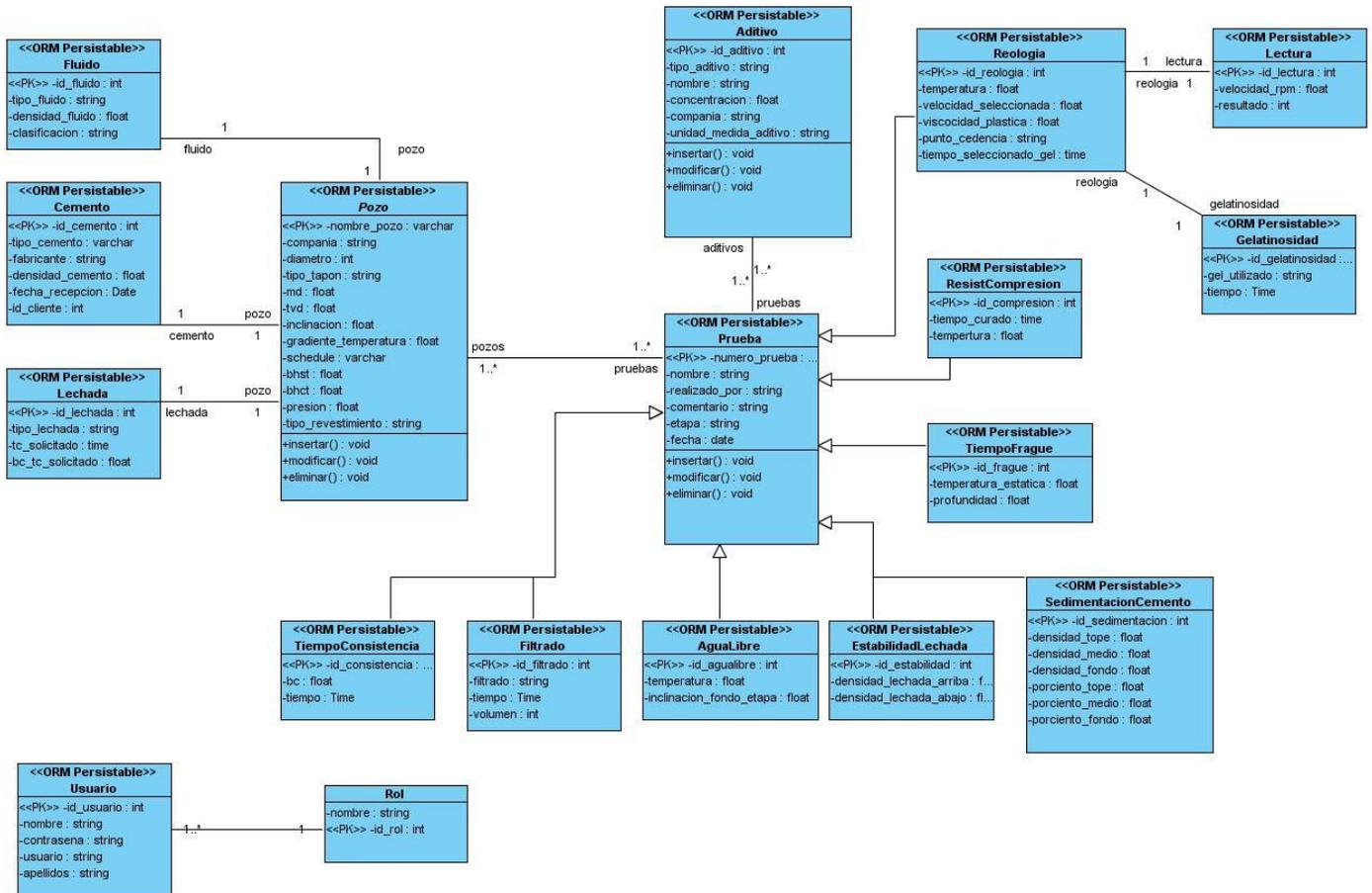


Figura 11: Diagrama de Clases Persistentes

4.5.2 Modelo físico de datos. Diagrama Entidad-Relación.

El Modelo de Datos es un conjunto de conceptos, reglas y convenciones que permiten describir y manipular los datos de un cierto mundo real que se desea almacenar en la Base de Datos (Jacobson y otros, 2000)

Este se genera a partir del diagrama de clases persistentes y muestra la relación existente entre las tablas que tendrá la BD que queremos construir. A continuación se muestra el Diagrama Entidad-Relación que cuenta con 28 tablas:

accedida directamente por las clases XML generadas con el Hibernate, así el resto de los componentes visuales acceden a la BD mediante clases.java, nunca va a ser directamente con la base de datos.

4.6.2 Descripción de los principales algoritmos a implementar.

Dentro de los principales algoritmos que se implementarán es el de Insertar Datos de pruebas, pozos, aditivos y otros.

En la inserción de los datos de los pozos, estarán presentes datos como el nombre, el diámetro, el Schedule utilizado para el diseño, la profundidad total y por la vertical del mismo y otros.

Para esta inserción el usuario pide gestionar un pozo, y el sistema le muestra la interfaz para la entrada de los datos, luego de que el usuario los entra y acepta el sistema guarda dichos datos en la BD.

Para la inserción de los aditivos y las pruebas se realiza el mismo procedimiento pero con los datos

4.7 Modelo de Despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos). Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes software, objetos, procesos (caso particular de un objeto). En general un nodo será una unidad de computación de algún tipo, desde un sensor a un mainframe. Las instancias de componentes software pueden estar unidas por relaciones de dependencia, posiblemente a interfaces (ya que un componente puede tener más de una interfaz).

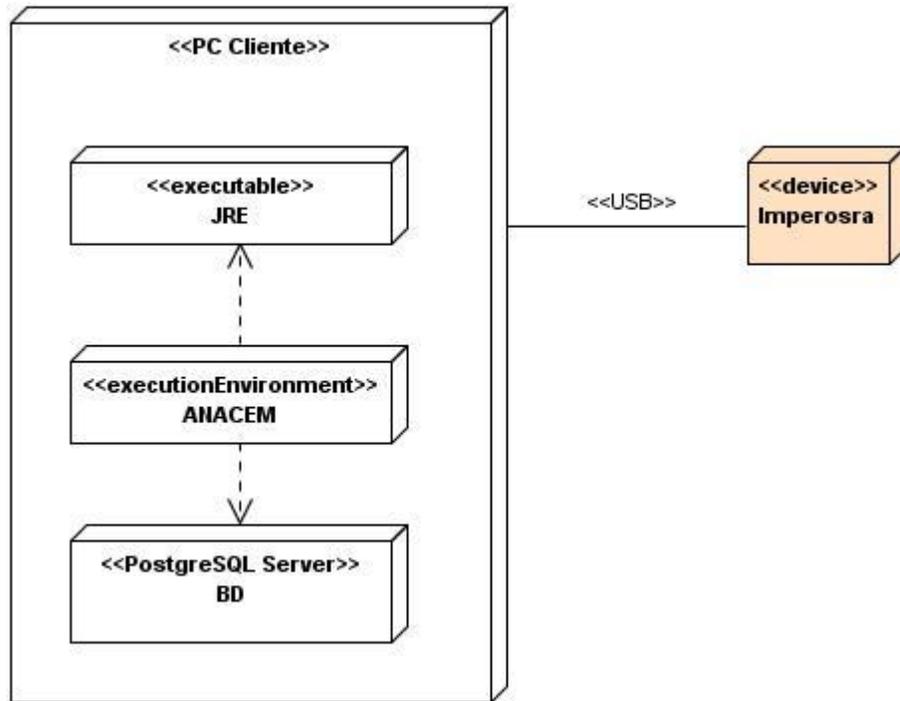


Figura 13: Modelo de Despliegue

4.8 Modelo de Implementación

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes se encuentran datos, archivos, ejecutables, código fuente y los directorios.

Los artefactos más importantes de este Flujo de Trabajo son los diagramas de despliegue y de componentes. Se describen los componentes a construir y la organización y dependencia entre los nodos físicos con los que funcionará la aplicación. El lenguaje de programación es un elemento imprescindible en este Flujo de Trabajo.

4.8.1 Diagrama de Componentes

Componente: Parte modular de un sistema, desplegable y reemplazable que encapsula implementación y un conjunto de interfaces y proporciona la realización de los mismos. Un componente software puede ser desde una subrutina de una librería matemática, hasta una clase, una base de datos, un archivo DLL, un JavaBeans, o incluso una aplicación que pueda ser usada por otra aplicación por medio de una interfaz especificada. **(Teleformacion.uci.cu, 2008)**

El Diagrama de Componentes propuesto para la solución del software, los componentes no son más que el empaquetamiento físico de los elementos de un modelo, como el Modelo de Diseño, donde presentan varios estereotipos como ficheros ejecutables, ficheros de código fuente, entre otros.

 <<executable>> es un programa que puede ser ejecutado en un nodo.

 <<file>> es un fichero que contiene código fuente o datos.

 <<library>> es una librería estática o dinámica.

 <<table>> es una tabla de base de datos.

 <<document>> es un documento.

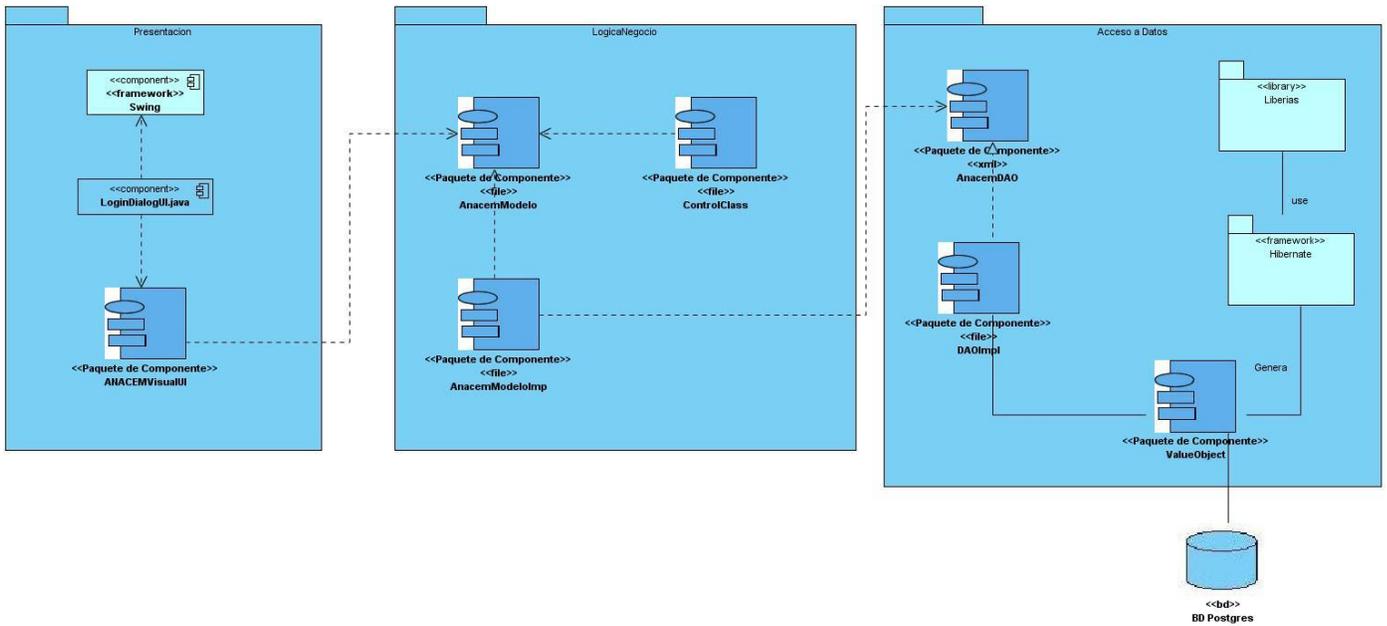


Figura 14: Diagrama de Componentes

4.9 Conclusiones Parciales.

En este capítulo se desarrollaron los diferentes diagramas de clases del análisis y del diseño, también se muestran los diferentes diagramas de interacción correspondientes a los distintos Casos de Uso del Sistema que conforman la aplicación. Se desarrolló un diseño de la base de datos del sistema así como los diferentes diagramas de componentes propios del sistema. Haciendo además una explicación de los principios de implementación del mismo.

CONCLUSIONES GENERALES

En el presente trabajo de diploma quedo demostrado la necesidad de implementar un sistema informático que ayudara a la empresa CEINPET en el diseño de las lechadas de cementación que diariamente se realizan en el laboratorio.

Con el desarrollo de la investigación se realizó un estudio sobre la problemática existente actualmente en el CEINPET, realizando una investigación del tema, así como de los lenguajes de programación, herramientas, y técnicas y tecnologías más factibles para la implementación de dicho sistema.

Para realizar este trabajo fue necesario darle solución a un grupo de problemas relacionados con la organización del proceso, siguiendo una serie de pasos definidos por los estándares de diseño y los pasos propuesto por la metodología RUP que fue la escogida para este trabajo, usando el Visual Paradigm como herramienta para generar los artefactos propuesto por esta metodología usando UML como lenguaje de modelado.

Para brindar una adecuada seguridad en el almacenamiento de la información se realizó el diseño de las clases persistentes y de la base de datos y junto con ellos se seleccionó el framework adecuado para facilitar el acceso y la persistencia de los mismos.

De esta forma se le da cumplimiento a los objetivos trazados para la elaboración del trabajo de diploma, obteniendo la implementación de un sistema informático que permitirá darle solución a la mayoría de los procesos de gestión de lechadas de cementación en el CEINPET.

RECOMENDACIONES

Luego de terminado el presente trabajo de diploma y haber estudiado el proceso de diseño de lechadas de cementación y haber implementado el sistema ANACEM, se recomienda lo siguiente:

- ✚ Analizar la propuesta de diseño y la arquitectura con el fin de lograr una integración con los otros módulos de servicios a pozos petroleros que se implementaron en el polo y realizar una base de Datos centralizada para esto.
- ✚ Continuar la investigación y el levantamiento de requisitos sobre lechadas de cementación para lograr mejoras en las futuras versiones del sistema.
- ✚ Implementar las funcionalidades de Crear Backup y Restablecer la BD desde el sistema para una mejor protección de los datos ante cualquier desastre que pueda ocurrir.
- ✚ Realizar la explotación del sistema en el laboratorio de lodos y cementos del CEINPET.

REFERENCIAS BIBLIOGRÁFICAS

Babylon.com. Definición de Sistema Informático. [En línea]

<http://diccionario.babylon.com/sistema%20inform%C3%A1tico>.

—. **2007.** Glosario de Informática. [En línea] 2007. <http://diccionario.babylon.com/JAVA> .

Bjarne, Stroustrup. 2006. C++. [En línea] 2006. <http://www.pergaminovirtual.com.ar/definicion/C.html>.

CIED. 2003. *Programa de inducción en ingeniería de perforación*. s.l. : CIED, 2003.

Definicion.de. 2008. Concepto de Gestión. [En línea] 2008. <http://definicion.de/gestion>.

Eterovics, Yadran. 2008. El proceso Unificado (RUP): Técnicas para desarrollar aplicaciones. [En línea] 2008. <http://www2.ing.puc.cl/~iic3194/rupppt.pdf>.

