



INSTITUTO SUPERIOR MINERO METALURGICO
"Dr. Antonio Núñez Jiménez".
Facultad Geología - Minas
Moa, Holguín

Trabajo de diploma para optar por el título de Ingeniero Informático.

TEMA:

Aplicación Web para la Gestión de Informes de
Ensayos.

AUTOR:

Roilan Martínez Pérez

TUTOR:

Ing. Eloy R. Jiménez Iglesias

CONSULTANTE:

Beatriz Riverón Zaldívar

Moa, Holguín,
Cuba, Julio 2012
Año del 54 Aniversario del Triunfo de la Revolución

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez” para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los _____ días del mes de _____ del 2012.

Roilan Martínez Pérez

Ing. Eloy R. Jiménez Iglesias

Firma del Autor

Firma del Tutor

Pensamiento.



"Los estudiantes son en su mayoría revolucionarios. Revolucionarios por naturaleza, porque pertenecen a ese estrato de jóvenes que se abren a la vida y que adquieren todos los días conocimientos nuevos".

Ernesto Guevara de la Serna.

Dedicatoria

Este trabajo va dedicado especialmente a mis Padres, por la magnífica educación que me dieron, por apoyarme en todas mis decisiones, y porque sé que este es su mayor anhelo.

*A mi madre, **Ramona**,*

por demostrarme que en la vida no debemos dejar que nos venzan las dificultades y haberme dado ánimo para lograr que este sueño sea posible.

*A mi padre, **Ramón**,*

porque me inculcó los deseos de superación y me enseñó a enfrentar la vida con mis propias decisiones.

*A mi novia, **Odilennis**,*

por ser mi compañera y amiga en esta larga travesía, por ser una mujer maravillosa, la cual me ha dado su apoyo incondicional y ha tenido una gran paciencia durante todo este tiempo que hemos estado separados.

Agradecimientos

Quisiera comenzar agradeciendo: a mis padres que son lo más importante de mi vida, también por verme dado esta gran oportunidad de vivir y realizar este sueño.

También a mi novia, por verme esperado todo este tiempo y por darme las fuerzas necesarias para continuar adelante, antes cualquier obstáculo.

A Zaida y Gabriel que son como padres para mí, que siempre estuvieron ahí cuando los necesité y siempre me brindaron su apoyo, y también agradecerle a Leo, Carlos, Gabrielito.

No puede faltar el agradecimiento a los compañeros de aula y de cuarto que de una forma u otra son partícipes de que todo este trabajo sea posible: Lecusay, Yariel, Angel, Daniar, Yuniel, Leo, Leonel, Aluchy, Evelio, Rafael, Hinojosa.

Un agradecimiento especial para Yohandi por brindarme su ayuda incondicional, muchas gracias.

En fin agradecer a mi familia, a mis amigos, a todos los profesores que durante este período de mi vida universitaria se encargaron de formarme como un profesional.

Resumen

La formación del Ingeniero Geólogo en el ISMM de Moa reviste gran importancia para el país, ya que forma parte de un pequeño grupo de instituciones de educación superior donde se estudia esta carrera. En el proceso enseñanza aprendizaje (PEA) se han detectado insuficiencias en cuanto al déficit de medios de enseñanza para el estudio e interpretación de los ensayos de laboratorio de la asignatura Mecánica de suelos y rocas, donde esto va a influir directamente en la integración de los conocimientos de los futuros profesionales.

Este trabajo consiste en el desarrollo de una aplicación informática, que resuelva el problema anteriormente expuesto dentro del Instituto Superior Minero Metalúrgico de Moa (ISMMM), facilitando la rapidez y la eficiencia en este proceso.

Abstract

The Engineer's Geologist formation in the ISMM of Moa had great importance for the country, since it is part of a small group of institutions of superior education where this career is studied. In the process teaching learning (he/ BREAKS WIND) inadequacies have been detected as for the deficit of teaching means for the study and interpretation of the laboratory tests of the Mechanical subject of floors and rocks, where this will influence directly in the integration of the knowledge of the professional futures.

This work consists on the development of a computer application that solves the previously exposed problem inside the Institute Superior Mining Metallurgist of Moa (ISMMM), facilitating the speed and the efficiency in this process.

INDICE

Introducción.....	1
Aportes de la investigación:.....	4
Capítulo 1: Fundamentación teórica.....	6
Introducción:	6
1 Estado del Arte.	6
1.2 Sistemas informáticos existentes vinculados al campo de acción.	6
1.3 Tendencias y Tecnologías actuales para el desarrollo de aplicaciones. ..	6
1.3.1 ¿Qué es una aplicación Web? ¿Por qué la utilizamos?.....	6
1.3.2 Cualidades de las aplicaciones Web.	7
1.3.3. Servidor para aplicaciones WEB.	10
1.3.4 Lenguajes de programación.	11
1.3.5 Entornos de programación Web.	13
1.3.6 Sistemas gestores de bases de datos [SGBD].	14
1.4 Arquitectura para el desarrollo de Software.....	16
1.4.1 Patrones Arquitectónicos.	16
1.4.2 Arquitectura en tres capas.	18
1.5 Metodologías de desarrollo de Software.....	20
1.5.1 ¿Por qué elegir XP?.....	22
1.5.1.1 Características fundamentales de XP.....	22
1.6 Arquitectura a utilizar.	24
1.7 Tecnologías Utilizadas.....	24
1.7.1 PHP, HTML, JQuery y JAVASCRIPT Como Lenguajes de Programación.	24
1.7.2 PostgreSQL como Gestor de Base de Datos.	26
1.7.3 Como entornos de Desarrollo.	26
1.8 Herramientas Utilizadas.....	27
Capítulo 2: Planificación y Diseño.	29
Introducción:	29
2.1 Personal relacionado con el Sistema.....	29
2.2 Requisitos Funcionales.....	29
2.2.1 Requisitos No Funcionales.	30
2.3 Historias de usuarios.	31
2.4 Planificación de entregas.....	32
2.4.1 Estimación de esfuerzo por HU.	33
2.4.2 Plan de entregas.....	34
2.4.3. Plan de duración de las Iteraciones.	34
2.5-Tarjetas CRC.....	36
2.5.1 Tarjeta C.R.C Módulo #1: Gestionar Usuarios.....	37
Capítulo 3: Implementación y Pruebas.	39
Introducción:	39
3- Implementación.	39
3.1 Tareas por HU	40
3.1.1 Tareas de la HU #1	40
3.2-Pruebas	40
3.2.1 Pruebas de Aceptación (PA).....	40
3.2.2 Pruebas del módulo #1: Gestionar Usuarios.	41
3.3 Conclusiones del Capítulo.	43

Capítulo 4: Estudio de Factibilidad del Proyecto.....	44
Introducción:	44
4- Efectos Económicos.	44
4.1 Beneficios y Costos Intangibles en el proyecto.....	46
4.2 Ficha de costo.	46
4.3 Conclusiones del Capítulo.	49
Conclusiones Generales.....	50
Recomendaciones.	51
Referencias Bibliográficas.	52
Anexo 1 Historias de Usuarios.	54
Anexo-2 Tarjetas CRC	60
Anexo-3 Tareas de Ingeniería.	61
Anexo-4 Pruebas de Aceptación.	68

Índice de Figuras y Tablas

<u>Figura 1: Modelo Vista Controlador.....</u>	17
<u>Figura 2: Representación de la arquitectura en tres capas.....</u>	19
<u>Tabla 1: Personal relacionado con el sistema.....</u>	29
<u>Tabla 2: Plantilla de historia de usuario.....</u>	31
<u>Tabla 4: Estimación de esfuerzo por HU.....</u>	33
<u>Tabla 5: Plan de entregas.....</u>	34
<u>Tabla 6: Plan de duración de las iteraciones.....</u>	34
<u>Tabla7: Tabla de releases.....</u>	36
<u>Tabla 8: Plantilla de tarjeta CRC.....</u>	36
<u>Tabla 9: Tarjeta CRC Autenticar usuarios.....</u>	37
<u>Tabla10: Plantilla de Tareas de Ingeniería.....</u>	39
<u>Tabla11: Tareas de Ingeniería.....</u>	40
<u>Tabla12: Planilla de Prueba de Aceptación.....</u>	41
<u>Tabla13: P.A Gestionar usuarios del sistema.....</u>	41
<u>Tabla14: Costo en Moneda Librementemente Convertible.....</u>	46
<u>Tabla15: Costo en Moneda Nacional.....</u>	47
<u>Tabla 16: HU Autenticar usuario.....</u>	54
<u>Tabla 17: HU Mostrar determinación del límite líquido.....</u>	54
<u>Tabla 18: HU Mostrar determinación del límite plástico.....</u>	54
<u>Tabla 19: HU Mostrar resultados del ensayo.....</u>	55
<u>Tabla 20: HU Obtener gráfica de Humedad versus # de Golpes.....</u>	55
<u>Tabla 21: HU Mostrar determinación de hinchamiento controlado.....</u>	56
<u>Tabla 22: HU Mostrar determinación de humedades.....</u>	56
<u>Tabla 23: HU Mostrar propiedades iniciales y finales.....</u>	56
<u>Tabla 24: HU Mostrar resultados del ensayo.....</u>	57
<u>Tabla 25: HU Mostrar gráfico de Presión Axial versus Logaritmo del tiempo... </u>	57
<u>Tabla 26: HU Mostrar datos iniciales del ensayo.....</u>	57
<u>Tabla 27: HU Mostrar resultados del ensayo.....</u>	58
<u>Tabla 28: HU Mostrar ángulo de reposo en suelos arenosos.....</u>	58
<u>Tabla 29: HU Mostrar ángulo de reposo en suelos sumergidos.....</u>	59
<u>Tabla 30: HU Mostrar resultados del ensayo.....</u>	59
<u>Tabla 31: Tarjeta C.R.C. Gestionar usuarios.....</u>	60

<u>Tabla 32: Tarjeta C.R.C. Mostrar resultados</u>	60
<u>Tabla 33: Tarjeta C.R.C. Gráfica Humedad versus # Golpes</u>	61
<u>Tabla 34: Tarea de Ingeniería Insertar usuarios</u>	61
<u>Tabla 35: Tarea de Ingeniería Eliminar usuarios</u>	61
<u>Tabla 36: Tarea de Ingeniería Modificar usuarios</u>	62
<u>Tabla 37: Tarea de Ingeniería Visualizar usuarios</u>	62
<u>Tabla 38: Tarea de Ingeniería Loguear usuarios</u>	62
<u>Tabla 39: Tarea de Ingeniería Mostrar determinación límite líquido</u>	63
<u>Tabla 40: Tarea de Ingeniería Mostrar determinación límite plástico</u>	63
<u>Tabla 41: Tarea de Ingeniería Mostrar resultados del ensayo</u>	63
<u>Tabla 42: Tarea de Ingeniería Obtener gráfica de Humedad vs # de Golpes</u> .	64
<u>Tabla 43: Tarea de Ingeniería Mostrar hinchamiento controlado</u>	64
<u>Tabla 44: Tarea de Ingeniería Mostrar determinación de humedad</u>	64
<u>Tabla 45: Tarea de Ingeniería Mostrar propiedades iniciales y finales</u>	65
<u>Tabla 46: Tarea de Ingeniería Mostrar resultados del ensayo</u>	65
<u>Tabla 47: TI Obtener gráfico de Presión Axial/Logaritmo del tiempo</u>	65
<u>Tabla 48: Tarea de Ingeniería Mostrar datos iniciales del ensayo</u>	66
<u>Tabla 49: Tarea de Ingeniería Mostar resultados del ensayo</u>	66
<u>Tabla 50: Tarea de Ingeniería Mostar ángulo de reposo en suelos arenosos</u> ..	66
<u>Tabla 51: Tarea de Ingeniería Mostrar ángulo de reposo en suelos sumergidos</u>	67
<u>Tabla 52: Tarea de Ingeniería Mostrar resultados del ensayo</u>	67
<u>Tabla 53: Prueba para comprobar la Autenticación de los usuarios</u>	68
<u>Tabla 54: Prueba de aceptación No.3 Visualización del límite de contracción</u> ..	68
<u>Tabla 55: Prueba de aceptación No.4 Visualizar los resultados del ensayo</u> ...	69
<u>Tabla 56: Prueba de aceptación No.5 Visualizar determinación límite líquido</u> ..	69
<u>Tabla 57: Prueba de aceptación No.6 Visualizar determinación límite plástico</u>	69
<u>Tabla 58: Prueba de aceptación No.7 Visualizar resultados del ensayo</u>	70
<u>Tabla 59: Prueba de aceptación No.8 Visualizar Ángulo de reposo Seco</u>	70
<u>Tabla 60: Prueba de aceptación No.9 Visualizar Ángulo de reposo sumergido</u>	71
<u>Tabla 61: Prueba de aceptación No.10 Visualizar resultados del ensayo</u>	71
<u>Tabla 62: Prueba de aceptación No.11 Visualizar Hinchamiento Controlado</u> .	71

<u>Tabla 63: Prueba de aceptación No.12 Visualizar Determinación humedad....</u>	72
<u>Tabla 64: Prueba de aceptación No.13 Visualizar Propiedad Inicial y Final.....</u>	72
<u>Tabla 65: Prueba de aceptación No.14 Visualizar resultados del ensayo.....</u>	73
<u>Tabla 66: Prueba de aceptación No.15 Visualizar Gráfico obtenido.</u>	73
<u>Tabla 67: Prueba de aceptación No.16 Visualizar Gráfico obtenido.</u>	74

Introducción

La función del profesor es insustituible, por su incidencia fundamental en la labor educativa, en la formación de valores y en la conducción del proceso de enseñanza-aprendizaje; este papel encuentra refuerzo en la preparación de documentos y/o materiales didácticos y recursos tecnológicos, que tributan en la práctica a muchas de las funciones de los docentes: orientación, motivación, transmisión, recordación, indagación, discusión, retroalimentación y evaluación, entre otras.