Ezpinosa, Humberto. 2005. PostgreSQL, una alternativa de DBMS Open Source. [En línea] 2005.

Girón, Marco A. Pereira. 2007. Introducción a la Programación Plataforma Windows. UML. [En línea] 2007. <http://www.scribd.com/doc/2080534/UML>.

Hard-h2o.com. 2007. Diccionario Informático. [En línea] 2007. <http://www.hard-h2o.com/diccionario-informatico.html>.

Hernández, Arianna García. 2008. Definición de sistema. *Tecnológico*. [En línea] 2008. <http://www.mitecnologico.com/Main/DefinicionDeSistema>.

Ibm.com. 2006. Rational Rose Enterprise. [En línea] 2006. http://www-142.ibm.com/software/dre/ecatalog/detail.wss?locale=es_ES&synkey=M221280M46834Z27.

Ideasafines.com. 2008. Ideas afines y palabras relacionadas con cemento. [En línea] 2008. <http://www.ideasafines.com.ar/palabras-relacionadas-cemento.php?palabra=cemento>.

- Jacobson, Ivar, Rumbaugh, James y Booch, Grady. 2000.** *El lenguaje unificado de modelado*. s.l. : Addison-Wesley, 2000.
- Larman, Craig. 1999.** *UML y Patrones: Introducción al análisis y diseño orientado a objeto*. México : Prentice Hall, 1999.
- Netpecos.org. 2007.** MySQL. [En línea] Marzo de 2007.
http://www.netpecos.org/docs/mysql_postgres/x57.html.
- Reduser.com. 2005.** Software libre para sitios web dinámicos. [En línea] 2005.
<http://img.redusers.com/imagenes/libros/lpcu068/capitulogratis.pdf>.
- Robles, E. (2007).** *El flujo de Analisis*. Obtenido de <http://www.slideshare.net/juliopari/13-clase-flujo-de-analisis>
- Rumbaugh, Jacobson y Booch. 2000.** *El proceso Unificado de Software*. s.l. : Addison-Wesley, 2000.
- Sánchez, Sergio Ríos. 2007.** Análisis UML. Modelo Conceptual. [En línea] 2007.
http://www.uvmsf.cl/~ssanchez/images/Metodologias/Unidad4_MAD.pdf.
- Sentidowbe.com. 2004.** PhpPgAdmin: administra PostgreSQL desde la web. [En línea] 2004.
<http://sentidoweb.com/2007/03/29/phppgadmin-administra-postgresql-desde-la-web.php>.
- Teleformacion.uci.cu. (2008).** *Sitio de la Asignatura Ingeniería de Software 2. Conferencia4. FT Implementación*. Obtenido de <http://teleformacion.uci.cu/mod/resource/view.php?id=22199>
- Villa, Hugo. 2008.** Técnicas de cementación de pozos de gas con bajo gradiente de fractura en los campos de San Joaquín y Santa Rosa. PDVSA. [En línea] 2008. [Citado el: 5 de Diciembre de 2008.]
<http://www.seflucempo.com/common/pdf/TECNICAS1.pdf>.
- VP.com. 2007.** Visual Paradigm SDE 2.0 for Eclipse. [En línea] 2007. <http://www.visual-paradigm.com/sdeec.php>.

BIBLIOGRAFÍA

Babylon.com. Definición de Sistema Informático. [En línea]

<http://diccionario.babylon.com/sistema%20inform%C3%A1tico>.

Babylon.com. 2007. Glosario de Informática. [En línea] 2007. <http://diccionario.babylon.com/JAVA> .

Bjarne, Stroustrup. 2006. C++. [En línea] 2006. <http://www.pergaminovirtual.com.ar/definicion/C.html>.

CIED. 2003. *Programa de inducción en ingeniería de perforación*. s.l. : CIED, 2003.

Definicion.de. 2008. Concepto de Gestión. [En línea] 2008. <http://definicion.de/gestion>.

Eterovics, Yadran. 2008. El proceso Unificado (RUP): Técnicas para desarrollar aplicaciones. [En línea] 2008. <http://www2.ing.puc.cl/~iic3194/rupppt.pdf>.

Ezpinosa, Humberto. 2005. PostgreSQL, una alternativa de DBMS Open Source. [En línea] 2005.

Girón, Marco A. Pereira. 2007. Introducción a la Programación Plataforma Windows. UML. [En línea] 2007. <http://www.scribd.com/doc/2080534/UML>.

Hard-h2o.com. 2007. Diccionario Informático. [En línea] 2007. <http://www.hard-h2o.com/diccionario-informatico.html>.

Hernández, Arianna García. 2008. Definición de sistema. *Tecnológico*. [En línea] 2008. <http://www.mitecnologico.com/Main/DefinicionDeSistema>.

Ibm.com. 2006. Rational Rose Enterprise. [En línea] 2006. http://www-142.ibm.com/software/dre/ecatalog/detail.wss?locale=es_ES&synkey=M221280M46834Z27.

Ideasafines.com. 2008. Ideas afines y palabras relacionadas con cemento. [En línea] 2008. <http://www.ideasafines.com.ar/palabras-relacionadas-cemento.php?palabra=cemento>.

Jacobson, Ivar, Rumbaugh, James y Booch, Grady. 2000. *El lenguaje unificado de modelado*. s.l. : Addison-Wesley, 2000.

Larman, Craig. 1999. *UML y Patrones: Introducción al análisis y diseño orientado a objeto*. México : Prentice Hall, 1999.

Netpecos.org. 2007. MySQL. [En línea] Marzo de 2007.
http://www.netpecos.org/docs/mysql_postgres/x57.html.

Reduser.com. 2005. Software libre para sitios web dinámicos. [En línea] 2005.
<http://img.redusers.com/imagenes/libros/lpcu068/capitulogratis.pdf>.

Rumbaugh, Jacobson y Booch. 2000. *El proceso Unificado de Software*. s.l. : Addison-Wesley, 2000.

Sánchez, Sergio Ríos. 2007. Análisis UML. Modelo Conceptual. [En línea] 2007.
http://www.uvmsf.cl/~ssanchez/images/Methodologies/Unidad4_MAD.pdf.

Sentidowbe.com. 2004. PhpPgAdmin: administra PostgreSQL desde la web. [En línea] 2004.
<http://sentidoweb.com/2007/03/29/phppgadmin-administra-postgresql-desde-la-web.php>.

Teleformacion.uci.cu. (2008). *Sitio de la Asignatura Ingeniería de Software 2. Conferencia4. FT Implementación*. Obtenido de <http://teleformacion.uci.cu/mod/resource/view.php?id=22199>

Villa, Hugo. 2008. Técnicas de cementación de pozos de gas con bajo gradiente de fractura en los campos de San Joaquín y Santa Rosa. PDVSA. [En línea] 2008. [Citado el: 5 de Diciembre de 2008.]
<http://www.seflucentpo.com/common/pdf/TECNICAS1.pdf>.

VP.com. 2007. Visual Paradigm SDE 2.0 for Eclipse. [En línea] 2007. <http://www.visual-paradigm.com/sdeec.php>.

ANEXOS.

Anexo 1: Diagrama de Clases del análisis.

+ Diagrama de CA CU Autenticar Usuario.

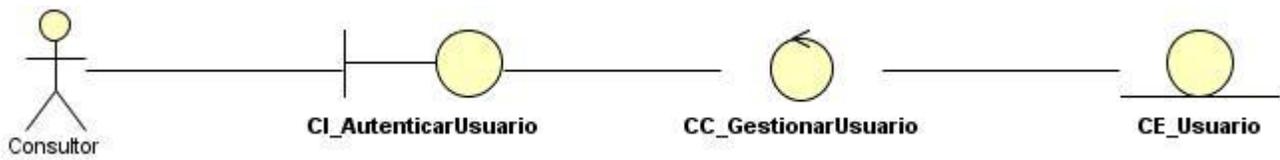


Figura 15: Diagrama de CA CU Autenticar Usuario

+ Diagrama de CA CU Gestionar Usuario.

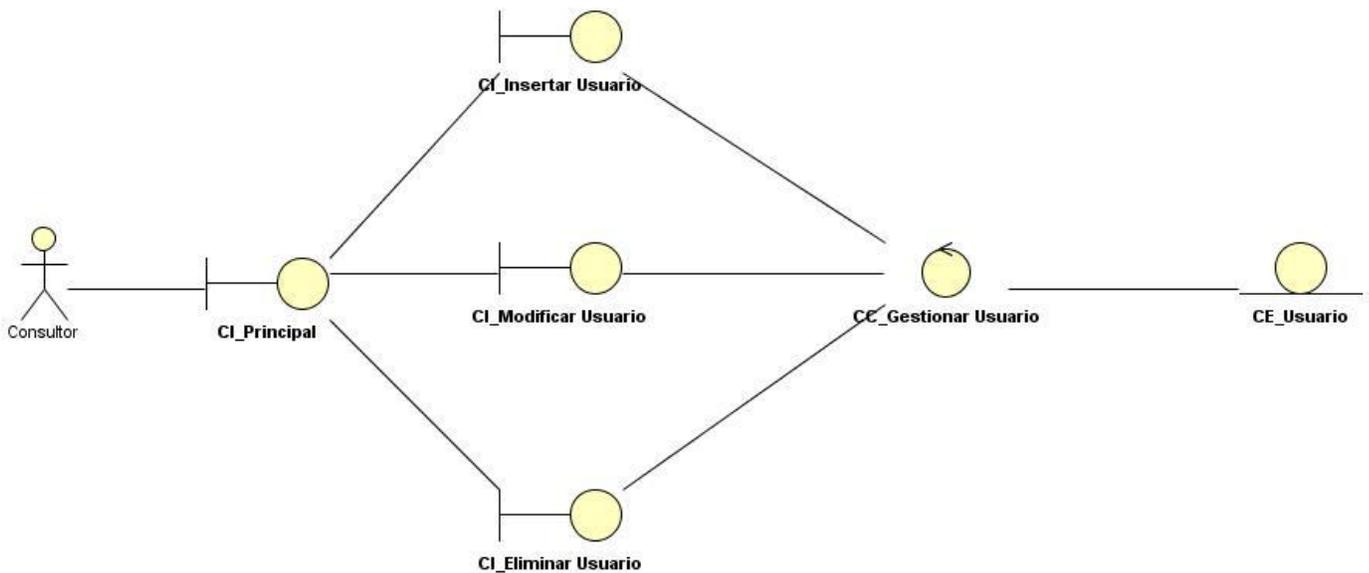


Figura 16: Diagrama de CA CU Gestionar Usuario

✚ Diagrama de CA CU Gestionar Pozo.

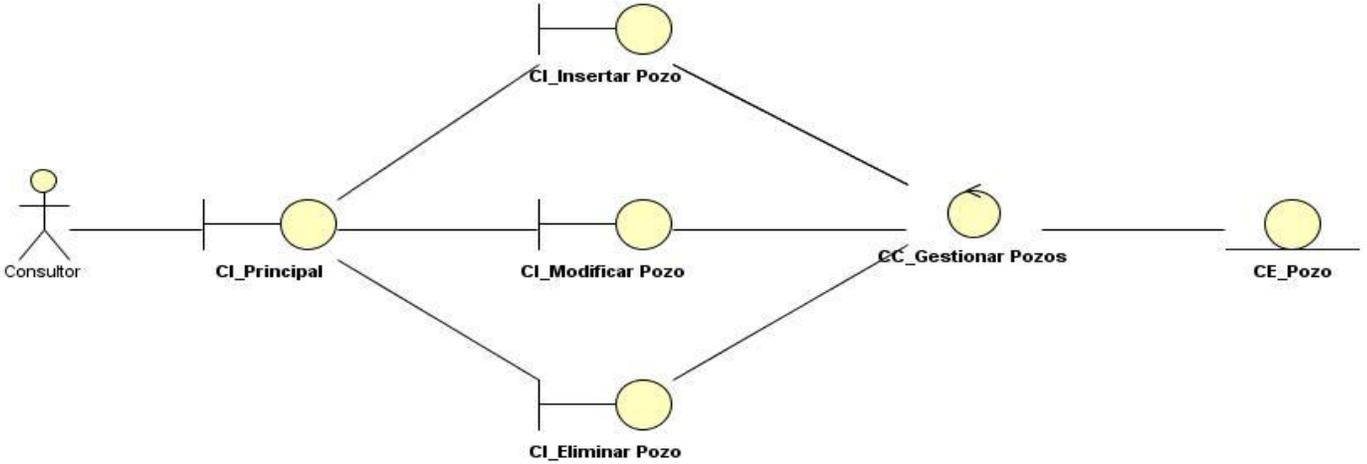


Figura 17: Diagrama de CA CU Gestionar Pozo

✚ Diagrama de CA CU Gestionar Prueba.

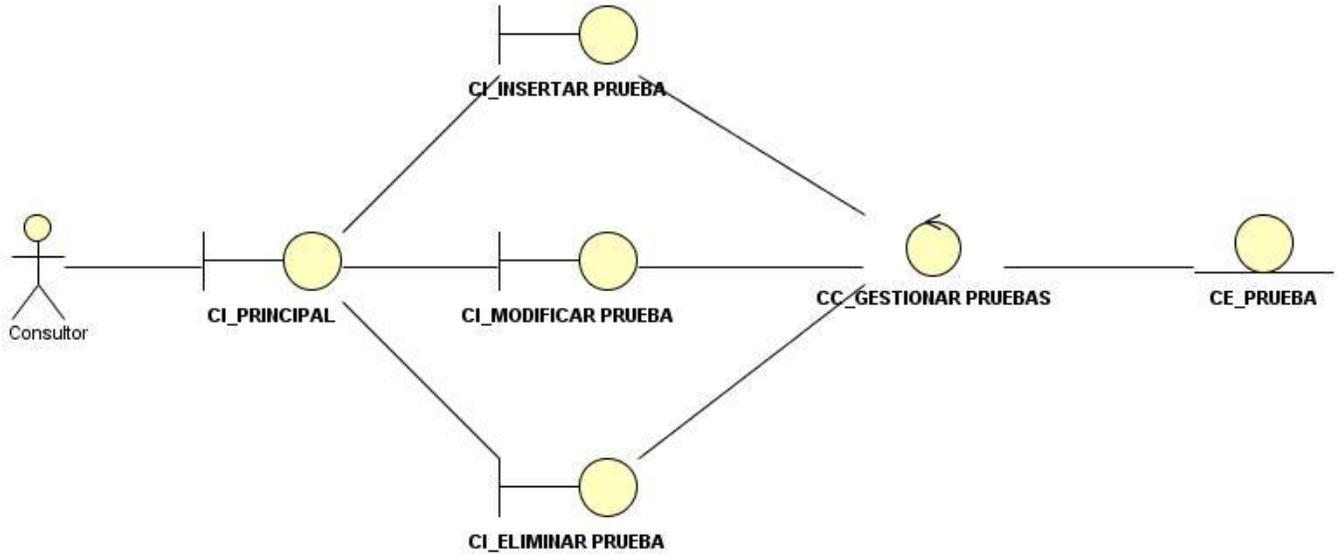


Figura 18: Diagrama de CA CU Gestionar Prueba

✚ Diagrama de CA CU Gestionar Aditivo.

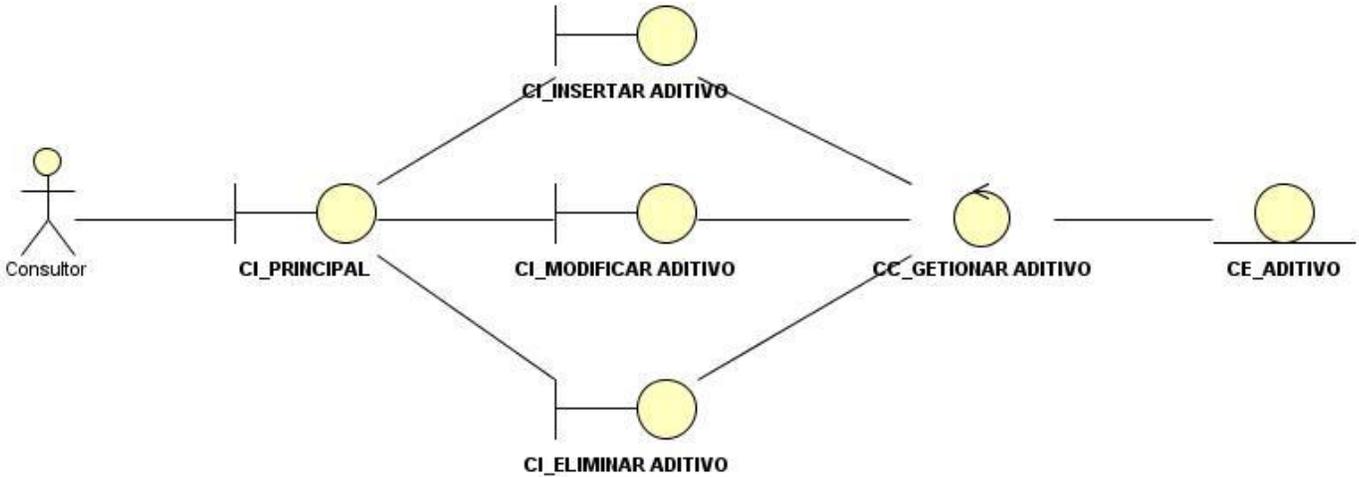


Figura 19: Diagrama de CA CU Gestionar Aditivo

✚ Diagrama de CA CU Visualizar Información.

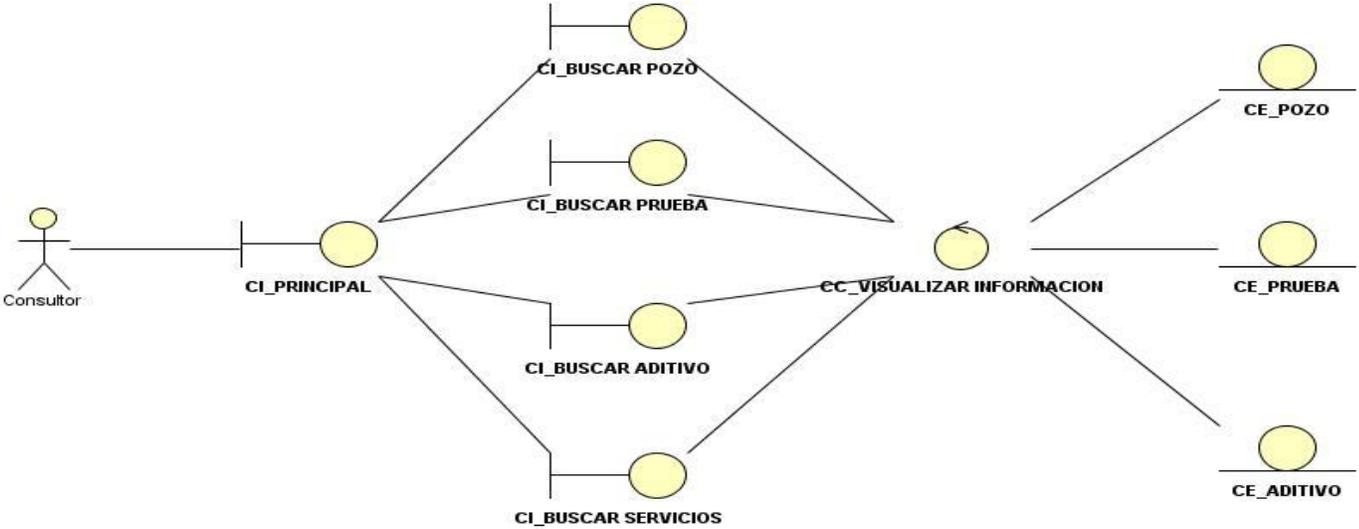


Figura 20: Diagrama de CA CU Visualizar Información

✚ Diagrama de CA CU Generar Reporte.

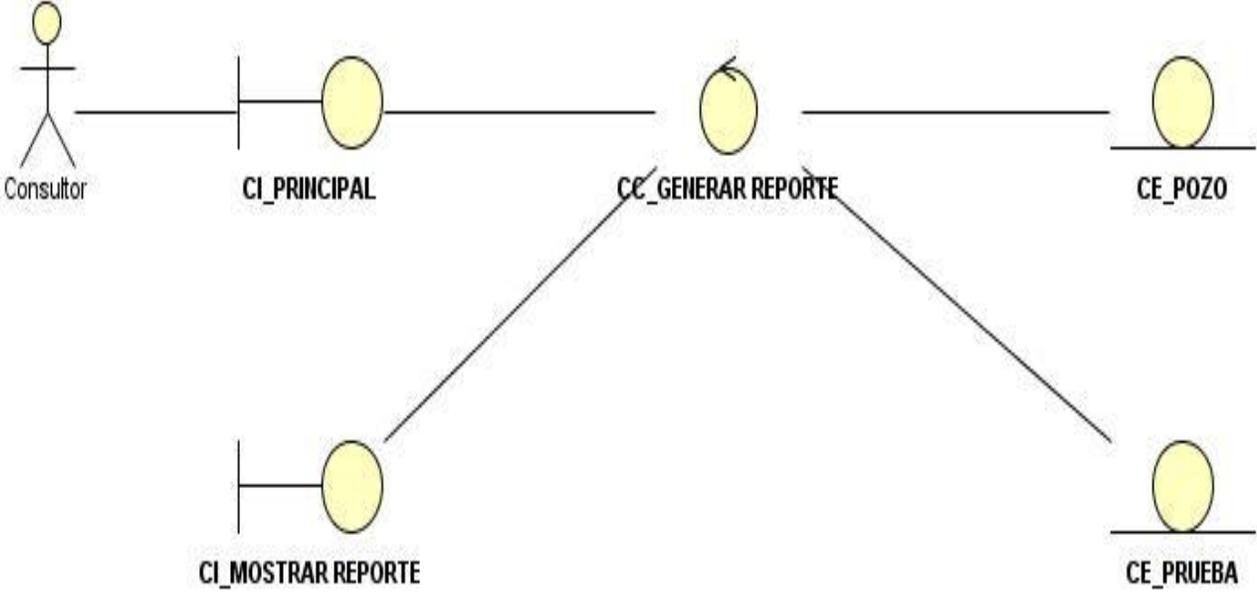


Figura 21: Diagrama de CA CU Generar Reporte

Anexo 2: Diagramas de Colaboración.

Diagrama de colaboración CU Autenticar Usuario.

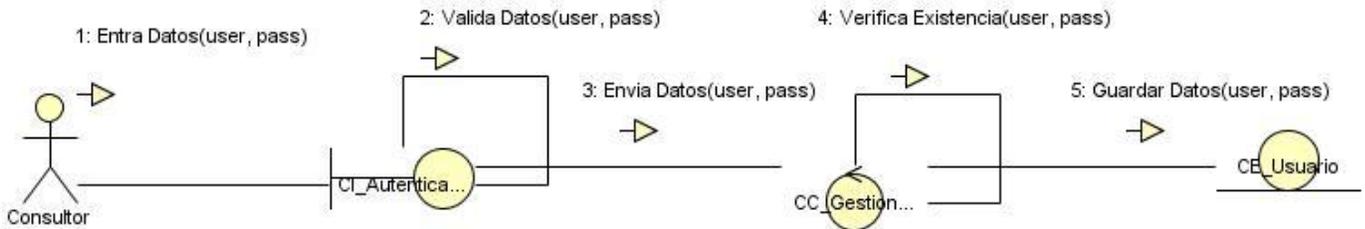


Figura 22: DC CU Autenticar Usuario

Diagrama de colaboración CU Gestionar Pozo Sección Insertar.

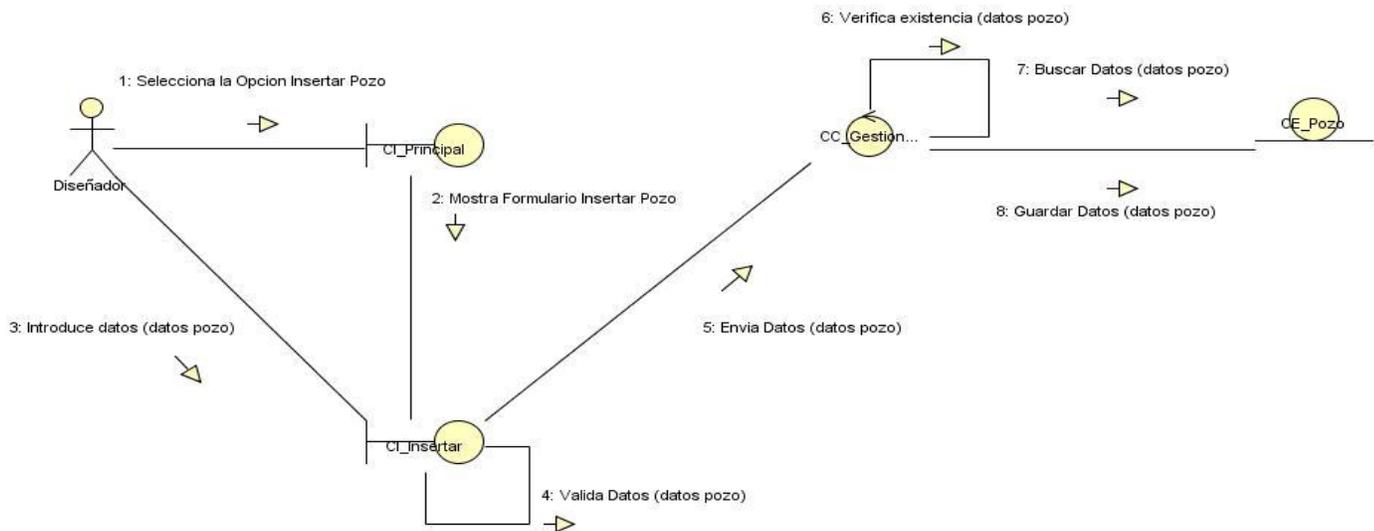


Figura 23: DC CU Gestionar Pozo Sección Insertar

Diagrama de colaboración CU Gestionar Pozo Sección Modificar.

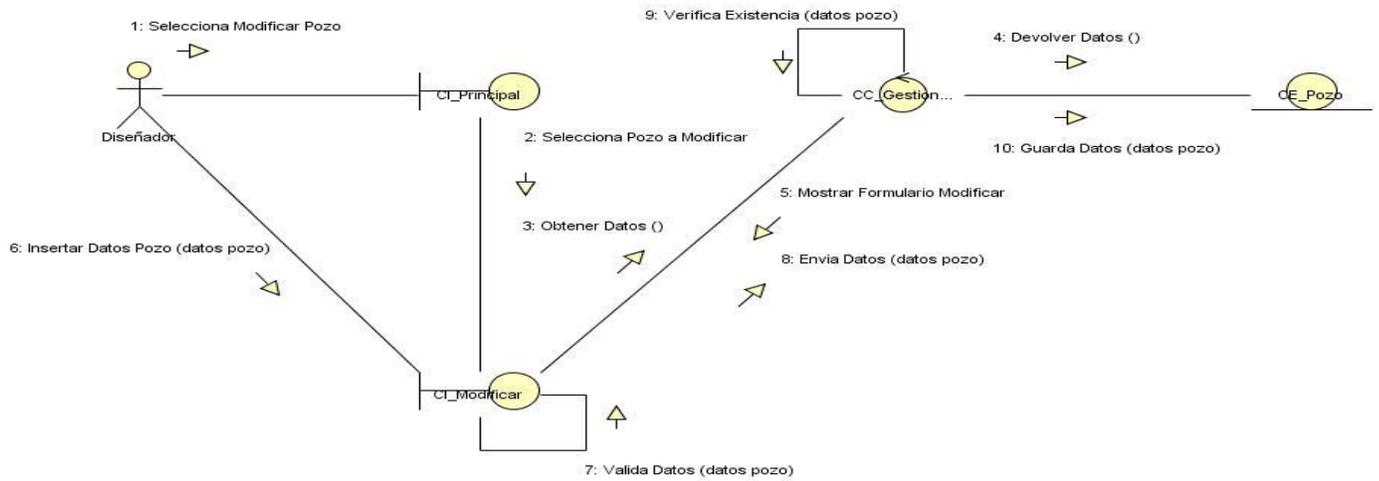


Figura 24: DC CU Gestionar Pozo Sección Modificar

Diagrama de colaboración CU Gestionar Pozo Sección Eliminar.

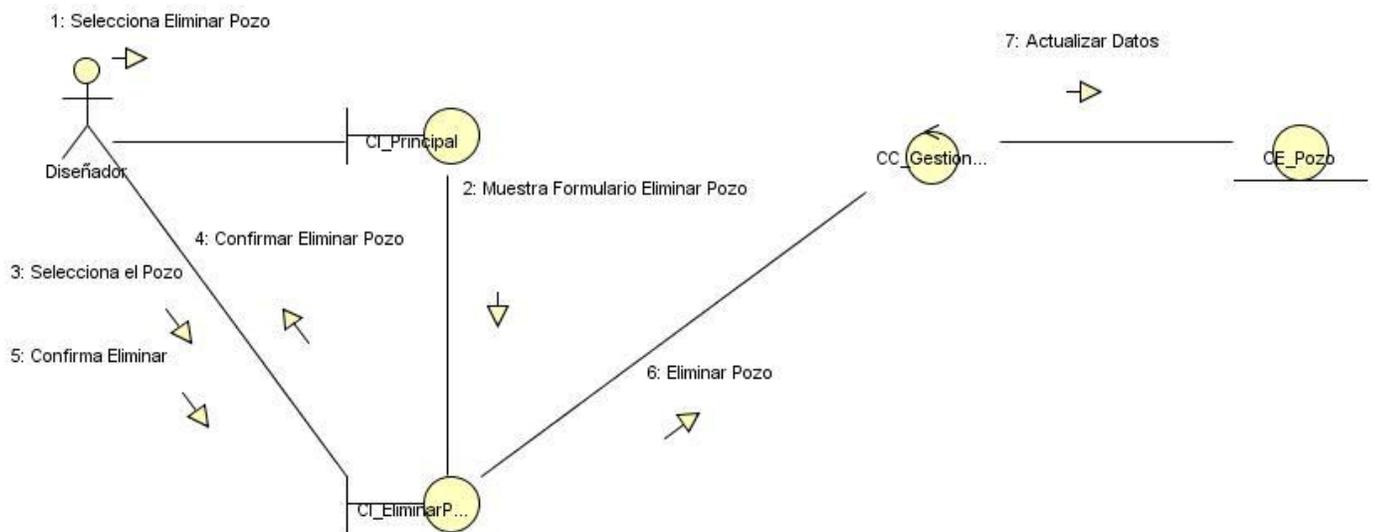


Figura 25: DC CU Gestionar Pozo Sección Eliminar

Diagrama de colaboración CU Gestionar Prueba Sección Insertar.