Con la implementación del plan D se requiere de medios de enseñanza con características específicas, que favorezcan la actividad independiente del estudiante, proporcionándole orientaciones metodológicas y bibliográficas para que pueda dominar el sistema de conocimientos de forma lógica y estructurada, a partir de sus propias estrategias de aprendizaje.

La formación del Ingeniero Geólogo en el ISMM de Moa reviste gran importancia para el país, ya que forma parte de un pequeño grupo de instituciones de educación superior donde se estudia esta carrera. En el plano de análisis más cercano la implantación del nuevo plan perfeccionado en la carrera de Ingeniería Geológica del ISMM (plan C) se llevó a cabo con el objetivo de formar un profesional de perfil amplio, el cual se debe caracterizar por tener un dominio profundo de la formación básica de la profesión y ser capaz de resolver en la base los problemas más generales que se le presenten una vez graduado.

El Instituto Superior Minero Metalúrgico de Moa (ISMMM), se ha trazado una política de automatizar la gran parte de los procesos docentes y extracurriculares por los que atraviesa la institución. El centro no cuenta con una aplicación que facilite la realización del proceso de gestión de informes de ensayos de laboratorios de Mecánica de suelos y rocas en la disciplina de Geología Aplicada por lo que este proceso se hace tedioso, y se ha dado la tarea de implementar una aplicación informática que resuelva este proceso.

Situación Problemática

Actualmente la carrera de Ingeniería Geológica se encuentra en el Plan D en sus tres primeros años y el resto en Plan C perfeccionado; en el proceso enseñanza aprendizaje (PEA) se han detectado insuficiencias en cuanto a la déficit de medios de enseñanza para el estudio e interpretación de los ensayos de laboratorio de la asignatura Mecánica de suelos y rocas, esto influye directamente en la integración de los conocimientos, a partir de la utilidad de los medios para la implementación de software, lo cual constituye prioridad en el trabajo metodológico del Colectivo de Carrera.

Entre las insuficiencias detectadas se encuentran:

- Se utilizan parcialmente las herramientas informáticas para elevar la calidad de las clases y facilitar el desarrollo del PEA.
- No se explota suficientemente el uso de las TIC por diversas razones:
 - Es limitado el acceso a los materiales que sirven de apoyo para la interpretación de los ensayos de laboratorios.
 - Ineficiente organización de los materiales bibliográficos (impresos y digitales), lo que dificulta el manejo de la información que contienen las normas.
 - La cantidad de ordenadores no es suficiente.
 - Insuficientes paquetes informáticos para el cálculo e interpretación de los ensayos de laboratorios de Mecánica de suelos y rocas.

A partir de los elementos anteriormente expuestos se declara como **problema**: Necesidad de informatizar la gestión e interpretación de los ensayos de laboratorios de Mecánica de suelos y rocas en la disciplina de Geología Aplicada.

Como **objeto de estudio de la investigación** se plantea:

La informatización de los procesos de gestión de información.

Campo de acción:

Informatización de la gestión de informes de ensayos de laboratorios de Mecánica de suelos y rocas en la carrera de Ingeniería Geológica.

Sobre la base del problema a resolver se establece la siguiente **Idea Científica a defender:**

Si se desarrolla e implementa una herramienta informática para la gestión de los informes de ensayos de laboratorio de Mecánica de suelos y rocas en la carrera de Ingeniería Geológica, favorecerá y mejorará el desarrollo del PEA.

En correspondencia con la Idea Científica planteada, se define como **objetivo general del trabajo:**

Elaborar una aplicación web para la gestión e interpretación de los ensayos de laboratorio de Mecánica de suelos y rocas en la carrera de Ingeniería Geológica.

Objetivos específicos:

- Realizar un análisis del estado actual del funcionamiento de la gestión de informes de ensayos.
- Realizar un estudio de las herramientas a utilizar para el diseño e implementación de la aplicación.
- Implementar una aplicación que permitirá realizar estudios e interpretaciones a raíz de los resultados obtenidos en los ensayos.
- Documentar el software implementado.

Para lograr el cumplimiento de los objetivos específicos declarados, se plantean las siguientes **tareas:**

- Estudio de los referentes teóricos acerca de los procesos de gestión de informes de ensayos.
- Análisis de las tendencias y tecnologías actuales.

- Selección del lenguaje de programación, metodología y herramientas para el desarrollo de la aplicación.
- Diseño e implementación de la herramienta informática para la gestión de informes de ensayos de laboratorio de Mecánica de suelos y rocas en la carrera de Ingeniería Geológica en el ISMM.
- Valoración de los resultados de la implementación del sistema creado.
- Confección del manual de usuario.

Entre los métodos y técnicas de investigación aplicados principalmente están:

Métodos teóricos:

- **Histórico y lógico:** para la búsqueda de antecedentes del software, las herramientas utilizadas, así como la forma de cómo se llevaba a cabo el cálculo e interpretación de los ensayos de laboratorio de Mecánica de suelos y rocas en la carrera de ingeniería geológica.
- **Observación:** Se empleó para caracterizar y analizar el previo funcionamiento del problema planteado, así como para llevar a cabo la implementación y diagnóstico del resultado obtenido en la investigación.

Métodos Empíricos:

- Entrevistas para determinar los requerimientos funcionales del sistema informático que se quiere construir. Se llevó a cabo un diálogo con personas expertas en la materia.

Aportes de la investigación:

Este trabajo tiene como **aporte práctico** la elaboración de software para la interpretación de los ensayos de laboratorios de mecánica de suelos y rocas.

El presente trabajo está estructurado por un resumen, introducción general, cuatro capítulos, conclusiones generales, recomendaciones, bibliografías y anexos:

Capítulo 1 “Fundamentación Teórica”: Se ofrece una breve descripción de diferentes conceptos imprescindibles que le sirven de pilar a la investigación que se lleva a cabo. Así como algunos elementos importantes de las herramientas a utilizar para la implementación del sistema.

Capítulo 2 “Planeación y Diseño”: se hace uso de la metodología expuesta en el capítulo inicial para el desarrollo del proyecto, abordando sus dos primeras fases. Se desarrollaron las tarjetas CRC.

Capítulo 3 “Implementación y Pruebas”: se presentan la implementación de las tarjetas de ingeniería así como las pruebas realizadas con sus resultados.

Capítulo 4 “Estudio de Factibilidad”: se realiza un estudio de los esfuerzos requeridos para la realización del sistema, y se valora la sostenibilidad del producto.

Capítulo 1: Fundamentación teórica.

Introducción:

En este capítulo abordaremos los conceptos fundamentales para el desarrollo del proceso de cálculo e interpretación de los ensayos de laboratorio de Mecánica de suelos y rocas, sus principales características, así como las herramientas que se pueden utilizar para obtener un buen producto. Haremos un estudio de los diferentes Gestores de Base de Datos, lenguajes de programación y metodologías existentes para determinar cuáles son más convenientes. Además se expone la metodología de Ingeniería de Software que se usará para el desarrollo de la aplicación.

1 Estado del Arte.

1.2 Sistemas informáticos existentes vinculados al campo de acción.

Durante el período investigativo del presente trabajo, no se encontró referencia alguna de un sistema que pueda gestionar el proceso en cuestión, ya sea nacional o internacionalmente.

1.3 Tendencias y Tecnologías actuales para el desarrollo de aplicaciones.

Ante el incesante avance de las tecnologías, la sociedad, ávida de nuevas herramientas y funcionalidades, exige a los desarrolladores de software nuevos retos y nuevas concepciones para satisfacer sus exigencias, cada vez más ambiciosas. Para satisfacer estas exigencias, los desarrolladores deben buscar nuevas ideas surgiendo así nuevas metodologías y formas de desarrollo que permiten confeccionar productos cada vez más complejos. (Cala, 2011)

1.3.1 ¿Qué es una aplicación Web? ¿Por qué la utilizamos?

Una aplicación Web es un sistema informático que los usuarios usan accediendo a un servidor Web a través de los protocolos de Internet. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero. La facilidad para actualizar y mantener las aplicaciones

Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su creciente popularidad. Una aplicación Web está comúnmente estructurada como una aplicación en tres-capas. En su forma más común el navegador es la primera capa, un motor usando alguna tecnología de Web dinámica (Ej.: CGI, PHP, ASP, Java Servlets o Pearl) es la capa del medio, y algún tipo de gestor de bases de datos como última capa, consultas y actualizaciones a la bases de datos generando una interfaz de usuarios. (Subiros, 2009)

En tiempos recientes se ha usado la estrategia de generalizar esta arquitectura mediante la adición de piezas de hardware que permitan balancear la carga de los servidores Web y de las aplicaciones.

Las Aplicaciones Web son de un desarrollo poco costoso, sencillo y rápido. Presenta acceso ubicuo, sin necesidad de distribución e idealmente, con pocos requerimientos técnicos. Con datos centralizados y fácil integración de datos múltiples fuentes. Debido a estas ventajas que presenta es la razón por la cual es utilizada en este trabajo para llevar a cabo la implementación del software. (Subiros, 2009)

1.3.2 Cualidades de las aplicaciones Web.

Para que la Web sea efectiva, debe poseer algunas cualidades indispensables, como la seguridad, la escalabilidad, portabilidad y un diseño eficiente de la interfaz de usuario. La escalabilidad y la portabilidad son las que generalmente caen en olvido.

Escalabilidad: la aplicación tendrá la capacidad de crecer junto a las necesidades de sus usuarios.

Portabilidad: se enlaza el concepto de escalabilidad, y se refiere a la compatibilidad de la aplicación con múltiples sistemas operativos y plataformas utilizadas en el mercado. En general, cuando la demanda de los usuarios crece, las aplicaciones deben cambiar de plataforma para pasar de un servidor pequeño a uno mediano o grande. Es entonces, cuando la portabilidad es

crítica para poder “escalar” sin problemas independientemente del entorno que se requiera.

Sin dudas, las aplicaciones Web aún tienen un largo camino por recorrer, debido a que son una opción muy interesante, especialmente, en tareas donde las bases de datos juegan un papel predominantes y los usuarios se encuentran dispersos. (Vargas, 2011)

Ventajas Web: (Cala, 2011)

Compatibilidad multiplataforma: Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.

Actualización: Las aplicaciones basadas en Web están siempre actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o interferir con sus hábitos de trabajo, pues no se hace necesario iniciar nuevas descargas y/o procedimientos de instalación (algunas veces imposible cuando usted está trabajando dentro de grandes organizaciones).

Inmediatez de acceso: Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y están listas para trabajar sin importar cuál es su configuración o su hardware.

Menos requerimientos de memoria: Las aplicaciones basadas en Web tienen menos demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, esas aplicaciones basadas en Web usan en muchos casos la memoria de las computadoras donde ellas corren, dejando más espacio para correr múltiples aplicaciones sin incurrir en frustrantes deterioros en el rendimiento.