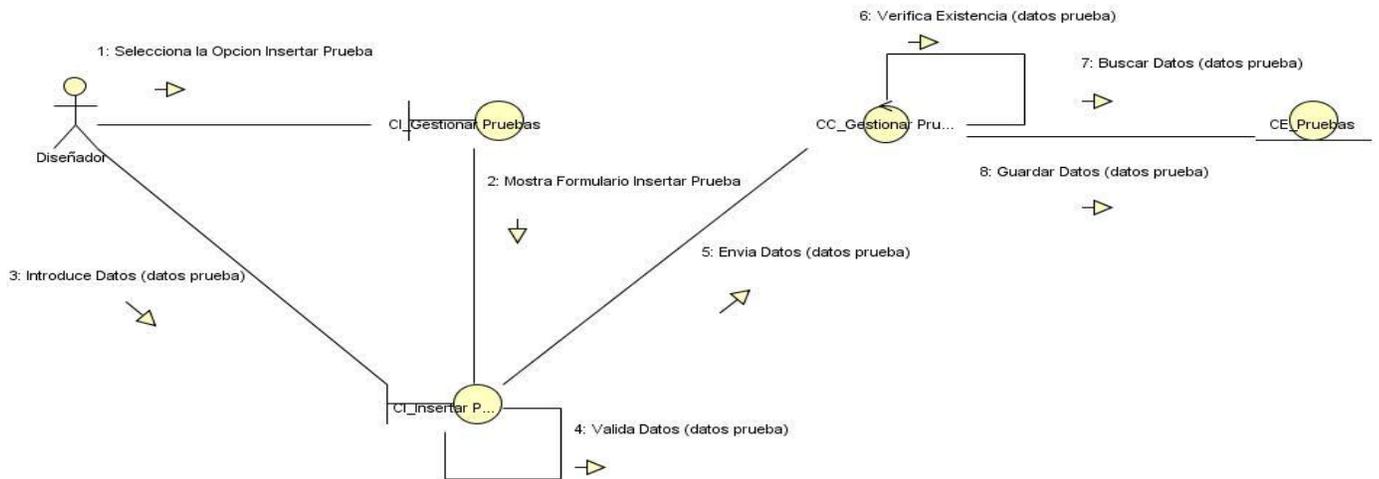


Figura 26: DC CU Gestionar Prueba Sección Insertar

Diagrama de colaboración CU Gestionar Prueba Sección Modificar.

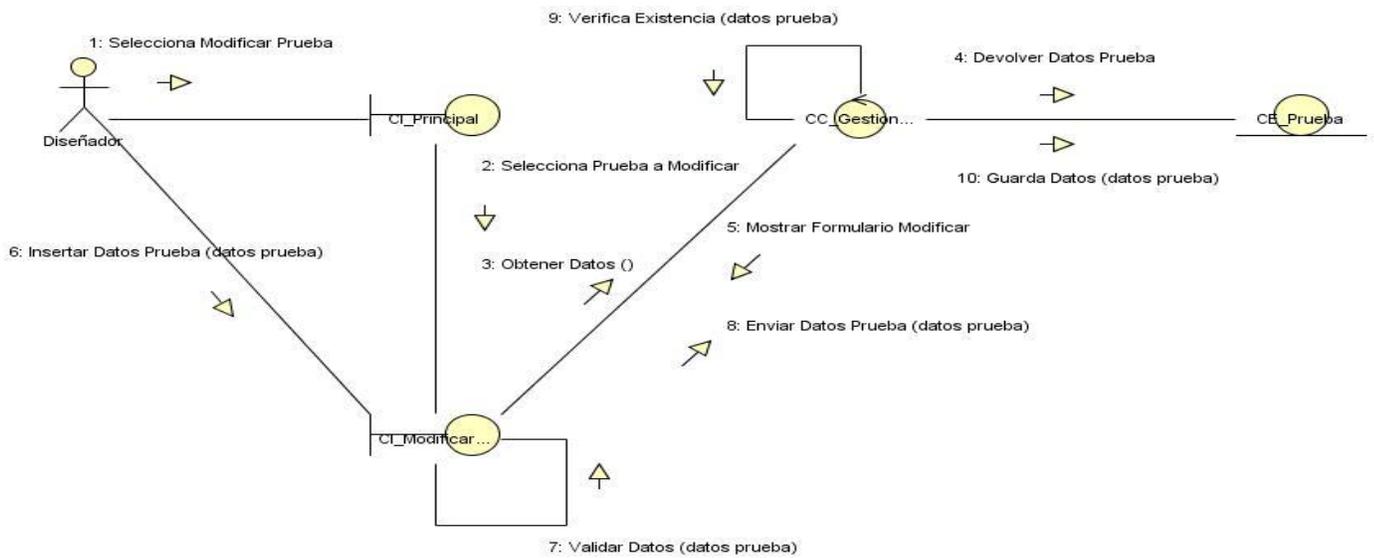


Figura 27: DC CU Gestionar Prueba Sección modificar

Diagrama de colaboración CU Gestionar Prueba Sección Eliminar.

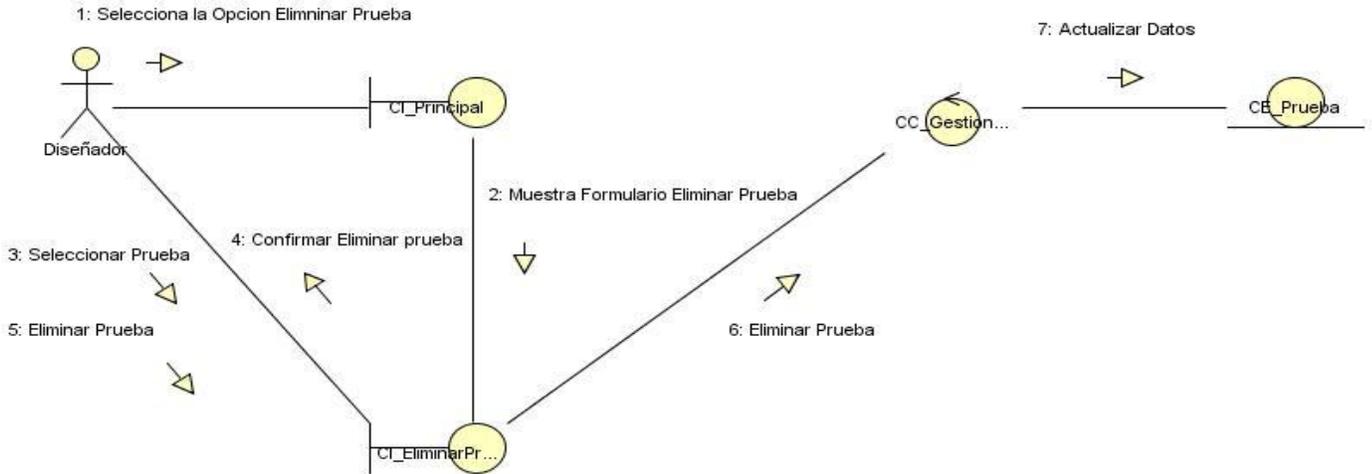


Figura 28: DC CU Gestionar Prueba Sección eliminar

Diagrama de colaboración CU Gestionar Aditivo Sección Insertar.

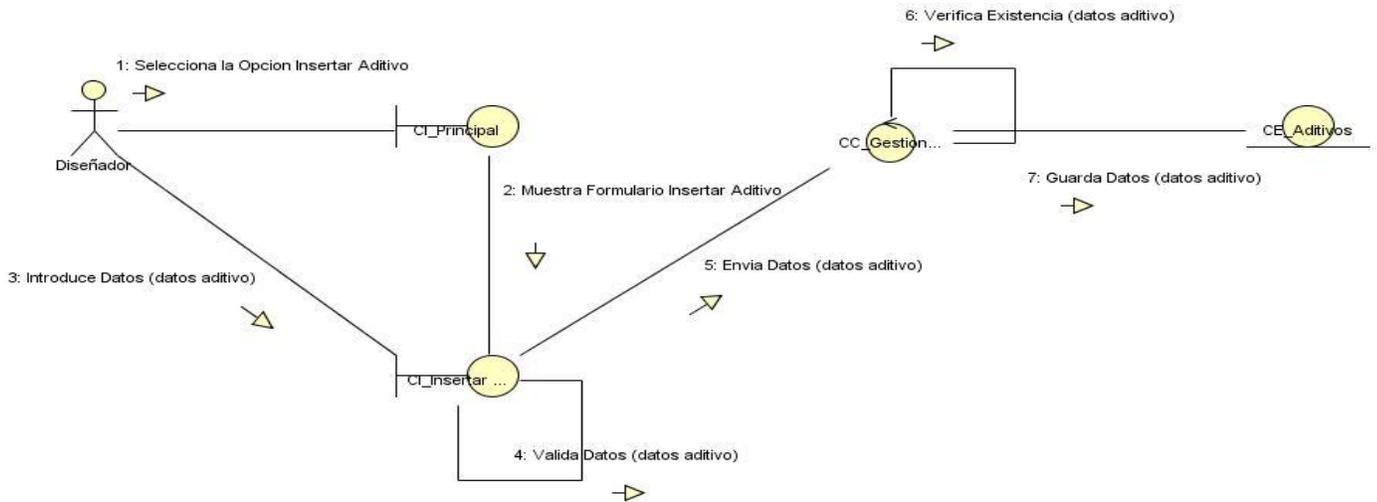


Figura 29: DC CU Gestionar Aditivo Sección Insertar

Diagrama de colaboración CU Gestionar Aditivo Sección Modificar.

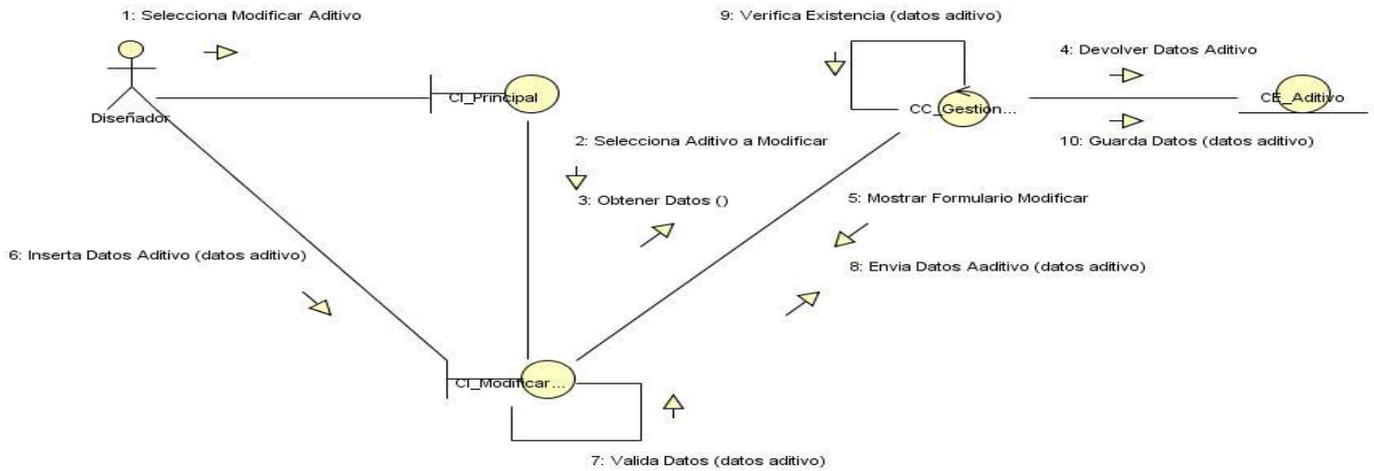


Figura 30: DC CU Gestionar Aditivo Sección Modificar

Diagrama de colaboración CU Gestionar Aditivo Sección Eliminar.

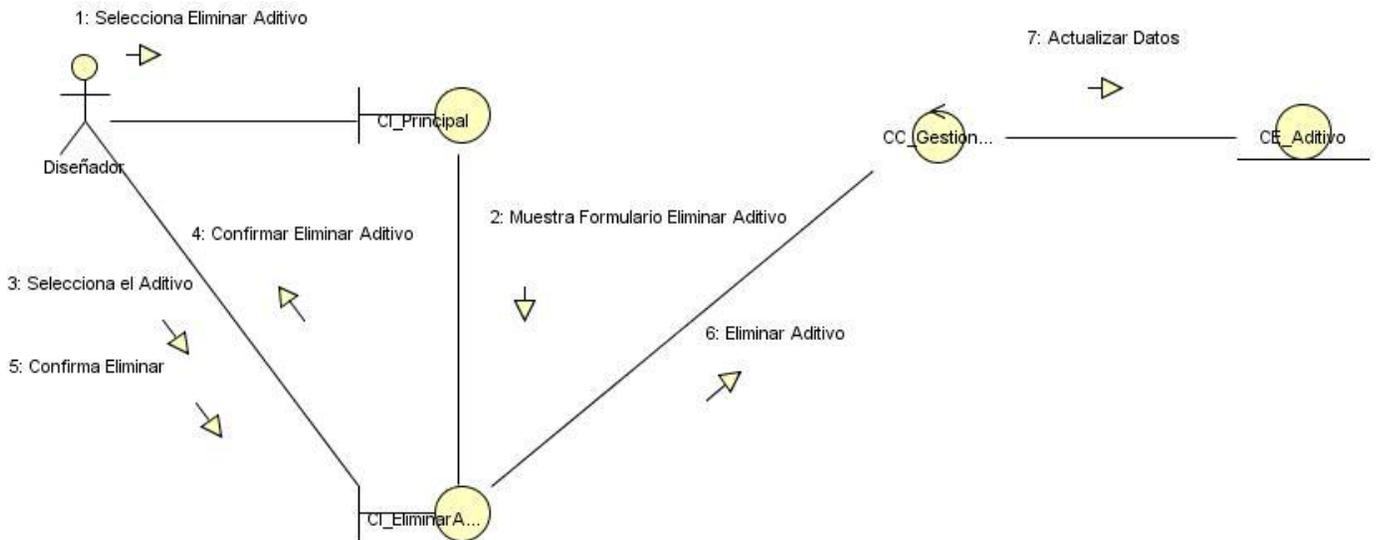


Figura 31: DC CU Gestionar Aditivo Sección Eliminar

Diagrama de colaboración CU Gestionar Usuario Sección Insertar.

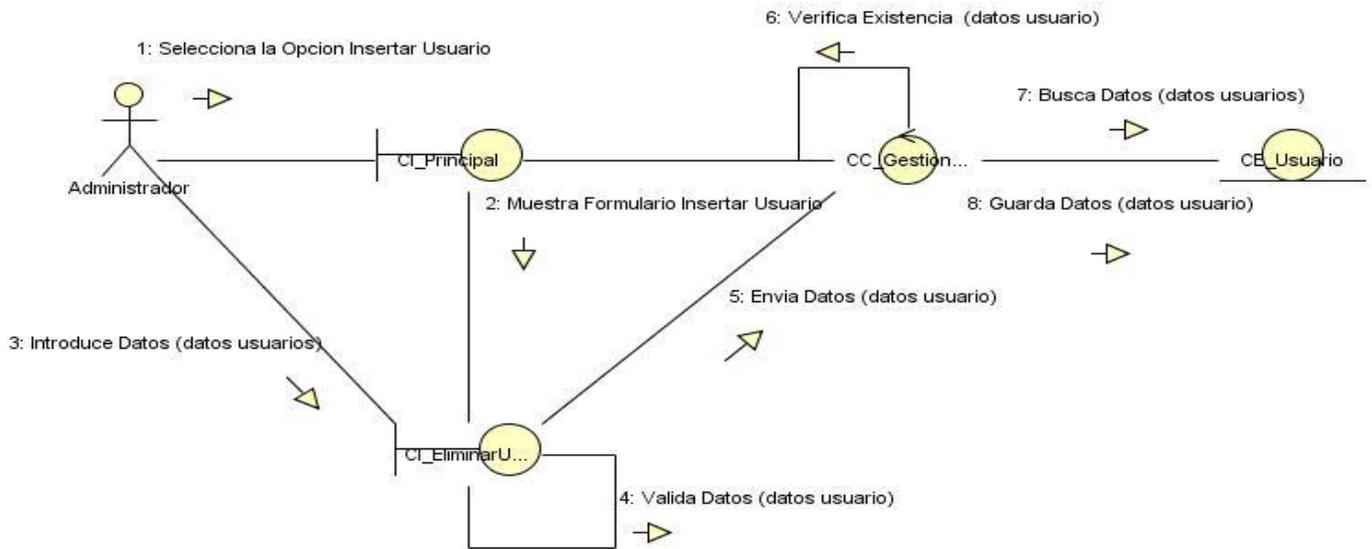


Figura 32: DC CU Gestionar Usuario Sección Insertar

Diagrama de colaboración CU Gestionar Usuario Sección Modificar.

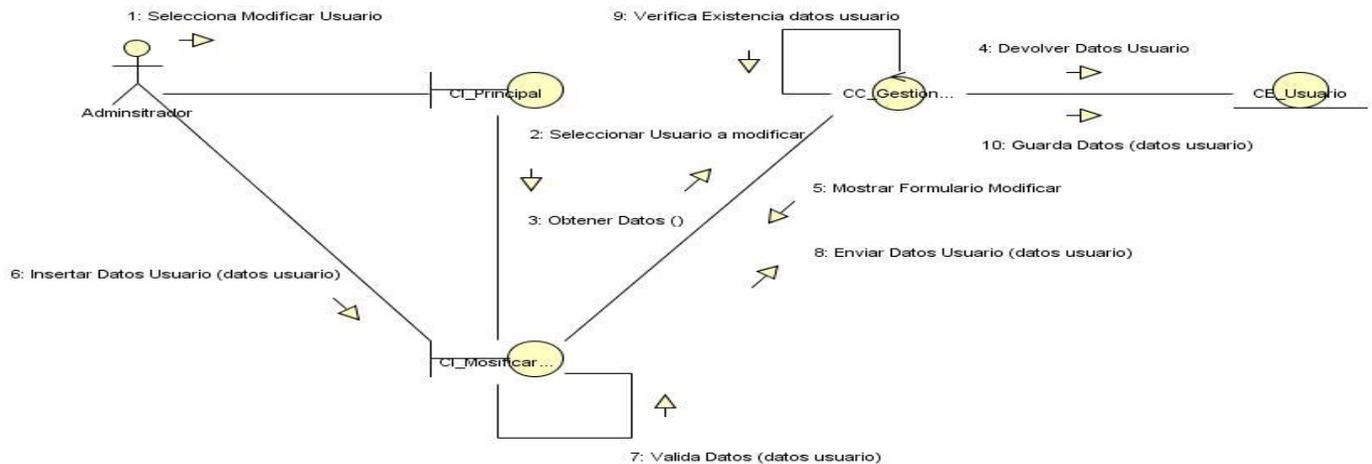


Figura 33: DC CU Gestionar Usuario Sección Modificar

Diagrama de colaboración CU Gestionar Usuario Sección Eliminar.

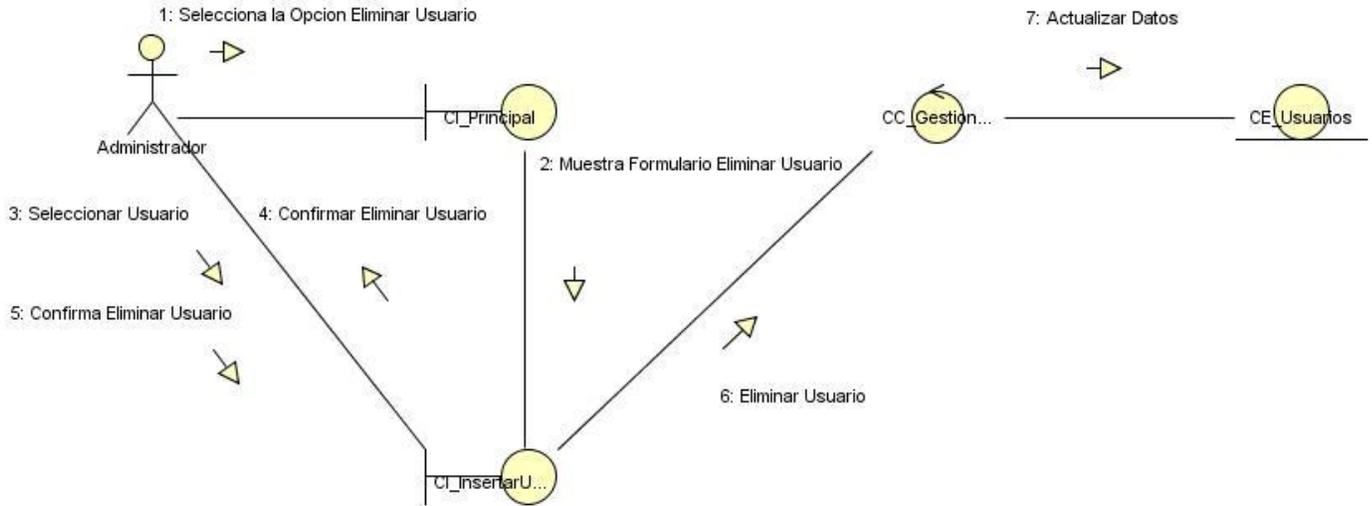


Figura 34: DC CU Gestionar Usuario Sección eliminar

Diagrama de colaboración CU Visualizar Información Sección Buscar Pozo.

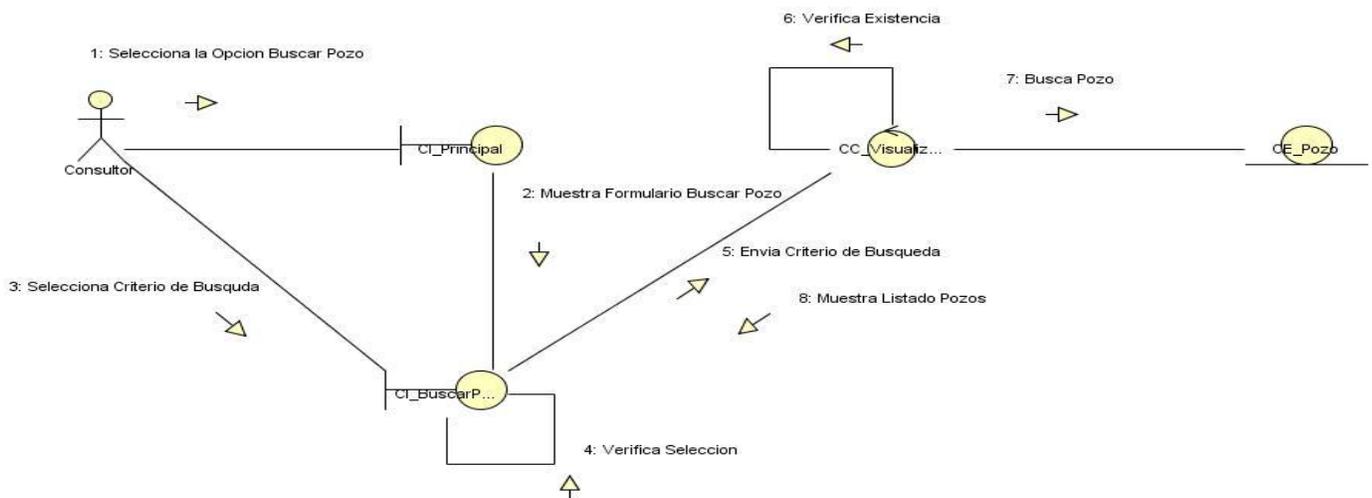


Figura 35: DC CU Visualizar Información Sección Buscar Pozo

Diagrama de colaboración CU Visualizar Información Sección Buscar Prueba.

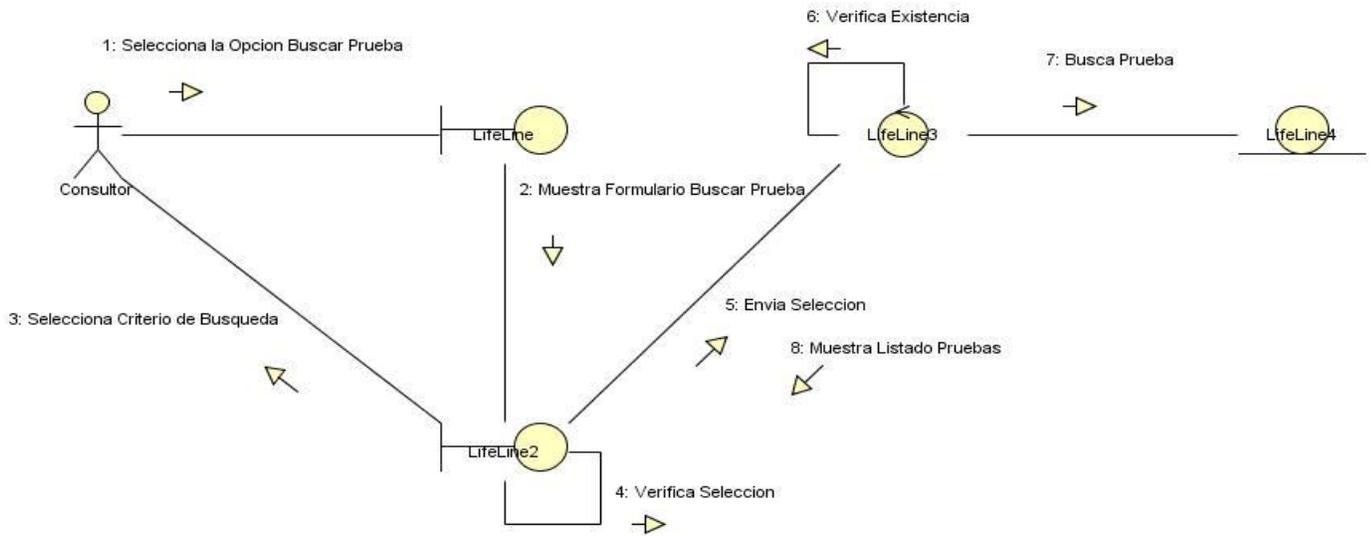


Figura 36: DC CU Visualizar Información Sección Buscar Prueba

Diagrama de colaboración CU Visualizar Información Sección Buscar Aditivo.

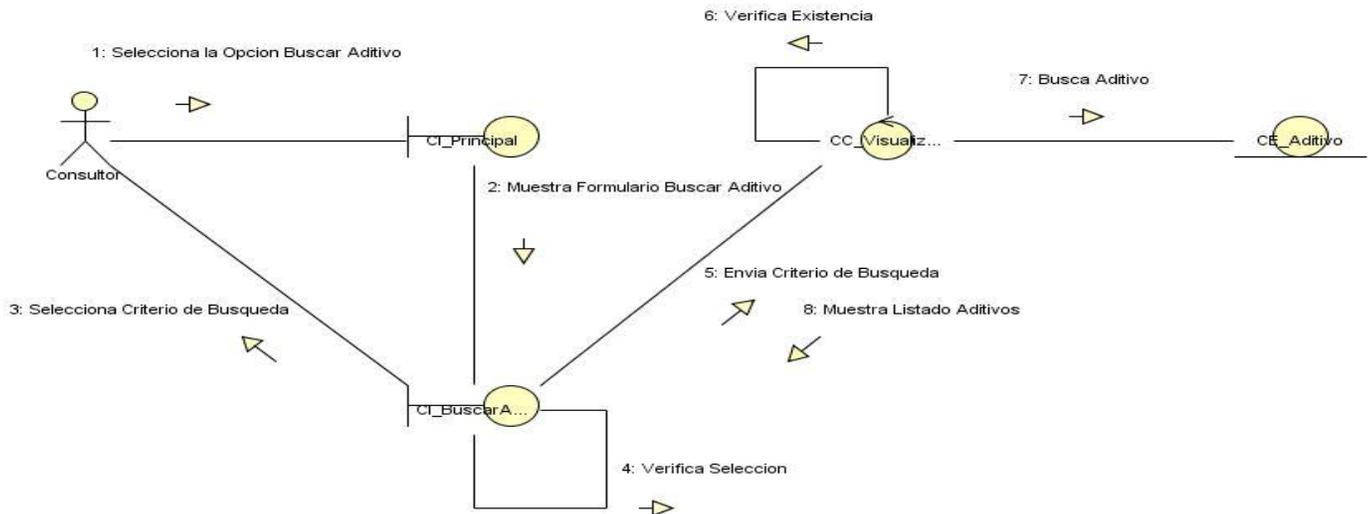


Figura 37: DC CU Visualizar Información Sección Buscar Aditivo

Diagrama de colaboración CU Visualizar Información Sección Buscar Servicios.

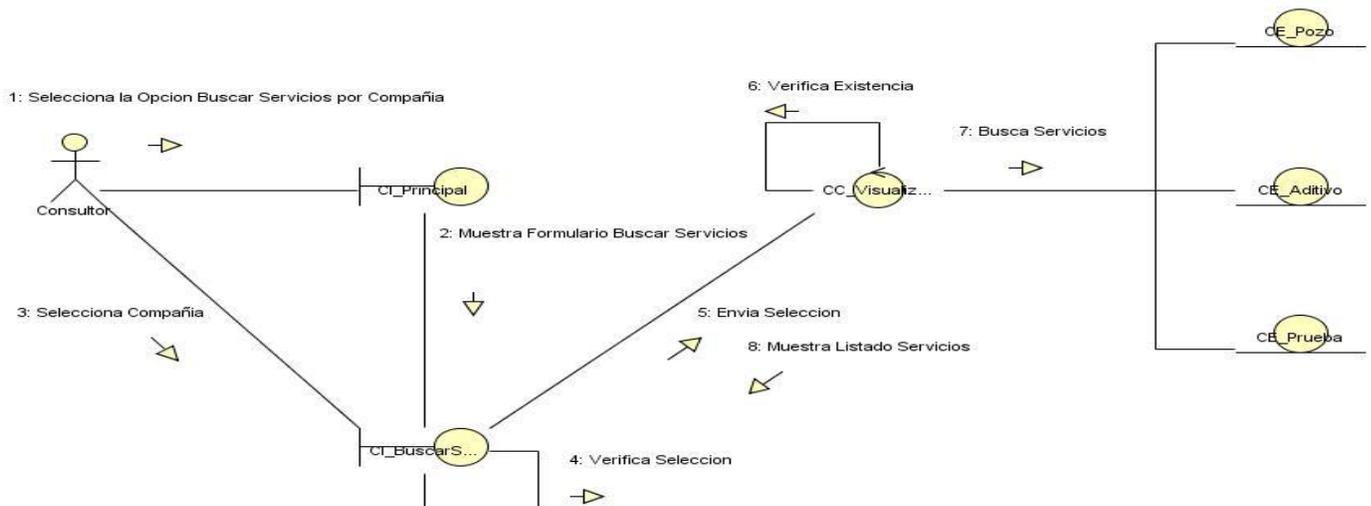


Figura 38: DC CU Visualizar Información Sección Buscar servicios

Diagrama de colaboración CU Generar Reporte.