Los datos también van online: El hecho de que el manejo de los datos sea realizado de forma remota libra al usuario de la responsabilidad en la protección de los mismos, y al mismo tiempo logra que los recursos sean accesibles en cualquier momento.

Múltiples usuarios concurrentes: Las aplicaciones basadas en Web pueden ser utilizadas por múltiples usuarios al mismo tiempo. No hay más necesidad de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver e incluso editar el mismo documento de manera conjunta.

Los datos son más seguros: Si bien la ruptura de discos no va a desaparecer, es probable que los usuarios escuchen mucho menos del tema. A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, granjas de almacenamiento de datos redundantes, altamente fiables, los usuarios van a tener mucho menos riesgo de perder sus datos debido a una ruptura de disco impredecible o a un virus de la computadora. Las compañías que proveen aplicaciones basadas en Web van a brindar amplios servicios de resguardo de datos ya sea como una parte integral del servicio básico o como una opción paga.

Desventajas WEB: (Cala, 2011)

- Acceso limitado, la necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- La interactividad no se produce en tiempo real, en las aplicaciones Web cada acción del usuario conlleva un tiempo de espera hasta que se obtiene la reacción del sistema.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.

- Diferencias de presentación entre plataformas y navegadores. La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

1.3.3. Servidor para aplicaciones WEB.

El servidor Web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor Web buscará una página Web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición. (Cala, 2011)

Apache: sustancialmente, es un proyecto nacido para crear un servidor de web estable, fiable y veloz para plataformas Unix. Apache nace, por una parte, de un código ya existente y de una serie de parches (patch) para mejorar su fiabilidad y sus características; de ahí su nombre. (Cala, 2011)

Características: (Cala, 2011)

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita con un código fuente disponible. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esta característica le ofrece al software un grado de transparencia tal que es posible determinar en todo momento qué es lo que se está instalando, sin secretos ni puertas traseras.
- Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que se instalen cuando se necesiten. Otra cosa importante es que cualquiera que posea alguna experiencia en la programación de C o Perl puede escribir un módulo para realizar una función determinada.

- Trabaja con Perl, PHP y otros lenguajes de script. Perl destaca en el mundo del script y Apache utiliza su parte del pastel de Perl tanto con soporte CGI como con soporte mod perl. También trabaja con Java y páginas JSP. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurarlo para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de logs³⁰. Permite la creación de ficheros de log a la medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.
- Se pueden extender las características de Apache hasta donde la imaginación y los conocimientos lleguen, debido a que el equipo de desarrollo está formado por voluntarios, diseminados por todo el mundo, que sigue manteniendo este servidor de web libre.

1.3.4 Lenguajes de programación.

PHP (Personal Home Page) es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. Actualmente PHP se encuentra en su versión 5, que utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades de las aplicaciones Web actuales. (López, 2011)

PHP es un lenguaje de programación de estilo clásico, esto quiere decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser HTML o XML.

A diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor, como por ejemplo podría ser, una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML. (López, 2011)

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable, puesto que cada día son muchísimas más las páginas Web que lo utilizan para su funcionamiento, según las estadísticas, PHP se utiliza en más de 10 millones de páginas, y cada mes realiza un aumento del 15%. Como síntesis, PHP corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 Bases de Datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de aplicaciones web muy robustas, y contiene unas 40 extensiones estables sin contar las que se están experimentando, también tiene soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros. (López, 2011)

HTML: Lenguaje de Marcas de Hipertexto (Hypertext Markup Language) por sus siglas en inglés, es el lenguaje de marcado predominante para la construcción de páginas Web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores Web y otros procesadores de HTML. (López, 2011)

HTML también es usado para referirse al contenido del tipo de MIME text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del XML (como XHTML 1.0 y posteriores) o en forma

descendida directamente de SGML (como HTML 4.01 y anteriores).

HTML consiste de varios componentes vitales, incluyendo elementos y sus atributos, tipos de data, y la declaración de tipo de documento. (López, 2011)

JavaScript: este es un lenguaje interpretado por lo que no requiere compilación. Fue creado por la empresa Netscape Communication. Es similar al Java, aunque no es un lenguaje orientado a objetos, no dispone de ninguna de los privilegios de la programación orientada a objetos (POO). La mayoría de los navegadores en sus últimas versiones interpretan JavaScript. El código de JavaScript puede ser integrado a nuestras páginas Web.

Ventajas:

- Lenguaje de scripting fiable y seguro.
- Los scripts tienen capacidad limitada, por razones de seguridad.
- El código se ejecuta en lado del cliente.

Desventajas:

- Código visible por cualquier usuario.
- El código debe descargarse completamente.

Puede poner en riesgo la seguridad del sitio.

1.3.5 Entornos de programación Web.

Macromedia Dreamweaver: sin lugar a dudas, es una de las herramientas más utilizadas por los webmasters para el trabajo con aplicaciones visuales en este caso en el diseño e implementación de páginas Web. Se adapta increíblemente a las necesidades de todo tipo de profesional de diseño Web, tanto para lo que prefieren programar el código directamente en el editor de texto como para los que gustan del ambiente visual. (Vargas, 2011)

Se trata de un editor de texto especialmente diseñado para trabajar con documentos Web como HTML, PHP, ASP, JavaScript, entre otros. (Ruiz, 2004)

PhpDesigner: Es un completo entorno de desarrollo y programación especialmente diseñado para los “gurús” de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL. Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que se puede personalizar con nada menos que dieciocho temas distintos. Cuenta con cliente de FTP y navegador de ficheros integrado, utilidades de corrección y autocompletado, búsqueda integrada en google y soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación. PhpDesigner soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL. (López, 2011)

1.3.6 Sistemas gestores de bases de datos [SGBD].

Un Sistema Gestor de Bases de Datos (SGBD) o DBMA es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos.

Un SGBD debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla, generar informes.

Características de un Sistema Gestor de Base de Datos: (Cala, 2011)

Abstracción de la información: Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios niveles de abstracción.

Independencia: La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.

Redundancia mínima: Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias.

Consistencia: En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.

Seguridad: La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra protegida frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipularla o destruirla; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.

Integridad: Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.

Respaldo y recuperación: Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.

Actualmente existen muchos sistemas gestores de bases de datos, entre ellos, analizaremos las características y facilidades que brinda cada uno de los que se han tenido en consideración, los que siguen son: MySQL y PostgreSQL.

MySQL: es un sistema de gestión de base de datos relacional, multihilo y multiusuario, con más de seis millones de instalaciones. Al contrario de proyectos como el Apache, donde el software es desarrollado por una comunidad pública, y el copyright del código está en poder del autor individual, MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. (Leyva, 2011)

MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows. (Leyva, 2011)

PostgreSQL: es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la universidad de Berkeley. Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. (Leyva, 2011)

PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, no es un sistema de gestión de bases de datos puramente orientado a objetos. (Leyva, 2011)

PostgreSQL está considerado como la base de datos de código abierto más avanzada del mundo y proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. (Leyva, 2011)

1.4 Arquitectura para el desarrollo de Software.

1.4.1 Patrones Arquitectónicos.

Para el diseño de aplicaciones con sofisticadas interfaces se emplea el patrón de diseño MCV. La lógica de una interfaz de usuario cambia con más

frecuencia que los almacenes de datos y la lógica de negocio. (Basulto, 2010)

Si se realiza un diseño ofuscado, es decir, una forma de mezclar los componentes de interfaz y de negocio, entonces, la consecuencia será que, cuando se necesite cambiar la interfaz, tendrá que modificarse trabajosamente los componentes de negocio, por lo que propiciará mayor trabajo y más riesgo de error. (Cala, 2011)

Se trata de realizar un diseño que desacople la vista del modelo, con el fin de perfeccionar la reusabilidad. De este modo las modificaciones son las que impactan en menor medida en la lógica de negocio o de datos. (Cala, 2011)

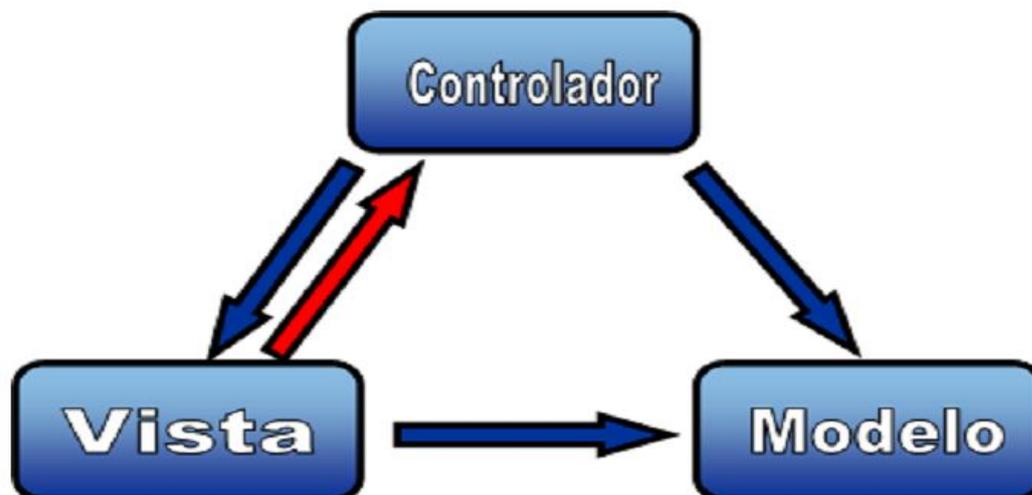


Figura 1: Modelo Vista Controlador.

- **Modelo:** Representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- **Vista:** Presenta el modelo en un formato adecuado, como en una página Web que le permite al usuario interactuar con ella, usualmente un elemento de interfaz de usuario.
- **Controlador:** Responde a eventos, usualmente acciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación

(la vista) logrando un mantenimiento más rápido y sencillo de las aplicaciones.

Ejemplo, para el caso de la Web, si se fuera a mostrar una misma aplicación en un navegador estándar, como en un navegador de un dispositivo móvil, sólo es necesario crear una vista nueva por cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (Aplicación de escritorio, HTTP, consola de comandos, e-mail, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. (Vargas, 2011)

Ventajas del MVC

- La separación del Modelo de la Vista, es decir, separa los datos de la representación visual de los mismos.
- Crea independencia de funcionamiento.
- Facilita el mantenimiento en caso de errores.
- Permite el escalamiento de la aplicación en caso de ser requerido.

1.4.2 Arquitectura en tres capas.

Arquitectura en capas: es donde se define como organizar el modelo de diseño a través de capas, que pueden estar físicamente distribuidas, lo que quiere decir que los componentes de una capa solo puede hacer referencia a componentes en capas inmediatamente inferiores. Este patrón es importante porque simplifica la comprensión y la organización de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no son consistentes de ningún detalle o interfaz de las superiores. La programación por capas es un estilo de programación en el que el objetivo primordial es separar la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación del usuario. (Vargas, 2011)

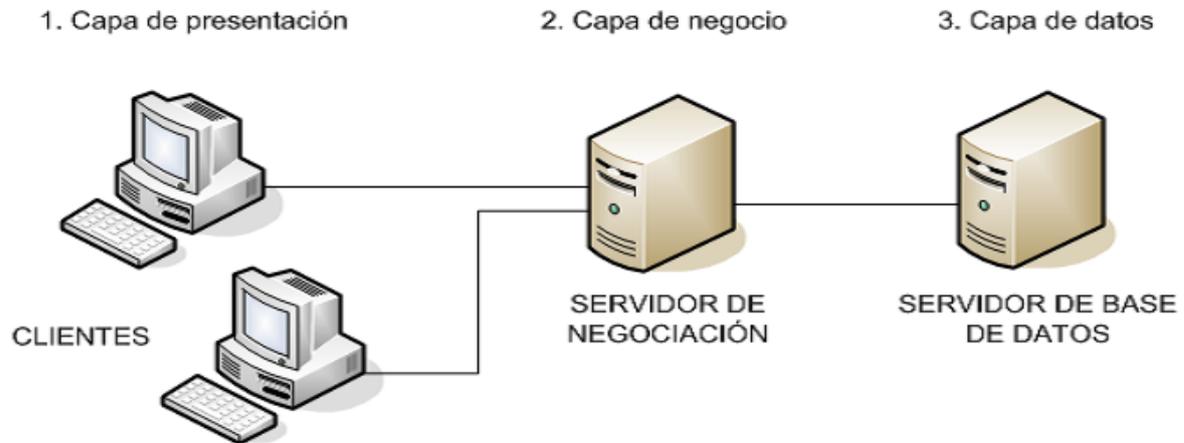


Figura 2: Representación de la arquitectura en tres capas. Capas o niveles.