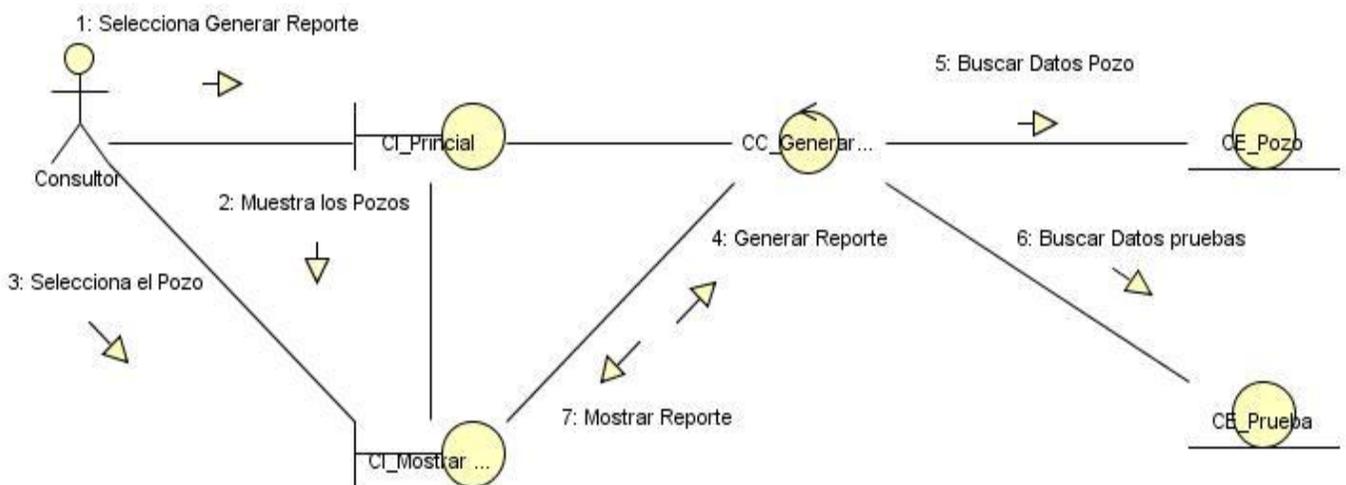


Figura 39: DC CU Generar Reporte

Anexo 3: Diagramas de Clases del diseño.

Diagrama de CD CU Autenticar Usuario.

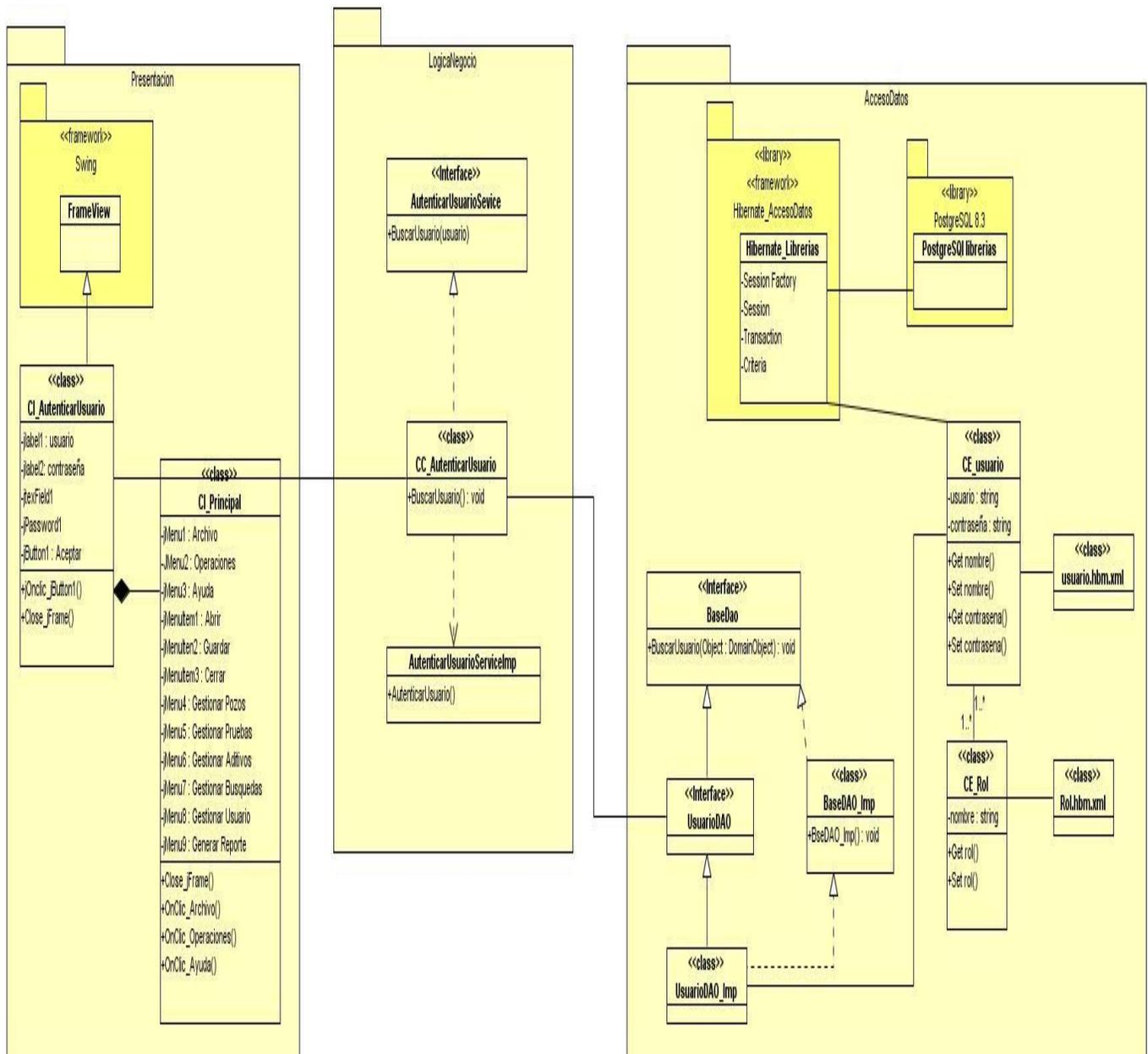


Figura 40: Diagrama de CD Autenticar Usuario.

Diagrama de CD CU Gestionar Pozos.

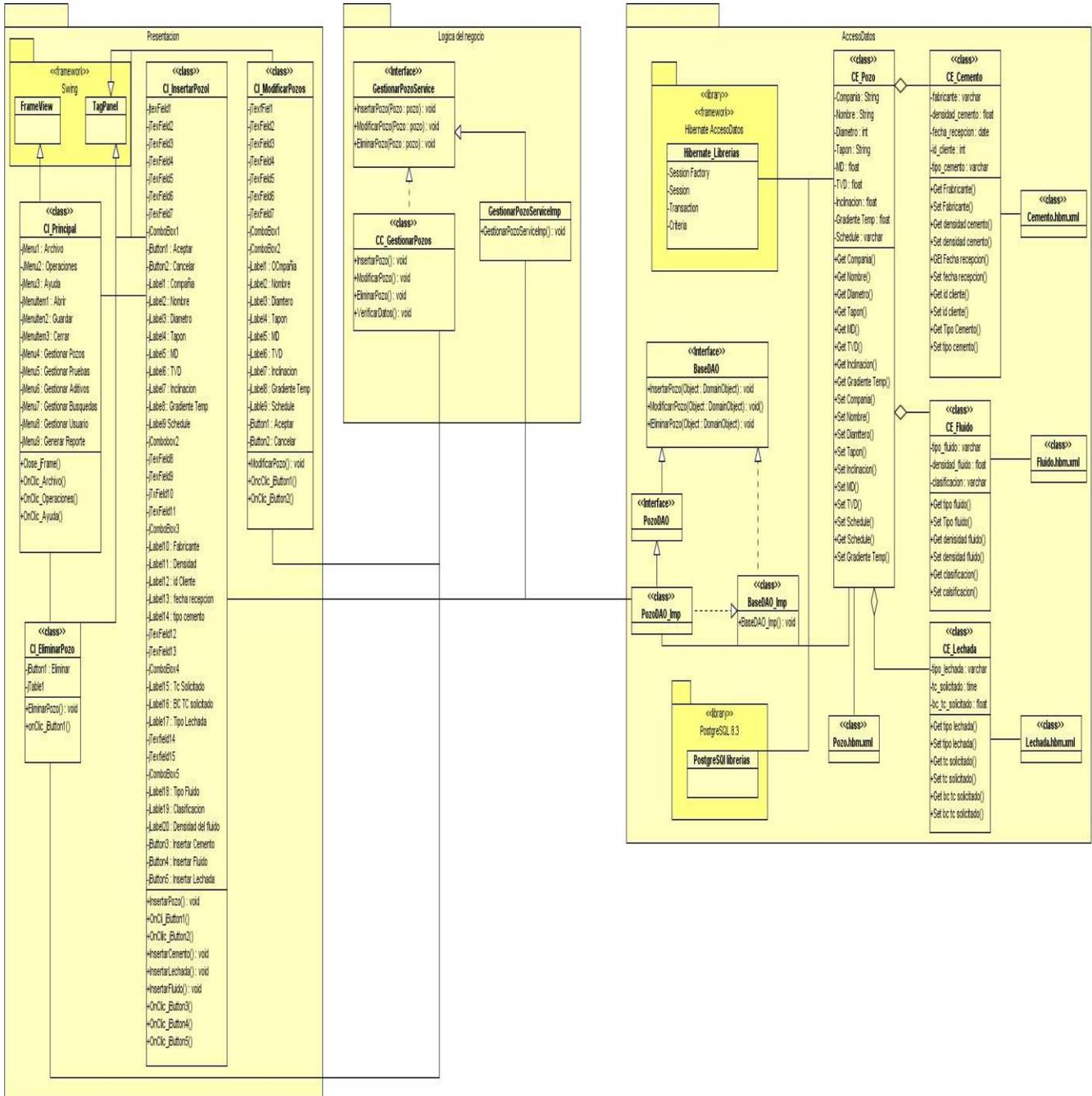


Figura 41: Diagrama de CD CU Gestionar Pozo

Diagrama de CD CU Gestionar Aditivos.

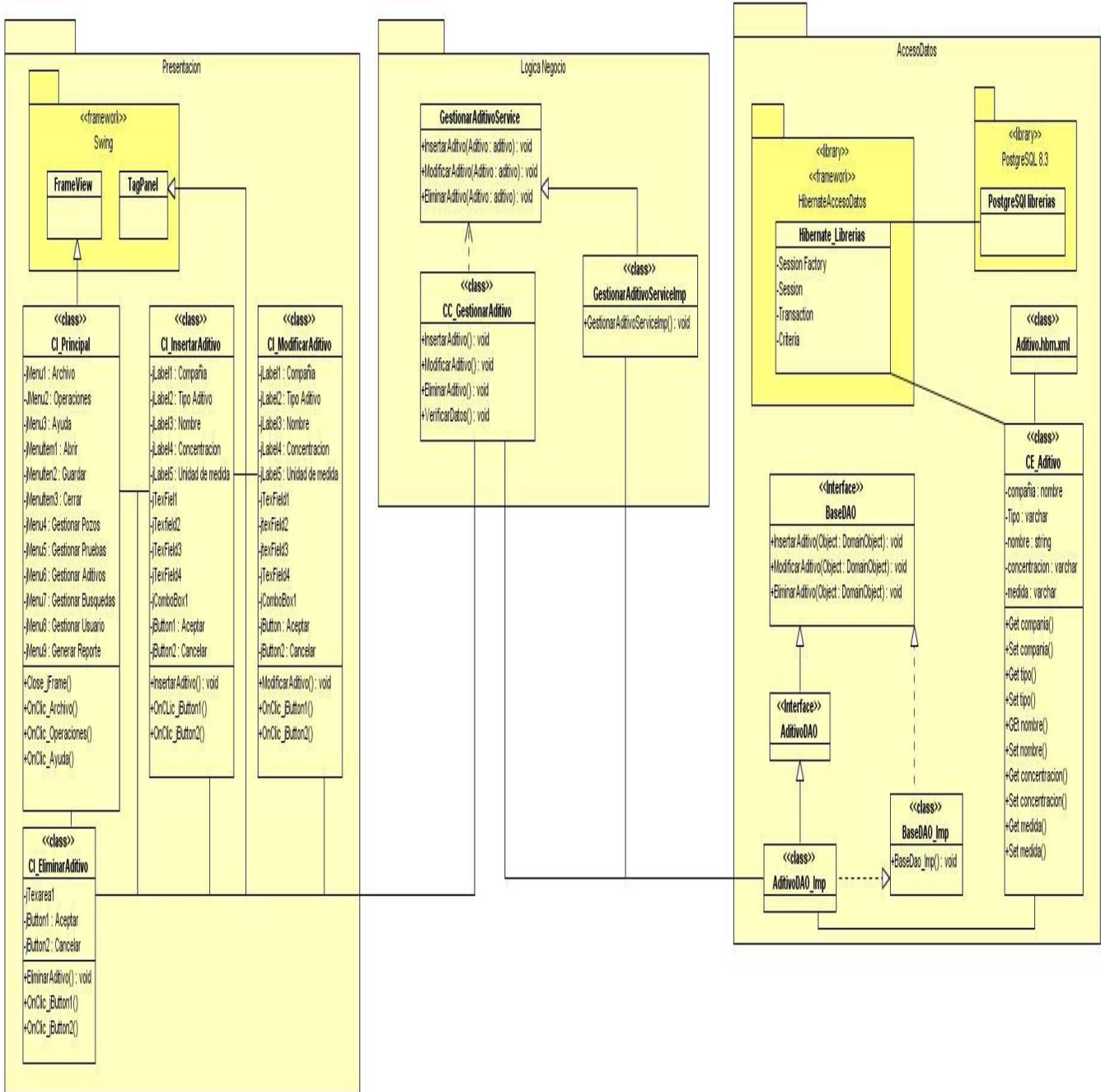


Figura 43: Diagrama de CD CU Gestionar Aditivo

Diagrama de CD CU Visualizar Información.