- **Capa de presentación o interfaces:** es la capa que le permite al usuario interactuar con el sistema, captura y le comunica la información al mismo, dando un mínimo de proceso (realiza un filtrado previo para asegurarse que no haya errores de formato). Esta capa se comunica únicamente con la de negocio. (Vargas, 2011)
- **Capa de negocio:** es donde reside los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitarle al gestor de bases de datos almacenar o recuperar datos de él. (Vargas, 2011)
- **Capa de datos:** es donde residen los datos y es la encargada de acceder ellos. Está formada por uno o más gestores de bases de datos que se encargan de realizar el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. (Vargas, 2011)

Todas estas capas pueden residir en un mismo ordenador aunque no es lo

típico. Lo más usual es que haya una multitud de ordenadores donde reside la capa de interface (son los clientes de la arquitectura cliente/servidor). Las capas de negocios y de datos pueden residir en un mismo ordenador, y si el crecimiento de las necesidades lo aconseja, pueden dividirse en dos o más ordenadores. Así, si el tamaño o complejidad de las base de datos aumenta, pueden dividirse en varios ordenadores los cuales recibirán las peticiones del ordenador donde resida la capa de negocio. Si por el contrario, la complejidad fuese en la capa de negocio lo que obligase a la separación, esta lógica del negocio podría residir en uno o más ordenadores que realizarían las solicitudes a una única base de datos. (Vargas, 2011)

Ventajas:

- El estilo soporta un diseño basado en niveles de abstracción crecientes, lo cual, permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
- El estilo admite muy naturalmente optimizaciones y refinamientos.
- Proporciona una amplia reutilización. Al igual que los tipos de datos abstractos, se pueden utilizar diferentes implementaciones o versiones de una misma capa a medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto conduce a la posibilidad de definir interfaces de capa estándar, a partir de las cuales pueden construirse extensiones o presentaciones específicas. (Rodríguez, 2008)

1.5 Metodologías de desarrollo de Software.

Hoy en día, llevar a cabo el desarrollo de un buen software depende de un gran número de actividades y etapas donde elegir la mejor metodología, para el equipo influye directamente en el futuro éxito del producto. El papel preponderante de las metodologías es sin duda esencial en un proyecto y en el paso inicial, que debe encajar en el equipo, guiar y organizar actividades que conlleven a las metas trazadas en el grupo.

Una metodología para el desarrollo de un proceso de software es un conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de sistemas informáticos. Las metodologías existentes en la actualidad se dividen en dos grandes grupos atendiendo a sus características: las metodologías tradicionales (RUP, MSF) y las metodologías ágiles (XP, SCRUM). Las primeras están pensadas para el uso exhaustivo de documentación durante todo el ciclo del proyecto mientras que las segundas ponen vital importancia en la capacidad de respuesta a los cambios, la confianza en las habilidades del equipo y al mantener una buena relación con el cliente. (RUP) Teniendo en cuenta ambos enfoques damos paso al análisis de dos de las metodologías más usadas actualmente. (Vargas, 2011)

- RUP. (Proceso Unificado del Rational)-Es una metodología para proyectos más largos, debido a su gran cantidad de diagramas que lleva consigo, además se documenta poco sobre el Sistema que se está llevando a cabo.
- XP. (Programación Extrema)-Metodología que adopta 12 prácticas que se pueden utilizar todas o no, eso lo deciden el programador y el cliente según las necesidades de este último o si la aplicación no requiere de todas. Se centra especialmente en documentar en forma de plantillas, tiene cuatro fases: Planeación, Diseño, Desarrollo o Implementación y Pruebas. En la primera fase se generan como artefactos los usuarios del negocio, las historias de usuarios, la lista de reserva del producto, el plan de iteraciones, entre otros. En la segunda se tiene el modelo de datos, tarjetas CRC. En tercera fase se desarrollaron las tareas de ingeniería y la cuarta fase son efectuadas las pruebas al software para verificar que el mismo cumpla con todas las funcionalidades acordadas, estas pruebas pueden ser aceptadas por el cliente o denegadas por el mismo.
- SXP. (Scrum + Programación Extrema). Esta Metodología fue elaborada en la Universidad de las Ciencias Informáticas, escogieron una parte de

Scrum y otra parte de XP, para una mayor organización dividen en carpetas todas sus plantillas. Pero no existe una documentación amplia de esta metodología como en XP.

1.5.1 ¿Por qué elegir XP?

XP, es una metodología ágil, centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, se preocupa por el aprendizaje de los desarrolladores, y propicia un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Vargas, 2011)

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, que describe la filosofía de XP en el "Manifiesto Ágil" 5, sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea. (Vargas, 2011)

1.5.1.1 Características fundamentales de XP.

- Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.
- Pruebas unitarias: continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación.
- Programación por parejas: se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código está escrito de esta manera –el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.

- Frecuente interacción: del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- Refactorización del código: es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.
- Propiedad del código compartida: en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.
- Simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo. (Letelier, 2008)

La simplicidad y la comunicación son extraordinariamente complementarias. Con más comunicación resulta más fácil identificar qué se debe y qué no se debe hacer. Mientras más simple es el sistema, menos tendrá que comunicar sobre este, lo que lleva a una comunicación más completa, especialmente si se puede reducir el equipo de programadores.

Ventajas:

- Apropiado para entornos volátiles.
- Estar preparados para el cambio, significa reducir su coste.
- Planificación más transparente para los clientes, ya conocen las fechas de entrega de funcionalidades. Vital para su negocio.

- Permite definir en cada iteración cuales son los objetivos de la siguiente.
- Permite la retroalimentación.
- La presión está a lo largo de todo el proyecto y no en una entrega final.

1.6 Arquitectura a utilizar.

Una arquitectura es el conjunto de decisiones significativas sobre la organización del sistema software, la selección de los elementos estructurales y sus interfaces, con los que se compone el sistema, junto con su comportamiento tal como se especifica en las colaboraciones entre esos elementos, la composición de esos elementos estructurales y de comportamiento en subsistemas progresivamente más amplios, y el estilo de arquitectura que guía esta organización, estos elementos y sus interfaces, sus colaboraciones, y su composición. Ejemplo: los patrones de diseño relacionados con el diseño de los objetos y frameworks de pequeña y mediana escala, que son aplicables al diseño de una solución para conectar los elementos de gran escala que se definen mediante los patrones de arquitectura, y durante el trabajo de diseño detallado para cualquier aspecto del diseño local. También se conocen como patrones de micro-arquitectura. El patrón fachada, que se puede utilizar para proporcionar la interfaz de una capa a la siguiente. La arquitectura en capas o arquitectura n capas es la que se ha seleccionado para darle solución a la problemática planteada anteriormente, debido a las facilidades presentadas anteriormente.

1.7 Tecnologías Utilizadas.

1.7.1 PHP, HTML, JQuery y JAVASCRIPT Como Lenguajes de Programación.

Luego de hacer el análisis entre los lenguajes que implementan servicios Web, se decide utilizar el PHP embebido en el código HTML ya que:

- Está soportado en la mayoría de las plataformas de Sistemas Operativos, mientras que con ASP por ser propiedad de Microsoft no es multiplataforma.
- El PHP no tiene costo oculto, o sea que cuando se adquiere incluye un sin número de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros. En caso de que no se tengan las bibliotecas están se pueden encontrar gratis en Internet. En el caso de ASP forma parte del Internet Information Server que viene integrado en Windows NT-2000 Server con su elevado costo de adquisición.
- PHP y ASP son parecidos en cuanto a la forma de utilización, pero PHP es más rápido, gratuito y multiplataforma.

Los lenguajes HTML y JavaScript fueron utilizados para atender las peticiones del lado del cliente. (López, 2011)

JQuery: es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. (jquery, 2006)

Es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. La característica principal de la biblioteca es que permite cambiar el contenido de una página web sin necesidad de recargarla, mediante la manipulación del árbol DOM y peticiones AJAX. (jquery, 2006)

La librería jQuery en resumen nos aporta las siguientes ventajas:

- Manipulación de la hoja de estilos CSS.

- Nos hace transparente el soporte de nuestra aplicación para los navegadores principales.
- Nos provee de un mecanismo para la captura de eventos.
- Provee un conjunto de funciones para animar el contenido de la página en forma muy sencilla.
- Compatible con los navegadores Mozilla Firefox 2.0+, Internet Explorer.

1.7.2 PostgreSQL como Gestor de Base de Datos.

Se Escogió a Postgres como Gestor de Base de Datos teniendo en cuenta que es un servidor de base de datos relacional orientada a objetos de software libre, liberado bajo la licencia BSD y dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo.

Características

- Alta concurrencia.
- Amplia variedad de tipos nativos.
- Es altamente extensible: soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- Mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos y es capaz entonces de manejar.
- PostgreSQL soporta funciones que retornan "filas", donde la salida puede tratarse como un conjunto de valores que pueden ser tratados igual a una fila retornada por una consulta (query en inglés).
- Las funciones pueden ser definidas para ejecutarse con los derechos del usuario ejecutor o con los derechos de un usuario previamente definido.
- El concepto de funciones, en otros DBMS, son muchas veces referidas como "procedimientos almacenados" (stored procedures en inglés). (Urgelles,2011)

1.7.3 Como entornos de Desarrollo.

En la implementación del sistema, se utilizaron los dos entornos de programación antes descritos, Dreamweaver por las facilidades que brinda

para el trabajo de diseño Web y PhpDesigner 2008 v6.0.0 por las ventajas que proporciona a la hora de escribir algoritmos complejos en los cuales es necesario correr paso a paso el programa, además ambos entornos de desarrollo soportan el lenguaje de programación PHP , el cual fue seleccionado para el desarrollo de la aplicación por la características ventajosas que brinda en comparación con otros lenguajes de programación.

1.8 Herramientas Utilizadas.

XAMPP: Es un servidor independiente de plataforma, software libre, incluye el servidor Web Apache, los servidores de datos MySQL y SQLite, sus respectivos gestores phpMyAdmin y phpSQLiteAdmin, el intérprete del lenguaje homónimo PHP con los extras incluidos en Perl, el intérprete del lenguaje Perl, servidores de FTP como ProFTPD o FileZilla FTP Server, las estadísticas Webalizer y OpenSSL, eAccelerator, Freetype2, libjpeg, libpng, zlib, Ming, etc. entre otros. El programa esta liberado bajo la licencia GNU y actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y MacOSX. (XAMPP, 2008)

La aplicación se sustentará en una Arquitectura Cliente – Servidor de tres capas y se utilizaron algunos principios del Modelo-Vista-Controlador.

Luego del estudio realizado se arriba a la conclusión de que el Sistema estará guiado por la metodología de desarrollo XP por sus facilidades, documentación y flexibilidad, además es la que mejor se adapta a las condiciones de desarrollo del sistema propuesto.

1.9 Conclusiones del Capítulo.

En este capítulo se abordaron los conceptos fundamentales asociados al dominio del problema, relacionados con el objeto de estudio y el campo de acción. Además se realizó un estudio de lo más utilizado en cuanto a los múltiples lenguajes, metodologías y tecnologías para el desarrollo de nuestra aplicación. Se conceptualizó y caracterizó de forma flexible las herramientas y lenguajes para la implementación de la aplicación. Además se expuso el porqué de ajustarnos a las cualidades de la metodología de Ingeniería de Software XP.

Capítulo 2: Planificación y Diseño.

Introducción:

En este capítulo, se introduce la fase de planeación y diseño, donde se detallan las necesidades del cliente, se describen las funcionalidades que serán objeto de informatización mediante el empleo de las historias de usuarios (HU), se realiza una estimación del esfuerzo necesario para las mismas y se establece un plan de iteraciones necesarias sobre el sistema, para su terminación.

2.1 Personal relacionado con el Sistema.

Tabla 1: Personal relacionado con el sistema.

Personas relacionadas con el sistema	Justificación
Usuarios	Estudiantes de la carrera de Geología y otros que estén interesados en la materia.
Administrador	Esta es la persona que tiene conocimiento en la materia, y está encargada de la gestión de la información del mismo.

2.2 Requisitos Funcionales.

Después de conocer el personal relacionado, se procede a realizar el análisis de las funcionalidades que debe cumplir la aplicación para dar respuesta a los mismos. Para ello se enumerarán mediante una lista de reserva, las funcionalidades que el sistema debe ser capaz de cumplir.

De acuerdo a lo antes expuesto, el sistema debe ser capaz de:

1. Gestionar usuarios del sistema:

- Insertar usuarios.
- Modificar usuarios.
- Eliminar usuarios.
- Ver usuarios del sistema.

2. Autenticar usuario.

- Cambiar contraseña.

3. Ensayo determinación de Límites.

- Mostrar determinación del límite líquido.
- Mostrar determinación del límite plástico.
- Mostrar resultados del ensayo.
- Obtener gráfica de Humedad versus # de Golpes.

4. Ensayo Hinchamiento Controlado.

- Mostrar determinación de hinchamiento controlado.
- Mostrar determinación de humedad.
- Mostrar propiedades iniciales y finales.
- Mostrar resultado del ensayo.
- Obtener gráfico de Presión Axial versus Logaritmo del tiempo.

5. Ensayo Límite de Contracción.

- Mostrar datos iniciales y finales del ensayo.
- Mostrar resultados del ensayo.

6. Ensayo Ángulo de reposo.

- Mostrar ángulo de reposo en suelos arenosos.
- Mostrar ángulo de reposo en suelos sumergidos.
- Mostrar resultados del ensayo.

2.2.1 Requisitos No Funcionales.

- Seguridad: La aplicación debe ser capaz de tener acceso fácil.
- Software: El sistema debe contar con los programas necesarios para su buen comportamiento, entre estos encontramos un servidor que nos provea del gestor de base de datos seleccionado anteriormente y sea un intérprete homónimo del lenguaje PHP para su buen funcionamiento, recomendablemente XAMP, además debe contar con un sistema operativo capaz de soportar con facilidad el servidor seleccionado.
- Hardware: El computador que asumirá el rol de servidor debe ser una máquina capaz de responder a las exigencias de los requerimientos de

software aunque la computadora puede contar con un Processor: Intel Celeron, MMX, 433MHz una Memoria: 128 MB RAM.

- Usabilidad: El sistema será entendible por los usuarios, fácil de usar y manipular.

2.3 Historias de usuarios.

Las HU, son la técnica utilizada en XP para detallar los requisitos del software. Son el resultado directo del intercambio entre los usuarios y desarrolladores a través de reuniones donde las conocidas tormenta de ideas (brain storm) arrojan no solo los requerimientos, sino también las posibles soluciones; representan una forma rápida de administrar las necesidades de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para gestionarlos, debido a que un requerimiento de software es descrito de forma concreta y sencilla utilizando el lenguaje común del usuario.

Las HU permiten responder ágilmente a los requerimientos cambiantes y aunque se redactan desde las perspectivas de los clientes, también los desarrolladores pueden brindar ayuda en la identificación de las mismas. Para definir las se emplea la siguiente plantilla. (Vargas, 2011)

Modelo de planilla de historia de usuario.

Tabla 2: Plantilla de historia de usuario.

Historia de Usuarios	
Número No. Historia de usuario	Usuario: Usuario entrevistado para obtener la función requerida a automatizar.
Nombre: nombre de la historia de usuario que sirve para identificarla mejor entre los desarrolladores y el cliente.	
Prioridad en el negocio: Importancia: Alta / Media / Baja	Riesgo en desarrollo: Dificultad: Alta / Media / Baja
Puntos estimados: Estimación: de 1 a 3 puntos	Iteración asignada: Iteración a la que corresponde
Programador responsable: Nombre de encargado de programación.	

Descripción: Se especifican las operaciones por parte del usuario y las respuestas del sistema.
Observaciones: Algunas observaciones de interés, como glosario, información

HU No.1 Gestionar usuarios del sistema.

Tabla 3: HU Gestionar usuarios del sistema.

Historia de Usuarios	
Número: 1	Usuario: Administrador.
Nombre: Gestionar usuarios del sistema.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Roilan Martínez Pérez.	
Descripción: el administrador crea el perfil del usuario que va interactuar con el sistema.	
Observaciones: Confirmado.	

Las otras Historias de Usuarios se encuentran en el: [Anexo 1](#)

2.4 Planificación de entregas.

En esta parte se establece la prioridad de cada Historia de Usuario así como una estimación del esfuerzo necesario de cada una de ellas con el fin de determinar un cronograma de entregas. Las estimaciones de esfuerzo asociado a la implementación de las historias se establecen utilizando como medida, el punto. Un punto, equivale a una semana ideal de programación (6 días). Las historias generalmente valen de 1 a 3 puntos. Por otra parte, se mantiene un registro de la “velocidad” de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración.

La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará

implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación.

2.4.1 Estimación de esfuerzo por HU.

Tabla 4: Estimación de esfuerzo por HU.

Historia de usuario	Número	Puntos Estimados
Gestionar usuarios del sistema.	1	2
Autenticar usuarios.	2	1
Mostrar determinación del límite líquido.	3	0.5
Mostrar determinación del límite plástico.	4	0.5
Mostrar resultados del ensayo.	5	1
Obtener gráfica de Humedad versus # de Golpes.	6	2
Mostrar determinación de hinchamiento controlado.	7	1
Mostrar determinación de humedad.	8	0.5
Mostrar propiedades iniciales y finales.	9	0.5
Mostrar resultado del ensayo.	10	1
Obtener gráfico de Presión Axial versus Logaritmo del tiempo.	11	2
Mostrar datos iniciales del ensayo.	12	0.5
Mostrar resultados del ensayo.	13	0.5
Mostrar ángulo de reposo en suelos arenosos.	14	1
Mostrar ángulo de reposo en suelos	15	1
Mostrar resultados del ensayo.	16	1

2.4.2 Plan de entregas.

El plan de entregas se realiza teniendo en cuenta las unidades funcionales que se quieren entregar y cada uno de estos módulos abarca un número de historias de usuarios a implementar para dar cumplimiento al funcionamiento del mismo.

Tabla 5: Plan de entregas.

Módulo	Historia (s) de usuarios que abarca
➤ Gestionar Usuarios.	1 y 2.
➤ Mostrar resultados obtenidos.	3,4, 5, 7, 8, 9, 10, 12, 13, 14, 15, 16.
➤ Obtención de las gráficas.	6 y 11.

2.4.3. Plan de duración de las Iteraciones.

Como parte del ciclo de vida de un proyecto guiado por la metodología de desarrollo de software XP, se crea el plan de duración de cada una de las iteraciones que se llevarán a cabo durante el desarrollo del mismo. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de las mismas.

Tabla 6: Plan de duración de las iteraciones.

Iteración	Historias de usuario	Duración total (semanas)
Iteración 1	<ul style="list-style-type: none"> • Gestionar Usuarios. • Autenticar usuarios. 	2+1= 3 semanas.

Iteración 2	<ul style="list-style-type: none"> • Mostrar determinación del límite líquido. • Mostrar determinación del límite plástico • Mostrar resultados del ensayo. • Mostrar determinación de hinchamiento controlado • Mostrar determinación de humedad. • Mostrar propiedades iniciales y finales. • Mostrar resultado del ensayo. • Mostrar datos iniciales del ensayo. • Mostar resultados del ensayo. • Mostrar ángulo de reposo en suelos arenosos. • Mostrar ángulo de reposo en suelos sumergidos. • Mostrar resultados del ensayo. 	$0.5+0.5+0.5+0.5+0.5+0.5+1+1+1+1+1+1=9$ semanas.
Iteración 3	<ul style="list-style-type: none"> • Obtener gráfico de Presión Axial versus Logaritmo del tiempo. • Obtener gráfica de Humedad versus # de Golpes. 	$2+2= 4$ semanas.

Combinando el plan de entrega y el plan de iteraciones se harán releases o liberaciones al sistema en las fechas mostradas a continuación:

Módulo #1 = Gestionar Usuarios.

Módulo #2 =Mostrar resultados obtenidos.

Módulo #3 = Obtención de las gráficas.

Tabla 7: Tabla de releases.

Iteración/Módulo	Módulo #1	Módulo #2	Módulo #3
Fin de la 1ra iteración	26/04/2012		
Fin de la 2da iteración		25/05/2012	
Fin de la 3ra iteración			12/06/2012

2.5-Tarjetas CRC.

En este epígrafe tiene lugar la realización de las tarjetas de clases, responsabilidades y colaboración, conocidas tradicionalmente como tarjetas CRC, las cuáles se realizan con el objetivo de facilitar la comunicación y documentar los resultados. Además, las mismas permiten la total participación y contribución del equipo de desarrollo en la tarea de diseño.

Una tarjeta CRC representa un objeto, por tanto es una clase, cuyo nombre se ubica en forma de título en la parte superior de la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda y las clases implicadas con cada responsabilidad a la derecha, en la misma línea de su requerimiento correspondiente. Esta nueva técnica de diseño es adoptada como alternativa a los diagramas UML de las clases, pues en estas se plasman las responsabilidades que tienen cada objeto y las clases con las que tienen que interactuar para darles respuesta brindando así la información que se necesita a la hora de implementar. (Vargas, 2011)

Tabla 8: Plantilla de tarjeta CRC

Clases	
Responsabilidades	Colaboraciones
Clase A	Clase B

Para una mejor comprensión de las Tarjetas C.R.C de nuestro sistema

procedemos agruparlas por módulos identificados en el [Plan de Entregas](#) en el Capítulo 2.

2.5.1 Tarjeta C.R.C Módulo #1: Gestionar Usuarios.

Tabla 9: Tarjeta CRC Autenticar usuarios.

Clases:Autenticar_users	
Responsabilidades	Colaboraciones
loguearse_Users	Conexxion.

Para consultar el resto de las tarjetas C.R.C ir: [Anexo-2](#)

2.6 Conclusiones del Capítulo.

En este capítulo de planeación y diseño se han expuesto las HU conjunto con el cliente a través de plantillas. Se realizó un plan de iteraciones para cada una de estas historias aplicando una estimación de esfuerzo de las mismas. Se presentaron además las principales clases que se utilizarán en el desarrollo de la aplicación a través de las llamadas tarjetas CRC, terminando de esta forma esta fase para pasar a la siguiente fase de implementación y pruebas.

Capítulo 3: Implementación y Pruebas.

Introducción:

En este capítulo se desarrollará la fase de implementación y pruebas conforme a lo que determina la metodología XP. Por otra parte se describen cada una de las tareas confeccionadas para cumplir con el desarrollo de cada una de las HU definidas. Además se mostrarán las pruebas de aceptación confeccionadas por el cliente para comprobar que la aplicación funcione correctamente. Estas pruebas fueron realizadas durante la entrega que se efectuaban a lo largo del desarrollo del proyecto.

3- Implementación.

En la metodología XP se convierte en un integrante más del equipo de desarrollo el cliente pues él crea las historias de usuario bajo la supervisión de los desarrolladores. Estas historias quedan confeccionadas cuando el cliente es capaz de identificar con precisión la funcionalidad deseada, además, también debe estar presente cuando se realicen las pruebas de aceptación para cada historia, por lo que su presencia es imprescindible.

En XP generalmente cada historia de usuario se divide en tareas de ingeniería (TI) o tareas de programación. Estas se crean para obtener una mejor planificación de la historia; con ellas se pretende cumplir con las funcionalidades básicas que luego conformarán las funcionalidades generales de cada historia. (Vargas ,2011)

Tabla10: Plantilla de Tareas de Ingeniería.

Tarea de Programación	
Número tarea:	Número historia:
Nombre tarea:	
Tipo de tarea:	Puntos estimados:
Fecha inicio:	Fecha fin:
Programador responsable:	
Descripción:	

A continuación se presentan las Tareas de Ingeniería agrupadas por las respectivas historias de usuario a las que pertenecen.

3.1 Tareas por HU

3.1.1 Tareas de la HU #1

Tabla 11: Tareas de Ingeniería.

Tarea de Ingeniería	
Número tarea: 1	Número historia: 1
Nombre tarea: Presentación de la vista de la inserción de nuevos usuarios al sistema.	
Tipo de tarea: Diseño	Puntos estimados: 2
Fecha inicio: 10/04/2011	Fecha fin: 12/04/2011
Programador responsable: Roilan Martínez Pérez.	
Descripción: Los datos que se suministran por el formulario son evaluados para ver el grado de veracidad de los datos; o sea se valida la correcta entrada de datos y si no hay campos vacíos.	