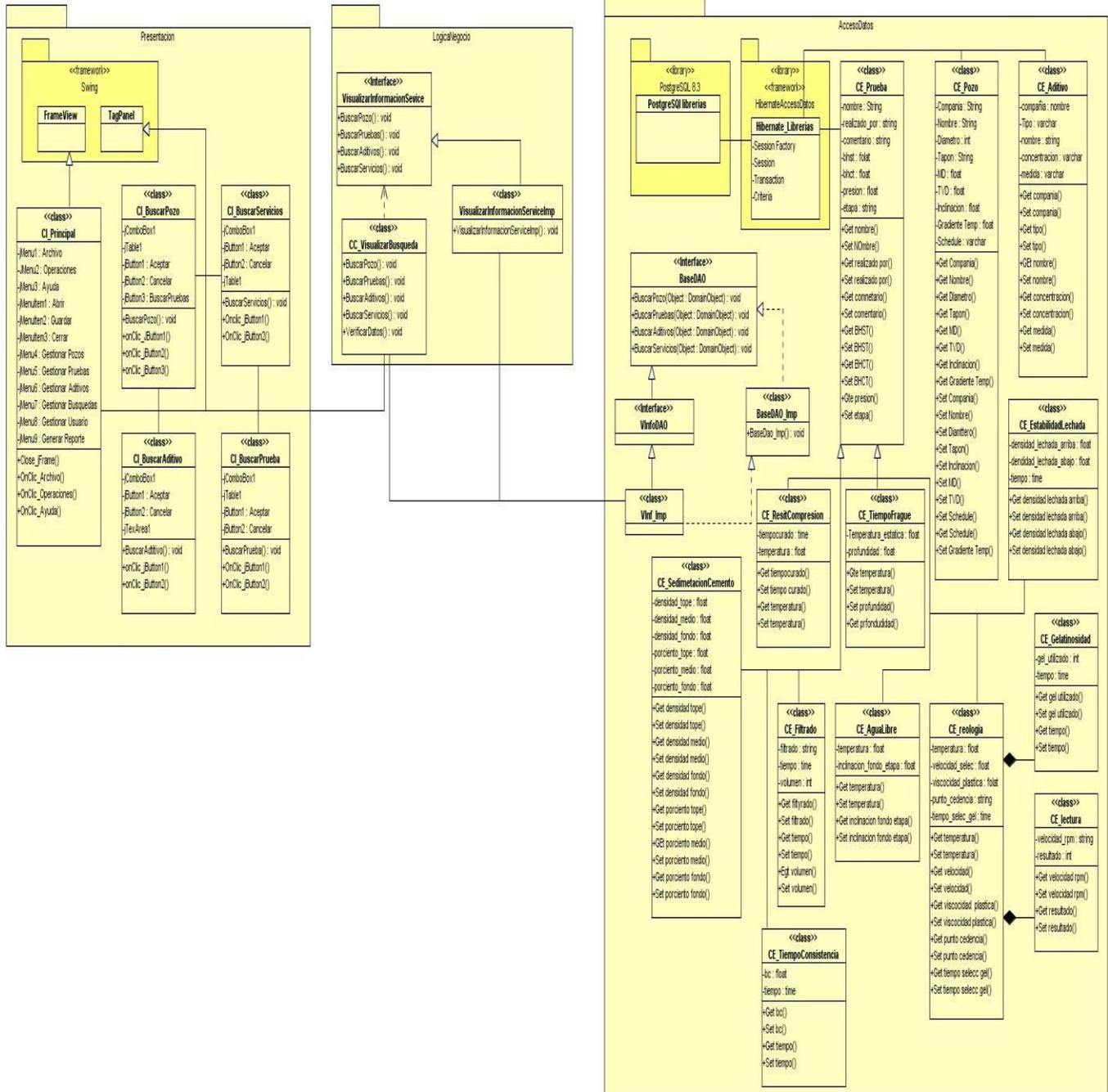


Figura 44: Diagrama de CD CU Visualizar Información

Diagrama de CD CU Generar Reporte.

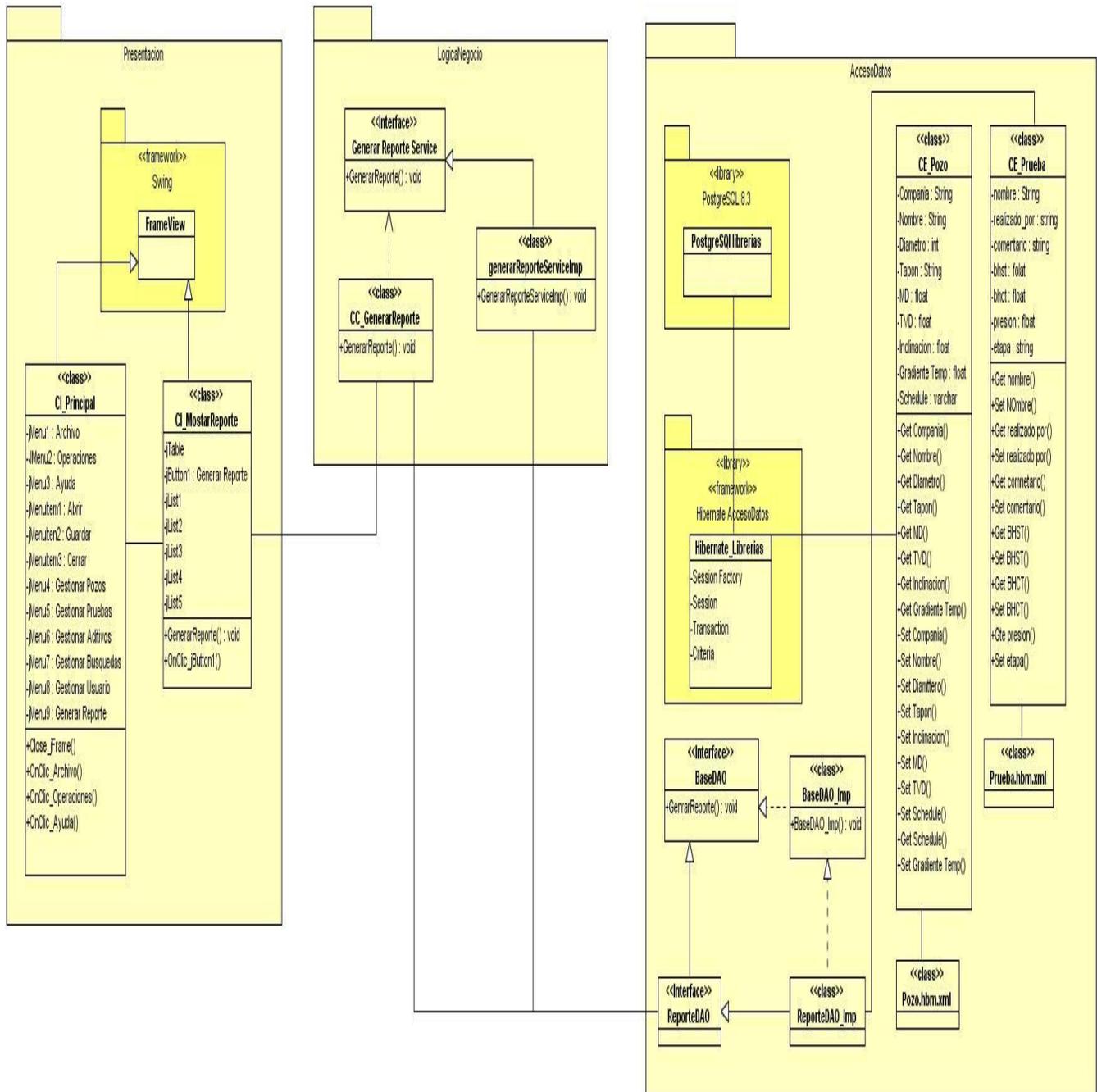


Figura 45: Diagrama de CD CU Generar Reporte

Diagrama de CD CU Gestionar Usuario.

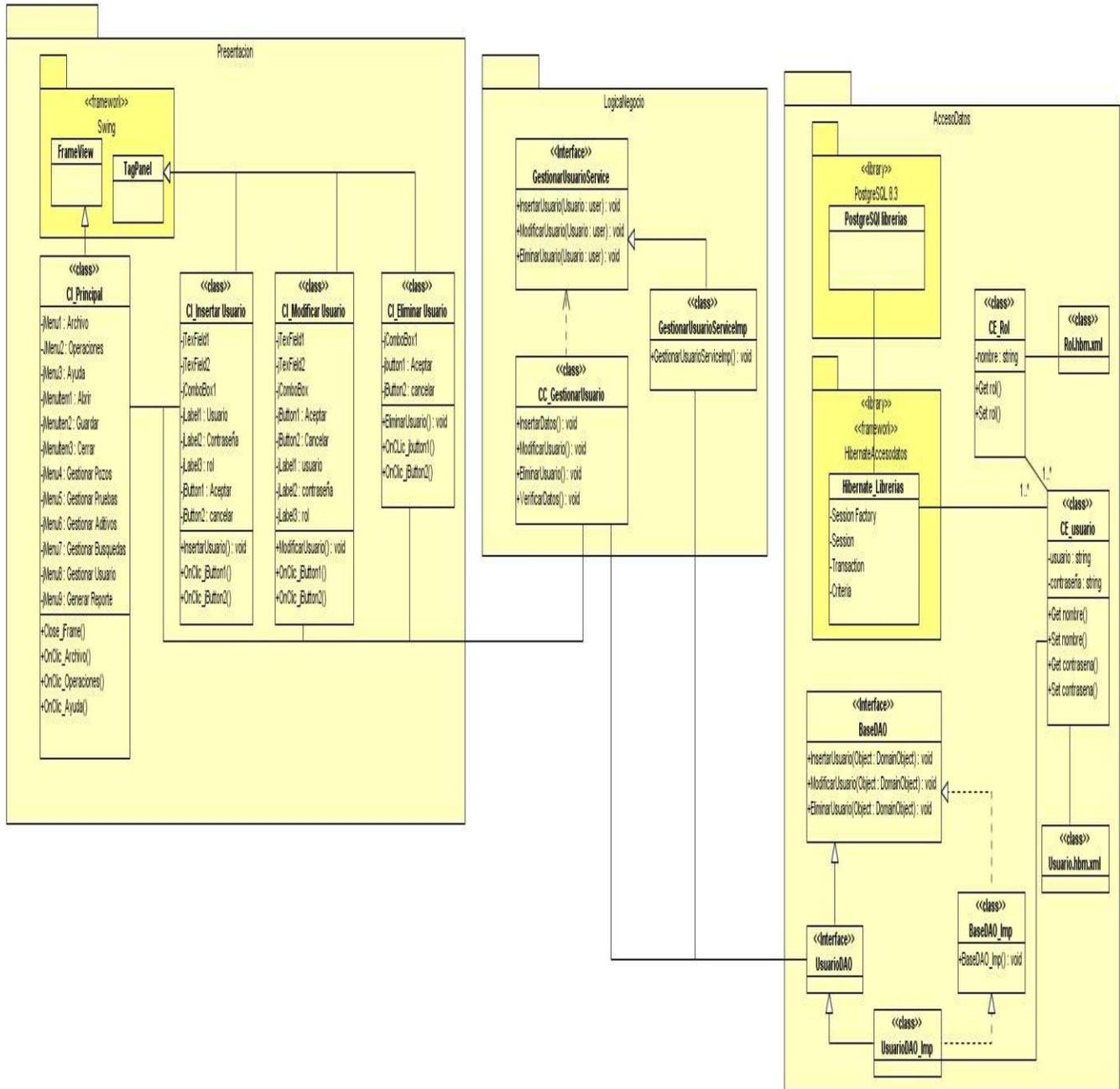


Figura 46: Diagrama de CD CU Gestionar Usuario

Anexo4: Descripción de las Tablas de la Base de Datos

Nombre de la tabla: n_Tipo_Fluido		
Descripción: Almacena el tipo de fluido que habar dentro del pozo. Se usa como un nomenclador		
Atributo	Tipo	Descripción
id_tipo	integer(serial)	Guarda el id del fluido, se toma como llave primaria de la tabla.
nombre	varchar	Almacena el nombre del fluido.

Tabla 9: Descripción de la Tabla Tipo_Fluido

Nombre de la tabla: Usuario		
Descripción: Almacena los usuarios registrados en la BD.		
Atributo	Tipo	Descripción
id_usuario	integer(serial)	Almacena el identificador del usuario, se toma como llave primaria de la tabla.
Usuario	String	Almacena el nombre del usuario registrado.
password	String	Almacena la contraseña del dicho usuario.

Nombre	String	Almacena el nombre real de la persona que se autentica con ese usuario
Apellidos	String	Almacena la los apellidos de la persona.

Tabla 10: Descripción de la Tabla Usuario

Nombre de la tabla: Rol		
Descripción: Almacena el rol que va a tener un usuario en específico.		
Atributo	Tipo	Descripción
Id_rol	Integer	Almacena el identificar del rol, s e toma como llave primaria
Rol	String	Almacena el rol del usuario.

Tabla 11: Descripción de la Tabla Rol

Nombre de la tabla: lechada		
Descripción: Almacena los datos del tipo de lechada que solicita un cliente para que se le realice el diseño.		
Atributo	Tipo	Descripción
Id_lechada	Integer(serial)	Almacena el identificador de la lechada, se toma como llave primaria de la tabla.

Tipo_lechada	String	Almacena el tipo de la lechada solicitada por el cliente (amarre, relleno, otro).
Tc_solicitado	Time	Almacena el tiempo de consistencia solicitado por el cliente.
Bc_tc_solicitado	Float	Almacena las unidades barden de consistencia solicitadas al tiempo solicitado.

Tabla 12: Descripción de la Tabla Lechada

Nombre de la tabla: Cemento		
Descripción: Almacena los datos del cemento que se utiliza para la realización del diseño de dicha lechada.		
Atributo	Tipo	Descripción
Id_cemento	Integer(serial)	Almacena el identificador y se toma como llave primaria de la tabla.
Tipo_cemento	Varchar	Almacena el nombre del cemento que se utiliza.
Fabricante	String	Almacena el nombre del fabricante del cemento
Densidad_cemento	String	Almacena la densidad del cemento.

Fecha_recepción	Date	Almacena la fecha en que el cliente compro el cemento
Id_cliente	Integer	Almacena el identificador que le da el laboratorio al cliente al cual se le está realizando el diseño de la lechada de cementación.

Tabla 13: Descripción de la Tabla Cemento

Nombre de la tabla: Fluido		
Descripción: Almacena los datos del tipo de fluido que se encuentra dentro del pozo.		
Atributo	Tipo	Descripción
Id_fluido	Integer(serial)	Almacena el identificador de la tabla, el mismo se toma como llave primaria.
Tipo_fluido	String	Almacena el nombre del fluido que se encuentra en el pozo.
Densidad_fluido	Float	Almacena la densidad del fluido dentro del pozo.
Clasificación	string	Almacena la clasificación del fluido.