Para consultar el resto de las Tareas de Ingenierías ir: [Anexo-3](#)

3.2-Pruebas

3.2.1 Pruebas de Aceptación (PA).

Una de las mejores características de la metodología XP es el proceso de pruebas. Esta metodología propone probar tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos no deseados a la hora de realizar modificaciones y refactorizaciones. XP propone la realización de pruebas unitarias, encargadas de verificar el código y diseñadas por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida por el cliente. (Wesley, 2000)

En este proyecto al codificar no se sigue la regla de XP que aconseja crear pruebas unitarias con entornos de desarrollo antes de programar. Las que se

obtuvieron de la descripción de requisitos plasmados en las historias de usuarios, y estas especifican las barreras que deben pasar las distintas funcionalidades del programa, procurando codificar teniendo en cuenta las pruebas que se deben vencer. Para realizar las pruebas de aceptación el cliente utiliza la siguiente plantilla. (Wesley, 2000)

Tabla 12: Planilla de Prueba de Aceptación.

Prueba de Aceptación
HU: Nombre de la historia de usuario que va a comprobar su funcionamiento.
Nombre: Nombre del caso de prueba.
Descripción: Descripción del propósito de la prueba.
Condiciones de ejecución: Precondiciones para que la prueba se pueda realizar.
Entrada/Pasos de ejecución: Pasos para probar la funcionalidad.
Resultado esperado: Resultado que se desea de la prueba.
Evaluación de la prueba: Aceptada o Denegada.

A continuación se presentan las pruebas que se tuvieron en cuenta para verificar el buen funcionamiento de cada módulo en las entregas que se le hacen al cliente cumpliendo con lo establecido en el cronograma de entregas.

3.2.2 Pruebas del módulo #1: Gestionar Usuarios.

Tabla 13: P.A Gestionar usuarios del sistema.

Prueba de Aceptación
HU: Gestionar usuarios del sistema.
Nombre: Prueba para verificar la Gestión de usuarios.
Descripción: Validación de la Gestión de usuarios.
Condiciones de ejecución: El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador de esta, para poder insertar usuarios, modificarlos, eliminarlos, cambiarles su contraseña sin necesidad de pedir su contraseña actual, así como listar todos los usuarios con acceso al Sitio.
Entrada/Pasos de ejecución: El administrador escribe todos los datos que se le piden para poder insertar un usuario determinado asignándole el nivel que se desee, luego presiona el botón Enviar. Después que el usuario este insertado es que se le puede cambiar su contraseña, eliminarlo y ver sus datos en el listado de usuarios.

Resultado esperado:

- Si se insertó el usuario correctamente muestra un mensaje de confirmación con este mensaje " Usuario insertado correctamente".
- Cuando se modifica un usuario, se emite un mensaje de confirmación.

Evaluación de la prueba: Aceptada.

Para consultar el resto de las Pruebas de Aceptación ir: [Anexo-4](#)

3.3 Conclusiones del Capítulo.

En este capítulo se llevó a cabo la fase de implementación y pruebas. Se realizó el desarrollo de las iteraciones a partir de la distribución de tareas por historias de usuarios y se le practicó las pruebas de aceptación a las mismas para verificar que las funcionalidades de la aplicación estén correctamente implementadas, siendo todas estas aceptadas por el cliente de la aplicación, además de ver las principales interfaces de la aplicación, así como las tarjetas de ingenierías.

Capítulo 4: Estudio de Factibilidad del Proyecto.

Introducción:

La probabilidad de restricciones, insuficiencia de recursos y presiones en las fechas de entregas implícitas en el desarrollo en un producto, es real, por lo que es necesario y sensato evaluar la viabilidad del mismo. De identificarse en la fase de definición, posibles riesgos en la concepción del proyecto, se necesita minimizar de forma razonable recursos humanos, materiales y financieros. En general los productos informáticos no están excepto de estas dificultades, por lo que es de vital importancia estimar la relación costo-beneficio.

La cual plantea que la conveniencia de la ejecución de un proyecto se determina por la observación de dos factores en conjunto: (García Pérez, A)

- El costo que involucra la implementación de la solución informática, y puesta en marcha del sistema hardware/software y los costos de operación asociados. (García Pérez, A)
- La efectividad que se entiende como capacidad del proyecto para satisfacer la necesidad, solucionar el problema o lograr el objetivo por el cual se ideó, es decir, un proyecto será más o menos efectivo con relación al mayor o menor cumplimiento que alcance en la finalidad para la cual fue ideado (costo por unidad del cumplimiento del objetivo). (García Pérez, A)

4- Efectos Económicos.

- Efectos directos.
- Efectos indirectos.
- Efectos externos.
- Intangibles.

Efectos directos:

POSITIVOS:

- Se mejora la eficiencia del proceso de gestión de Informes de Ensayos en la asignatura Mecánica de suelos y rocas en la carrera de Ingeniería Geológica.

- Se mejorará el proceso de estudio de ensayos.
- Se mejorará el proceso de aprendizaje de los futuros profesionales.

NEGATIVOS:

- Para el uso de esta aplicación implementada con tecnología Web se necesitará que la misma sea ejecutada con el navegador de Firefox, porque se adapta a cualquier tipo de diseño.

Efectos indirectos:

- Los efectos económicos observados que pudiera repercutir sobre otros mercados no son perceptibles, aunque este proyecto no está construido con la finalidad de comercializarse.

Efectos externos:

- Se tendrá una herramienta disponible que le facilitará a los estudiantes y los que estén interesados en el mismo, para realizar estudio e interpretaciones de los ensayos calculados.

Intangibles:

- En la valoración económica siempre hay elementos perceptibles por una comunidad como perjuicio o beneficio, pero al momento de ponderar en unidades monetarias esto resulta difícil o prácticamente imposible. A fin de medir con precisión los efectos, deberán considerarse dos situaciones:
- **SITUACIÓN SIN EL PRODUCTO.**
 - Los especialistas, profesores y estudiantes vinculados a este campo, deben realizar el estudio e interpretación de los distintos ensayos en hojas Excel donde este proceso se les hace muy engorroso por:
 1. Muchos cálculos.
 2. Mucha información.
- **SITUACIÓN CON EL PRODUCTO.**
 - Cargar la aplicación en el navegador de la computadora.

- Entrar los datos correspondientes del usuario que hará uso de la aplicación.
- Mostrará los resultados obtenidos a la hora de realizar los distintos ensayos.

4.1 Beneficios y Costos Intangibles en el proyecto.

Costos:

- Resistencia al cambio, es decir que pueden variar el costo.

Beneficios:

- Mayor comodidad para los usuarios.
- Mejora en la calidad de acceso a la información.
- Facilidad a la hora de realizar estudios e interpretaciones de los distintos Ensayos.
- Mayor integración usuarios-artefactos.

4.2 Ficha de costo.

Para determinar el costo económico del proyecto se utilizará el procedimiento para elaborar una Ficha de Costo de un producto informático (Dra. Ana María Gracia Pérez, UCLV). Para la elaboración de la ficha se consideran los siguientes elementos de costo, desglosados en moneda libremente convertible y moneda nacional.

Costos en Moneda Libremente Convertible:

Tabla14: Costo en Moneda Libremente Convertible.

Ficha de Costo.	
	Precio(s)
Costo Moneda Libremente Convertible	
Costos Directos	
Compras de equipos de cómputo	0,00
Alquiler de equipos de cómputo	0,00
Compra de licencia de Software	0,00
Depreciación de equipos	0,53
Materiales directos	0,00
Subtotal	0,53
Costos indirectos	
Formación del personal que elabora el proyecto	0,00

Gastos en llamadas telefónicas	0,00
Gastos para el mantenimiento del centro	0,00
Know How	0,00
Gastos en representación	0,00
Subtotal	
Gastos de Distribución y Venta	
Participación en ferias o exposiciones	0,00
Gastos en transportación	0,00
Compra de materiales de propagandas	0,00
Subtotal	0,00
Total	0,53

Costos en Moneda Nacional:

Tabla 15: Costo en Moneda Nacional.

Ficha de Costo.	
	Precio(s)
Costo Moneda Nacional	
Costos Directos	
Salario del personal que laborará en el proyecto	400,0
Seguridad Social	0,00
Vacaciones $100 \times 9,09\% = 9.09$	0,00
Gastos en llamadas telefónicas	00,0
Impuesto por la Fuerza de Trabajo	0,00
Subtotal	400,0
Costos indirectos	
Know How	0,00
Gasto por consumo de energía eléctrica $68.64 \times 4 \text{ meses} \times 0.67.$	183,95
Subtotal	183,95
Total	583,95

Como se hizo referencia anteriormente, la técnica seleccionada para evaluar la factibilidad del proyecto es la Metodología Costo-Efectividad. Dentro de esta metodología, la técnica de punto de equilibrio aplicable a proyectos donde los

beneficios tangibles no son evidentes, el análisis se basa exclusivamente en los costos. Para esta técnica es imprescindible definir una variable discreta que haga variar los costos. Teniendo en cuenta que el costo para este proyecto es despreciable, tómesese como costo el tiempo en minutos empleado por los estudiantes, profesores y especialista en realizar el estudio e interpretación de los ensayos.

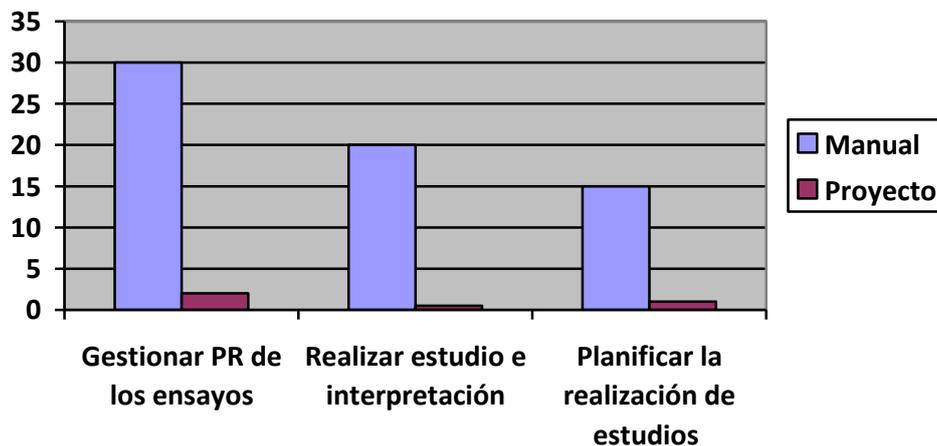
Valores de la variable (Solución manual):

- Gestionar el proceso de realización de los ensayos (30 min).
- Realizar el estudio e interpretación de los ensayos (20 min).
- Planificar la realización de estudios de los ensayos (15 min).

Valores de la variable (Solución con el software):

- Gestionar el proceso de realización de los ensayos (2 min).
- Realizar el estudio e interpretación de los ensayos (0.038 min).
- Planificar la realización de estudios de los ensayos (1 min).

Gráfica de la solución manual y solución con el producto.



Teniendo en cuenta los resultados reflejados en la gráfica queda demostrada la factibilidad del sistema, el tiempo que demora la solución del mismo de forma manual y con la aplicación.

4.3 Conclusiones del Capítulo.

Al terminar este capítulo se llegó a la conclusión que el producto obtenido es rentable, ya que al aplicar la metodología para calcular la factibilidad se obtuvo que el costo de desarrollo del producto software fue factible porque no se incurrieron en gastos excesivos para su desarrollo. En cuanto los beneficios obtenidos, tanto intangibles como tangibles, decir que: se mejora considerablemente el proceso de gestión de informes de ensayos. Y un factor importante todo esto tributa en el ahorro de recursos de oficinas y de tiempo.

Conclusiones Generales.

Llegada la culminación de esta investigación que nos llevó a implementar y desarrollar la aplicación propuesta, se arrojaron las siguientes conclusiones:

- ❖ Se desarrolló el producto final que consistía en la implementación de una aplicación informática con tecnología Web para la gestión de informes de ensayos en los laboratorios de Mecánica de suelos y rocas en la disciplina de Geología Aplicada en el ISMM.
- ❖ Se realizó la planificación y diseño de la aplicación, en los cuales se identificaron y especificaron los requerimientos funcionales, así como se llevaron a cabo las posteriores fases de implementación y prueba, que se definen en la metodología de desarrollo utilizada.
- ❖ Se llevó a cabo un estudio de las principales metodologías, lenguajes y herramientas que se consideraron factibles para el desarrollo del sistema.
- ❖ Se realizaron las pruebas de aceptación definidas por el cliente, lo que arrojó como resultado su aprobación, con lo cual se demostró el cumplimiento satisfactorio de las funcionalidades de la aplicación.

Como resultado de los objetivos propuestos en la investigación se logró la implementación del producto software que se esperaba. Con el propósito fundamental de simplificar las demoras que se producen en el tratamiento manual de la información, disminuir el grado de errores y para contribuir a elevar la calidad del desarrollo del trabajo.

Recomendaciones.

- Poner en explotación el software en todas las áreas del ISMM para aumentar la capacidad de respuestas en cuanto a la gestión de informes de ensayos.
- Realizar un estudio más profundo de este sistema en vista a perfeccionarlo en versiones futuras.
- Trabajar en la aplicación con navegadores Mozilla Firefox ya que traducen los estilos de diseño con mayor calidad.

Con la puesta en práctica de las recomendaciones que se expusieron anteriormente se logrará una capacidad de respuesta más eficiente a la hora del estudio de informes de ensayos. Además se incentiva para que se creen o mejoren las funcionalidades de la aplicación propuesta, para que conlleven a un mejor rendimiento de las mismas.

Referencias Bibliográficas.

Basulto, M. (2010). *"Sistema de gestión integral de la empresa empleadora del Níquel - módulo gestión de contratos de compras."*

Cala, A. (2011). *"Sistema de gestión energética asistida para el ISMMM- Módulo de control de consumo energético."*

Ferrer, J. (2007). *"Metodología ágil para el desarrollo de software."*

García, A.M. *"Procedimiento para la elaboración de la ficha de costo de un producto informático"*. Facultad MFC UCLV. Villa Clara.

JQuery.com. (2006). "Conceptos de JQuery." [en línea], 2011 Disponible en: <http://www.jquery.com>.

Letelier, P. y P., M. C. (2008). *"Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Valencia, Universidad Politécnica de Valencia."*

López, Y.M. (2011). *"Mapa Social de los Investigadores del Centro de Estudio de la Energía y la Tecnología Avanzada de Moa (CEETAM) usando algoritmos de clustering."*

Monografias.com. (2007). *"Definición arquitectura cliente servidor."* from <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>.

Rodríguez, M. y I. D. (2008). *"Herramienta para el control de los medios informáticos de la Empresa Cdte. "Ernesto Che Guevara."*

Rufino, A. (2007). *"Lenguaje de programación: PHP."* from Disponible en: <http://1sinfo.blogspot.com/2007/05/lenguaje-de-programacin-php.html>.

Ruiz, M. H. (2004). *"Programación Web avanzada."*

RUP. "Metodología de desarrollo software RUP." from Disponible en:

http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational

Subiros, D. (2009). "Desarrollo de una interfaz gráfica de usuario para el Preprocesador meteorológico AERMET."

Urgelles, A. (2011). "Sistema Informático para la Planificación y Control del Presupuesto del ISMMM."

Vargas, Y. (2011). "Herramienta para el control de reportes de averías y mantenimiento de los medios informáticos del ISMM."

Wesley, A. (2000). "Una explicación de la programación extrema. Aceptar el cambio."

XAMPP. (2008). "Paquete de tecnología Web Xampp ", from Disponible en:

<http://e.wikipedia.org/wiki/XAMPP>.

Zulueta, A. (2010). "Modulo para la Extracción, Preprocesamiento, Descripción y Almacenaje en formato XML, de la información recuperada por el Sistema Automatizado de Información Virtual del ISMMM."

Anexos

Anexo 1 Historias de Usuarios.

Tabla 16: HU Autenticar usuario.

Historia de Usuarios	
Número: 2	Usuario: Administrador.
Nombre: Autenticar usuario.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Roilan Martínez Pérez.	
Descripción: Los usuarios deben insertar de forma correctas los datos que le permitirán interactuar con el sistema.	
Observaciones: Se deberán insertar nombre de usuario y contraseña que sean válidos en el sistema.	

Tabla 17: HU Mostrar determinación del límite líquido.

Historia de Usuarios	
Número: 3	Usuario: Invitado
Nombre: Mostrar determinación del límite líquido.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar lo referente a lo relacionado con el límite líquido.	
Observaciones: Confirmado.	

Tabla 18: HU Mostrar determinación del límite plástico.

Historia de Usuarios	
Número: 4	Usuario: Invitado
Nombre: Mostrar determinación del límite plástico.	

Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar lo referente a lo relacionado con el límite plástico.	
Observaciones: Confirmado.	

Tabla 19: HU Mostrar resultados del ensayo.

Historia de Usuarios	
Número: 5	Usuario: Invitado
Nombre: Mostrar resultados del ensayo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo.	
Observaciones: Confirmado.	

Tabla 20: HU Obtener gráfica de Humedad versus # de Golpes.

Historia de Usuarios	
Número: 6	Usuario: Invitado
Nombre: Mostrar gráfica de Humedad versus # de Golpes.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario que esté interactuando con el sistema podrá ver la gráfica relacionada con el ensayo al seleccionar esa opción.	
Observaciones: Confirmado.	

Tabla 21: HU Mostrar determinación de hinchamiento controlado.

Historia de Usuarios	
Número: 7	Usuario: Invitado
Nombre: Mostrar determinación de hinchamiento controlado.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo.	
Observaciones: Confirmado.	

Tabla 22: HU Mostrar determinación de humedades.

Historia de Usuarios	
Número: 8	Usuario: Invitado
Nombre: Mostrar determinación de humedades.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos a la hora de determinar la humedad.	
Observaciones: Confirmado.	

Tabla 23: HU Mostrar propiedades iniciales y finales.

Historia de Usuarios	
Número: 9	Usuario: Invitado
Nombre: Mostrar propiedades iniciales y finales.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo.	

Observaciones: Confirmado.

Tabla 24: HU Mostrar resultados del ensayo.

Historia de Usuarios	
Número: 10	Usuario: Invitado
Nombre: Mostrar resultados del ensayo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo hinchamiento controlado.	
Observaciones: Confirmado.	

Tabla 25: HU Mostrar gráfico de Presión Axial versus Logaritmo del tiempo.

Historia de Usuarios	
Número: 11	Usuario: Invitado
Nombre: Mostrar gráfico de Presión Axial versus Logaritmo del tiempo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se selecciona la opción ver gráfica y se obtiene la petición.	
Observaciones: Confirmado.	

Tabla 26: HU Mostrar datos iniciales del ensayo.

Historia de Usuarios	
Número: 12	Usuario: Invitado
Nombre: Mostrar datos iniciales del ensayo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio

Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo.	
Observaciones: Confirmado.	

Tabla 27: HU Mostrar resultados del ensayo.

Historia de Usuarios	
Número: 13	Usuario: Invitado
Nombre: Mostrar datos iniciales del ensayo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar los resultados obtenidos en la realización del ensayo.	
Observaciones: Confirmado.	

Tabla 28: HU Mostrar ángulo de reposo en suelos arenosos.

Historia de Usuarios	
Número: 14	Usuario: Invitado
Nombre: Mostrar ángulo de reposo en suelos arenosos.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario podrá visualizar lo referente con la determinación del ángulo de reposo.	
Observaciones: Confirmado.	

Tabla 29: HU Mostrar ángulo de reposo en suelos sumergidos.

Historia de Usuarios	
Número: 15	Usuario: Invitado
Nombre: Mostrar ángulo de reposo en suelos sumergidos.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario que esté interactuando con el sistema podrá visualizar los datos obtenidos al calcular el ángulo de reposo en suelos sumergidos.	
Observaciones: Confirmado.	

Tabla 30: HU Mostrar resultados del ensayo.

Historia de Usuarios	
Número: 16	Usuario: Invitado
Nombre: Mostrar resultados del ensayo.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Roilan Martínez Pérez.	
Descripción: El usuario que esté trabajando con el sistema podrá ver los resultados obtenidos en el ensayo realizado.	
Observaciones: Confirmado.	

Anexo-2 Tarjetas CRC

Tarjeta C.R.C Módulo #1: Gestionar usuarios.

Tabla 31: Tarjeta C.R.C. Gestionar_usuarios.

Clase: Gestionar_Usuarios.	
Responsabilidades	Colaboraciones
Insertar_usuario modificar_usuario eliminar _ usuario listar_usuario	Conexxion.

Tarjeta C.R.C Módulo # 2: Mostrar resultados obtenidos.

Tabla 32: Tarjeta C.R.C. Mostrar_resultados.

Clases: Mostrar_resultados.	
Responsabilidades	Colaboraciones
Mostrar_límite_líquido. Mostrar_límite_plástico. Mostrar_resultados_Ensayo. Mostrar_hincha_controlado. Mostrar_humedad. Mostrar_propiedades. Mostrar_resultado_Ensayo. Mostrar_contraccion. Mostar_resultados_Ensayo. Mostrar_angulo_seco Mostrar_angulo_humedo Mostrar_resultados_Ensayo	Conexxion.

Tarjeta C.R.C Módulo #3: Obtención de las Gráficas.

Tabla 33: Tarjeta C.R.C. Gráfica_Humedad_versus_#_Golpes.

Clases: Obtención gráficas.	
Responsabilidades	Colaboraciones
Gráfica_Humedad_versus_#_Golpes	ImgData_Balls. Gradient. GRAD_HOR GRAD_MIDHOR GRAD_MIDVER
Grafica_Presión_Axial_versus_Logaritmo _ tiempo	Legend PlotMark RGB FieldArrow Text LanguageConv

Anexo-3 Tareas de Ingeniería.

Tabla 34: Tarea de Ingeniería Insertar usuarios.

Tarea de Ingeniería	
Número tarea: 2	Numero de historia: 1
Nombre tarea: Insertar usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 02/04/2012	Fecha fin: 02/04/2012
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se insertan los usuarios que van a interactuar con el sistema.	

Tabla 35: Tarea de Ingeniería Eliminar usuarios.

Tarea de Ingeniería	
Número tarea: 3	Numero de historia: 1
Nombre tarea: Eliminar usuarios.	

Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 03/04/2012	Fecha fin: 04/04/2012
Programador responsable: Roilan Martínez Pérez.	
Descripción: Elimina usuarios insertados en el sistema.	

Tabla 36: Tarea de Ingeniería Modificar usuarios.

Tarea de Ingeniería	
Número tarea: 4	Numero de historia: 1
Nombre tarea: Modificar usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 05/04/2012	Fecha fin: 06/04/2012
Programador responsable: Roilan Martínez Pérez.	
Descripción: Modifica algunos elementos de los usuarios.	

Tabla 37: Tarea de Ingeniería Visualizar usuarios.

Tarea de Ingeniería	
Número tarea: 5	Numero de historia: 1
Nombre tarea: Visualizar usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 11/04/2012	Fecha fin: 12/04/2012
Programador responsable: Roilan Martínez Pérez.	
Descripción: El administrador podrá ver toda la información sobre los usuarios que están registrados en el sistema.	

Tabla 38: Tarea de Ingeniería Loguear usuarios.

Tarea de Ingeniería	
Número tarea: 6	Numero de historia: 1
Nombre tarea: Loguear usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 1

Fecha inicio: 17/04/2012.	Fecha fin: 19/04/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Para la autenticación de los usuarios.	

Tabla 39: Tarea de Ingeniería Mostrar determinación límite líquido.

Tarea de Ingeniería	
Número tarea: 7	Numero de historia: 2
Nombre tarea: Mostrar determinación límite líquido.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha inicio: 13/04/2012.	Fecha fin: 14/04/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Muestra los resultados obtenidos en el ensayo realizado.	

Tabla 40: Tarea de Ingeniería Mostrar determinación límite plástico.

Tarea de Ingeniería	
Número tarea: 8	Numero de historia: 2
Nombre tarea: Mostrar determinación límite plástico.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha inicio: 02/05/2012.	Fecha fin: 03/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se va a mostrar los resultados obtenidos a la hora de calcular el límite plástico.	

Tabla 41: Tarea de Ingeniería Mostrar resultados del ensayo.