Tabla 14: Descripción de la Tabla Fluido

Nombre de la tabla: Pozo		
Descripción: Almacena los datos del pozo que fueron enviados por el cliente para que se le realizara el diseño de la lechada de cementación.		
Atributo	Tipo	Descripción
Nombre_pozo	Varchar	Almacena el nombre del pozo, se toma como llave primaria ya que para cada pozo el nombre es único.
Compañía	String	Almacena el nombre de la compañía que solicito el trabajo.
Diámetro	Int	Almacena el diámetro del pozo.
Tipo_tapón	String	Almacena el nombre del tipo de tapón con el que se esta trabajando.
Md	Float	Almacena la profundidad del pozo
Tvd	Float	Almacena la profundidad del pozo por la vertical.
Inclinación	Float	Almacena el ángulo de inclinación del pozo.
Gradiente_temperatura	Float	Almacena el gradiente de temperatura que tiene el pozo dentro.

Bhst	Float	Almacena la temperatura estática en el fondo de la etapa.
Bhct	Float	Almacena la temperatura de circulación en el fondo de la etapa.
Presión	Float	Almacena la presión en el fondo de la etapa.
Tipo_revestimiento	Varchar	Almacena el revestimiento que va a tener el pozo, puede ser liner, camisa y otros.
Schedule	Varchar	Almacena el Schedule utilizado a la hora de realizarle el diseño de una lechada a un pozo.

Tabla 15: Descripción de la Tabla Pozo

Nombre de la tabla: Aditivo		
Descripción: Almacena los datos de los aditivos que tiene una compañía determinada, con los cuales se realizan las pruebas para el diseño de la lechada.		
Atributo	Tipo	Descripción
Id_aditivo	Integer(serial)	Almacena el identificador del aditivo, se toma como llave primaria.

Tipo_aditivo	String	Almacena la clasificación del aditivo.
Nombre	String	Almacena el nombre del aditivo.
Concentración	Float	Almacena la concentración del aditivo.
Compañía	String	Almacena el nombre de la compañía que usa dicho aditivo.
Unidad_medida_aditivo	String	Almacena la unidad con que se trabaja el uso del aditivo.

Tabla 16: Descripción de la Tabla Aditivo

Nombre de la tabla: Prueba		
Descripción: Almacena los datos de las pruebas que se realizan, trabaja como padre de otras tablas de pruebas que tendrán datos específicos para cada prueba.		
Atributo	Tipo	Descripción
Id_prueba	Integer(serial)	Almacena el identificador de la prueba y se toma como llave primaria de la misma.
Nombre	String	Almacena el nombre de la prueba que se realiza.
Realizado_por	String	Almacena el nombre de la persona que

		realiza la prueba.
Comentario	String	Almacena algún comentario o sugerencia que se quiera hacer al cliente debido a la prueba realizada.
Etapa	String	Almacena el nombre de la etapa en que se realiza la prueba (única, primera, segunda, otra)
Número_prueba	Int	Almacena el numero de la prueba
Fecha	Date	Almacena la fecha en que se realiza la prueba.

Tabla 17: Descripción de la Tabla Prueba

Nombre de la tabla: TiempoConsistencia		
Descripción: Almacena los datos específicos de las pruebas de tiempo de consistencia que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_consistencia	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de esa tabla.

Bc	Float	Almacena las unidades barden de consistencia calculadas
Tiempo	Time	Almacena el tiempo que dura la lechada para alcanzar la consistencia deseada.

Tabla 18: Descripción de la Tabla TiempoConsistencia

Nombre de la tabla: Filtrado		
Descripción: Almacena los datos específicos de las pruebas de filtrado que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_filtrado	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
filtrado	String	Almacena el filtrado alcanzado para dicha prueba.
Tiempo	Time	Almacena el tiempo que tardo para obtener el filtrado completo de la prueba.
Volumen	Int	Almacena el volumen de la prueba.

Tabla 19: Descripción de la Tabla Filtrado

Nombre de la tabla: AguaLibre

Descripción: Almacena los datos específicos de las pruebas de agua libre que se realiza a la hora de diseñar una lechada.

Atributo	Tipo	Descripción
Id_agualibre	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Temperatura	Float	Almacena la temperatura que alcanza el agua para dicha prueba.
Inclinación_fondo_etapa	Float	Almacena la inclinación en el fondo de la etapa del pozo para esa prueba.

Tabla 20: Descripción de la Tabla AguaLibre

Nombre de la tabla: EstabilidadLechada		
Descripción: Almacena los datos específicos de las pruebas de estabilidad de la lechada que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_estabilidad	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Densidad_lechada_arriba	Float	Almacena la densidad de la lechada cuando está en la punta del pozo

Densidad_lechada_abajo	Float	Almacena la densidad de la lechada abajo.
------------------------	-------	---

Tabla 21: Descripción de la Tabla EstabilidadLechada

Nombre de la tabla: SedimentaciónCemento		
Descripción: Almacena los datos específicos de las pruebas de sedimentación de las piedras de cemento que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_sedimentación	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Densidad_tope	float	Almacena la densidad de la piedra-lechada en el tope del pozo.
Densidad_medio	float	Almacena la densidad de la piedra-lechada en el medio del pozo.
Densidad_fondo	float	Almacena la densidad de la piedra-lechada en el fondo del pozo.
Porcentaje_tope	float	Almacena el % de diferencia de las densidades piedra-lechada en el tope del pozo

Porciento_medio	float	Almacena el % de diferencia de las densidades piedra-lechada en el medio del pozo
Porciento_fondo	float	Almacena el % de diferencia de las densidades piedra-lechada en el medio del pozo

Tabla 22: Descripción de la Tabla SedimentaciónCemento

Nombre de la tabla: TiempoFrague		
Descripción: Almacena los datos específicos de las pruebas del tiempo de fragüe que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_frague	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Temperatura_estática	Float	Almacena la temperatura estática que existe en el momento de la prueba.
Profundidad	Float	Almacena la profundidad que alcanza antes del fragüe.

Tabla 23: Descripción de la Tabla TiempoFrague

Nombre de la tabla: ResistCompresión

Descripción: Almacena los datos específicos de las pruebas de resistencia a la compresión que se realiza a la hora de diseñar una lechada.

Atributo	Tipo	Descripción
Id_compresión	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Tiempo_curado	Time	Almacena el tiempo en que tarda la piedra de cemento en destruirse.
Temperatura	Float	Almacena la temperatura a la que fue sometida la piedra de cemento para la prueba.

Tabla 24: Descripción de la Tabla ResistCompresión

Nombre de la tabla: Reología		
Descripción: Almacena los datos específicos de las pruebas de reología que se realiza a la hora de diseñar una lechada.		
Atributo	Tipo	Descripción
Id_reología	Integer(serial)	Almacena el identificador de la prueba, se toma como llave primaria de la tabla.
Temperatura	float	Almacena la temperatura a la que se realiza la prueba.

Velocidad_seleccionada	Float	Almacena la velocidad que se selecciona a la hora de realizar la prueba.
Viscosidad_plástica	float	Almacena la viscosidad del la lechada cuando se realiza la prueba.
Punto_cedencia	String	Almacena el punto donde cede la lechada en el momento en que se está realizando la prueba.
Tiempo_seleccionado_gel	time	Almacena el tiempo seleccionado para el gel usado.

Tabla 25: Descripción de la Tabla Reología

Nombre de la tabla: lectura		
Descripción: Almacena los datos de las lecturas realizadas en la prueba de reologia.		
Atributo	Tipo	Descripción
Id_lectura	Integer(serial)	Almacena el identificador de la lectura, se toma como llave primaria de la tabla.
Velocidad_rpm	Float	Almacena la velocidad a la que se realiza la lectura.
Resultado	Float	Almacena el resultado de dicha lectura.

Tabla 26: Descripción de la Tabla Lectura

Nombre de la tabla: Gelatinosidad		
Descripción: Almacena los datos del gel con que se trabaja a la hora de realizar la prueba de reología.		
Atributo	Tipo	Descripción
Id_gelatinosidad	Integer(serial)	Almacena el identificador de la gelatinosidad, se toma como llave primaria de la tabla.
Gel_utilizado	Float	Almacena el gel que se utiliza para la prueba de reología.
Tiempo	Time	Almacena el tiempo con que se trabaja el gel seleccionado.

Tabla 27: Descripción de la Tabla Gelatinosidad

Nombre de la tabla: n_Tipo Cemento		
Descripción: Almacena los nombres de los diferentes cementos existentes que se usan para el diseño de las lechadas.		
Atributo	Tipo	Descripción
Id_tipo	Integer(serial)	Almacena el id del tipo de cemento, se usa como llave primaria.

Nombre	Varchar	Almacena el nombre del cemento
--------	---------	--------------------------------

Tabla 28: Descripción de la Tabla n_TipoCemento

Nombre de la tabla: n_compañía		
Descripción: almacena el nombre de las compañías a la cuales se les realiza el diseño de la lechada de cementación a sus pozos, se usa como un nomenclador.		
Atributo	Tipo	Descripción
Id_comp	Integer(serial)	Almacena el identificador de la tabla y se usa como llave primaria de la misma.
Nombre	Varchar	Almacena el nombre de la compañía.

Tabla 29: Descripción de la Tabla n_Compañía

Nombre de la tabla: n_tapon		
Descripción: almacena los nombres de los distintos tapones que usan los pozos, se usa como un nomenclador.		
Atributo	Tipo	Descripción
Id_tapón	Integer(serial)	Almacena el identificador de la tabla y se usa como llave primaria de la misma.
Nombre	Varchar	Almacena el nombre del tapón.

Tabla 30: Descripción de la Tabla n_Tapón

Nombre de la tabla: n_schedule		
Descripción: almacena el schedule que se utiliza en el diseño, se usa como un nomenclador.		
Atributo	Tipo	Descripción
Id_schedule	Integer(serial)	Almacena el identificador de la tabla y se usa como llave primaria de la misma.
Nombre	Varchar	Almacena el nombre del Schedule.

Tabla 31: Descripción de la Tabla n_Schedule

Nombre de la tabla: pozo_prueba		
Descripción: almacena los identificadores de las tablas pozo y prueba, representando la relación de m:n que existe entre ellas.		
Atributo	Tipo	Descripción
Id_pozo	Integer(serial)	Almacena el identificador del pozo y se usa como llave primaria de la misma.
Id_prueba	Integer(serial)	Almacena el identificador de la prueba y se usa como llave primaria de la misma.

Tabla 32: Descripción de la Tabla Pozo_Prueba

Nombre de la tabla: aditivo_prueba		
Descripción: almacena los identificadores de las tablas aditivo y prueba, representando la relación de m:n que existe entre ellas.		
Atributo	Tipo	Descripción
Id_aditivo	Integer(serial)	Almacena el identificador de la tabla aditivo y se usa como llave primaria de la misma.
Id_prueba	Integer(serial)	Almacena el identificador de la prueba y se usa como llave primaria de la misma.

Tabla 33: Descripción de la Tabla Aditivo_Prueba

Nombre de la tabla: n tipo_lechada		
Descripción: almacena los datos de las lechadas solicitadas por los clientes.		
Atributo	Tipo	Descripción
Id_tipo	Integer(serial)	Almacena el identificador de la tabla y se usa como llave primaria de la misma.
Nombre	Varchar	Almacena el nombre de la lechada.

Tabla 34: Descripción de la Tabla n_Tipo_Lechada

Nombre de la tabla: n_TipoRevestimiento		
Descripción: Almacena el tipo del revestimiento que tendrá el pozo, puede ser liner, camisa u otro. Se usa como un nomenclador		
Atributo	Tipo	Descripción
id_tipo	integer(serial)	Guarda el id del fluido, se toma como llave primaria de la tabla.
nombre	varchar	Almacena el nombre del fluido.

Tabla 35: Descripción de la tabla Tipo_Revestimiento

Nombre de la tabla: n_TipoAditivo		
Descripción: Almacena el tipo de aditivo que habrá dentro del pozo. Se usa como un nomenclador		
Atributo	Tipo	Descripción
id_tipo	integer(serial)	Guarda el id del fluido, se toma como llave primaria de la tabla.
nombre	Varchar	Almacena el nombre del fluido.

Tabla 36: Descripción de la Tabla Tipo_Aditivo

GLOSARIO DE TÉRMINOS

ANACEM (Análisis de Cementaciones): Es el Sistema de Gestión de Lechadas de Cementación para Pozos Petroleros.

ARRAY (Arreglo): Estructura de almacenamiento de datos utilizada en todos los lenguajes de programación.

API: Abreviaturas de American Petroleum Institute, localizada en Washington, D.C. Este es el centro de la asociación para las industrias petroleras.

BC: Unidades Bearden de consistencia.

BD: Base de Datos.

BHCT: Bottom Hole Circulation Temperature. Es la temperatura de circulación en la boca del pozo.

BHST: Bottom Hole Static Temperature. Es la temperatura estática que tiene el pozo en ese momento.

CASE (Computer Aided Software Engineerin) Ingeniería de Software Asistida por Ordenador.

CEINPET: Centro de Investigaciones del Petróleo.

CUPET: Empresa Cuba Petróleo.

CU: Casos de Usos.

C++: Lenguaje de Programación.

C#: Lenguaje de Programación.

ECLIPSE: Herramientas de Programación de Escritorio.

E & P: Exploración y Producción.

FK: Es la llave foránea de una tabla en la Base de Datos.

GNU/GPL (General Public License): Licencia bajo la cual se distribuye el Sistema Operativo Gnu/Linux.

HTML: Lenguaje de Programación.

JAVA: Lenguaje de Programación.

LINUX: Sistema Operativo

MAC OS X: Sistema Operativo

MYSQL: Sistema Gestor de Base de Datos.

MD: Measure Depth. Es la profundidad total medida en el pozo.

MB (Megabyte): Unidad de medida de espacio en discos o memorias en las computadoras.

NET BEANS: Herramientas de Programación de Escritorio.

OLAP (Online Analytical Processing): Proceso Analítico en Línea.

OPEN SOURCE: Código abierto.

ORDBMS: de Gestión de Bases de Datos Objeto-Relacionales.

ORACLE: Sistema Gestor de Base de Datos.

PC (Personal Computer): Computador Personal.

PDVSA: Petróleos de Venezuela.

PgADMIN: Herramienta para la Administración de Base de Datos.

PgEXPLORER: Herramienta para la Administración de Base de Datos.

PHP: Lenguaje de Programación.

PHP PgADMIN: Herramienta para la Administración de Base de Datos.

PHP MyADMIN: Herramienta para la Administración de Base de Datos.

PK (Primary Key): Es la llave primaria de una tabla en la Base de Datos.

POSTGRESQL: Sistema Gestor de Base de Datos.

PSQL: Herramienta para la Administración de Base de Datos.

RAM (Random Access Memory): Es la memoria virtual de la computadora donde se guardan los datos de archivos abiertos durante el trabajo.

RUP: Proceso Unificado de Desarrollo de Software.

SGBD ó DBMS (por sus siglas en inglés): Sistemas Gestores de Base de Datos.

SQL (Structured Query Language): Lenguaje que se utiliza para realizar una consulta a la Base de Datos.

SQL SERVER: Servidor sobre el que trabaja la Base de Datos con el Lenguaje SQL.

TD (Total Dephts): Profundidad Total Del Pozo.

TVD (Total Vertical Dephts): Profundidad del Pozo por la Vertical.

TICs: Tecnologías de la Información y las Comunicaciones.

UML: Lenguaje Unificado de Modelado.

VISUAL BASIC: Lenguaje de Programación.

VISUAL ESTUDIO: Herramientas de Programación de Escritorio.