Tarea de Ingeniería	
Número tarea: 9	Numero de historia: 2
Nombre tarea: Mostrar resultados del ensayo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1

Fecha inicio: 10/05/2012.	Fecha fin: 10/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados finales del ensayo de determinación de Límites.	

Tabla 42: Tarea de Ingeniería Obtener gráfica de Humedad vs # de Golpes.

Tarea de Ingeniería	
Número tarea: 10	Numero de historia: 3
Nombre tarea: Obtener gráfica de Humedad versus # de Golpes.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 04/06/2012.	Fecha fin: 07/06/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará la gráfica del ensayo.	

Tabla 43: Tarea de Ingeniería Mostrar hinchamiento controlado.

Tarea de Ingeniería	
Número tarea: 11	Numero de historia: 2
Nombre tarea: Mostrar hinchamiento controlado.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 14/05/2012.	Fecha fin: 15/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados obtenidos.	

Tabla 44: Tarea de Ingeniería Mostrar determinación de humedad.

Tarea de Ingeniería	
Número tarea: 12	Numero de historia: 2
Nombre tarea: Mostrar determinación de humedad.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5

Fecha inicio: 16/05/2012.	Fecha fin: 16/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados obtenidos.	

Tabla 45: Tarea de Ingeniería Mostrar propiedades iniciales y finales.

Tarea de Ingeniería	
Número tarea: 13	Numero de historia: 2
Nombre tarea: Mostrar propiedades iniciales y finales.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha inicio: 17/05/2012.	Fecha fin: 18/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados obtenidos.	

Tabla 46: Tarea de Ingeniería Mostrar resultados del ensayo.

Tarea de Ingeniería	
Número tarea: 14	Numero de historia: 2
Nombre tarea: Mostrar resultados del ensayo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 24/05/2012.	Fecha fin: 25/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados obtenidos en el ensayo Hinchamiento Controlado.	

Tabla 47: TI Obtener gráfico de Presión Axial/Logaritmo del tiempo.

Tarea de Ingeniería	
Número tarea: 15	Numero de historia: 3
Nombre tarea: Obtener gráfico de Presión Axial / Logaritmo del tiempo.	
Tipo de tarea: Desarrollo	Puntos estimados: 2

Fecha inicio: 28/05/2012.	Fecha fin: 30/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará la gráfica del ensayo.	

Tabla 48: Tarea de Ingeniería Mostrar datos iniciales del ensayo.

Tarea de Ingeniería	
Número tarea: 16	Numero de historia: 2
Nombre tarea: Mostrar datos iniciales del ensayo.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha inicio: 24/03/2012.	Fecha fin: 24/03/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los datos obtenidos que se obtuvieron en el ensayo.	

Tabla 49: Tarea de Ingeniería Mostar resultados del ensayo.

Tarea de Ingeniería	
Número tarea: 17	Numero de historia: 2
Nombre tarea: Mostar resultados del ensayo.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha inicio: 21/05/2012.	Fecha fin: 21/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados que se obtuvieron a la hora de realizar el ensayo Límites de contracción.	

Tabla 50: Tarea de Ingeniería Mostar ángulo de reposo en suelos arenosos.

Tarea de Ingeniería	
Número tarea: 18	Numero de historia: 2
Nombre tarea: Mostar ángulo de reposo en suelos arenosos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1

Fecha inicio: 22/05/2012.	Fecha fin: 22/05/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados que se obtuvieron.	

Tabla 51: Tarea de Ingeniería Mostrar ángulo de reposo en suelos sumergidos.

Tarea de Ingeniería	
Número tarea: 19	Numero de historia:2
Nombre tarea: Mostrar ángulo de reposo en suelos sumergidos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 12/04/2012.	Fecha fin: 12/04/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados que se obtuvieron.	

Tabla 52: Tarea de Ingeniería Mostrar resultados del ensayo.

Tarea de Ingeniería	
Número tarea: 20	Numero de historia:2
Nombre tarea: Mostrar resultados del ensayo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 15/04/2012.	Fecha fin: 16/04/2012.
Programador responsable: Roilan Martínez Pérez.	
Descripción: Se mostrará los resultados que se obtuvieron en el ensayo calculado Angulo de reposo.	

Anexo-4 Pruebas de Aceptación.

Tabla 53: Prueba para comprobar la Autenticación de los usuarios.

Prueba de aceptación
HU: Autenticar usuarios
Nombre: Prueba para comprobar la Autenticación de usuarios. (Nombre de usuario y contraseña).
Descripción: Validación de entrada de los datos de los usuarios.
Condiciones de ejecución: El usuario debe introducir su nombre de usuario y Contraseña.
Entrada/Pasos de ejecución: El usuario escribe su nombre de usuario y Contraseña y luego dar clic en el botón Entrar.
<p>Resultado esperado:</p> <ul style="list-style-type: none"> • Si el usuario tiene acceso para entrar a la aplicación e inserta sus datos correctamente entrará sin problemas al Sistema. <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> • Se inserte los datos de un usuario no válido para el Sistema o incorrectos. (Ya sea su nombre de usuario o su contraseña). • Se da clic en el botón Entrar sin insertar nada en los campos de texto. (Presenta un mensaje que debe insertar los datos correctamente)
Evaluación de la prueba: Aceptada.

Tabla 54: Prueba de aceptación No.3 Visualización del límite de contracción.

Prueba de aceptación
HU: Mostrar datos iniciales y finales del ensayo Límite de Contracción.
Nombre: Mostrar datos iniciales y finales del ensayo Límite de Contracción.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Límite de Contracción.
Entrada/Pasos de ejecución: el usuario selecciona la opción límite de contracción.
Resultado esperado:

Es visualizado correctamente lo seleccionado por el usuario.

Evaluación de la prueba: Satisfactoria.

Tabla 55: Prueba de aceptación No.4 Visualizar los resultados del ensayo.

Prueba de aceptación
HU: Mostrar resultados del ensayo Límite de Contracción.
Nombre: Mostrar resultados del ensayo Límite de Contracción.
Descripción: Se muestra los resultados obtenidos en el ensayo.
Condiciones de ejecución: El usuario debe seleccionar la opción ver resultados del ensayo.
Entrada/Pasos de ejecución: el usuario selecciona la opción ver resultados.
Resultado esperado: Es visualizado correctamente los resultados obtenidos.
Evaluación de la prueba: Satisfactoria.

Tabla 56: Prueba de aceptación No.5 Visualizar determinación límite líquido.

Prueba de aceptación
HU: Mostrar determinación de límite líquido.
Nombre: Mostrar determinación del límite líquido.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Límite líquido.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo límite líquido.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.
Evaluación de la prueba: Satisfactoria.

Tabla 57: Prueba de aceptación No.6 Visualizar determinación límite plástico.

Prueba de aceptación
HU: Mostrar determinación de límite plástico.
Nombre: Mostrar determinación del límite plástico.

Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Límite plástico.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo límite plástico.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.
Evaluación de la prueba: Satisfactoria.

Tabla 58: Prueba de aceptación No.7 Visualizar resultados del ensayo.

Prueba de aceptación
HU: Mostrar resultados del ensayo determinación de Límites.
Nombre: Mostrar resultados del ensayo determinación de Límites.
Descripción: Se muestra los resultados obtenidos en el ensayo.
Condiciones de ejecución: El usuario debe seleccionar la opción ver resultados del ensayo.
Entrada/Pasos de ejecución: el usuario selecciona la opción ver resultados.
Resultado esperado: Es visualizado correctamente los resultados obtenidos.
Evaluación de la prueba: Satisfactoria.

Tabla 59: Prueba de aceptación No.8 Visualizar Ángulo de reposo Seco.

Prueba de aceptación
HU: Mostrar Ángulo de reposo en suelos arenosos.
Nombre: Mostrar Ángulo de reposo en suelos arenosos.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Ángulo de reposo seco.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo Ángulo de reposo.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.

Evaluación de la prueba: Satisfactoria.

Tabla 60: Prueba de aceptación No.9 Visualizar Ángulo de reposo sumergido.

Prueba de aceptación
HU: Mostrar Ángulo de reposo en suelos sumergidos.
Nombre: Mostrar Ángulo de reposo en suelos sumergidos.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Ángulo de reposo sumergido.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo Ángulo de reposo.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.
Evaluación de la prueba: Satisfactoria.

Tabla 61: Prueba de aceptación No.10 Visualizar resultados del ensayo.

Prueba de aceptación
HU: Mostrar resultados del ensayo Ángulo de reposo.
Nombre: Mostrar resultados del ensayo Ángulo de reposo.
Descripción: Se muestra los resultados obtenidos en el ensayo.
Condiciones de ejecución: El usuario debe seleccionar la opción ver resultados del ensayo.
Entrada/Pasos de ejecución: el usuario selecciona la opción ver resultados.
Resultado esperado: Es visualizado correctamente los resultados obtenidos.
Evaluación de la prueba: Satisfactoria.

Tabla 62: Prueba de aceptación No.11 Visualizar Hinchamiento Controlado.

Prueba de aceptación
HU: Mostrar determinación de Hinchamiento Controlado.
Nombre: Mostrar determinación de Hinchamiento Controlado.

Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo Hinchamiento Controlado.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo Hinchamiento Controlado.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.
Evaluación de la prueba: Satisfactoria.

Tabla 63: Prueba de aceptación No.12 Visualizar Determinación humedad.

Prueba de aceptación
HU: Mostrar determinación de humedad.
Nombre: Mostrar determinación de humedad.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo humedad.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo Hinchamiento Controlado.
Resultado esperado: Es visualizado correctamente lo seleccionado por el usuario.
Evaluación de la prueba: Satisfactoria.

Tabla 64: Prueba de aceptación No.13 Visualizar Propiedad Inicial y Final.

Prueba de aceptación
HU: Mostrar propiedades iniciales y finales.
Nombre: Mostrar propiedades iniciales y finales.
Descripción: Se muestra lo seleccionado por el usuario.
Condiciones de ejecución: El usuario debe seleccionar la opción del ensayo propiedades iniciales y finales.
Entrada/Pasos de ejecución: dar clic en la opción que tiene el ensayo Hinchamiento Controlado.
Resultado esperado:

Es visualizado correctamente lo seleccionado por el usuario.

Evaluación de la prueba: Satisfactoria.

Tabla 65: Prueba de aceptación No.14 Visualizar resultados del ensayo.

Prueba de aceptación
HU: Mostrar resultados del ensayo Hinchamiento Controlado.
Nombre: Mostrar resultados del ensayo Hinchamiento Controlado.
Descripción: Se muestra los resultados obtenidos en el ensayo.
Condiciones de ejecución: El usuario debe seleccionar la opción ver resultados del ensayo.
Entrada/Pasos de ejecución: el usuario selecciona la opción ver resultados.
Resultado esperado: Es visualizado correctamente los resultados obtenidos.
Evaluación de la prueba: Satisfactoria.

Tabla 66: Prueba de aceptación No.15 Visualizar Gráfico obtenido.

Prueba de aceptación
HU: Obtener gráfica de Humedad versus # de Golpes.
Nombre: Obtener gráfica de Humedad versus # de Golpes.
Descripción: Se crea la gráfica según los datos que estén almacenados en la base de datos.
Condiciones de ejecución: deben de existir en la BD los valores que deben conformar la gráfica.
Entrada/Pasos de ejecución: el interesado va a ver en la pantalla la opción de graficar y al hacer clic sobre esta se mostrará la gráfica que arroja la aplicación.
Resultado esperado: Se muestra la gráfica según los datos que están en la BD.
Evaluación de la prueba: Satisfactoria.

Tabla 67: Prueba de aceptación No.16 Visualizar Gráfico obtenido.

Prueba de aceptación
HU: Obtener gráfico de Presión Axial versus Logaritmo del tiempo.
Nombre: Obtener gráfico de Presión Axial versus Logaritmo del tiempo.
Descripción: Se crea la gráfica según los datos que estén almacenados en la base de datos.
Condiciones de ejecución: deben de existir en la BD los valores que deben conformar la gráfica.
Entrada/Pasos de ejecución: el interesado va a ver en la pantalla la opción de graficar y al hacer clic sobre esta se mostrará la gráfica que arroja la aplicación.
Resultado esperado: Se muestra la gráfica según los datos que están en la BD.
Evaluación de la prueba: Satisfactoria.