

Trabajo de Diploma

Para Optar por el Título de

Ingeniero Informático

**Título: Sistema de Gestión de Información sobre
Derrames de Hidrocarburos en el puerto de Moa**

Autor: Aurelio Alejandro Calvo Burke

Tutor: MSc. Roiky Rodríguez Noa

Consultantes: Dr. Yuri Almaguer Carmenates

MSc. Yosbaní Cervantes Guerra

Moa, 2013

“Año 55 de la Revolución



Declaración de Autoría

Declaro que soy el único autor de este trabajo y autorizo al Instituto Superior Minero Metalúrgico para que haga el uso que estime conveniente con este trabajo.

Para que así conste firmo la presente a los _____ días del mes de _____ del 2013.

Aurelio Alejandro Calvo Burke

Firma del Autor

MSc. Roiky Rodríguez Noa

Firma del Tutor



“De aquel que opina que el dinero puede hacerlo todo, cabe sospechar con fundamento que será capaz de hacer cualquier cosa por dinero.”

Benjamín Franklin



Les dedico este trabajo a mis padres porque sin ellos no hubiera podido llegar hasta aquí, por haberme apoyado siempre, por haberme dado la confianza necesaria para continuar adelante.



Quiero agradecer a todas las personas que de una forma u otra me han servido de apoyo para el desarrollo de este trabajo:

A mis padres Alicia Burke Blake y Aurelio Calvo Pérez que son las personas más importantes en mi vida.

A mi familia pues el apoyo de ellos ha sido muy importante para mí, en especial a mis tíos Humberto y Rezi por ayudarnos a mi madre y a mí en el momento más difícil cuando mi viejo estuvo ingresado en el hospital.

A mis compañeros del CEMA por haberme apoyado en el desarrollo de esta investigación.

A mis amigos de la universidad y de toda una vida por ayudarme en los momentos más importantes, por haberme dado su confianza. A mis compañeros de aula, un grupo con características muy particulares donde puedes encontrar un gato, varios cuervos, personas a las que les digan Body y Face, a viciosos jugadores de Dota o fanáticos del Real Madrid y a Cristiano Ronaldo, a la chica más guapa del aula a la cual cariñosamente le digo la loquita, en fin agradecerles a todos ellos por haberme aguantado durante 5 largos años.



El mundo actual gira y se sustenta sobre el constante flujo de información. Las necesidades de acelerar este flujo impulsan a un mayor uso de las Tecnologías de la Información y las Comunicaciones (TIC) dada las ventajas que estas proporcionan. La Empresa Puerto Moa Comandante Raúl Díaz Arguelles no está ajena de este proceso evolutivo de las ciencias informáticas. Un proceso tan importante como el de gestión de información de derrames de hidrocarburos en la bahía del puerto se maneja de forma deficiente de manera que no se garantiza la disponibilidad y fiabilidad de la información generada, lo que propicia la necesidad de una mayor informatización del actual sistema de gestión de información sobre derrames de hidrocarburos en el puerto de Moa. En la actual investigación se presenta una aplicación web que será capaz de lograr la disponibilidad y fiabilidad de información referente a los derrames de hidrocarburos, que visualice en un mapa del puerto de Moa la ubicación de los derrames y muestre mediante gráficas la cantidad de derrames ocurridos y el volumen de hidrocarburo derramado por año.



The world turns and is based on the constant flow of information. The flow needs to accelerate this push for greater use of Information Technology and Communications (ICT) given the benefits that they provide. Puerto Moa Company Commander Raul Diaz Arguelles is not immune from this evolution of computer science. A process as important as the information management of oil spills in the bay of Port handled so poorly not guarantee the availability and reliability of the information generated, which promotes the need for greater computerization of the current information management system for oil spills at the port of Moa. In the present investigation we present a web application that will be able to achieve the availability and reliability of information about oil spills, what you see on a map the location Moa port of spills and graphic display by the number of oil spills and the volume of oil spilled every year.



Índice

Introducción	1
Capítulo 1.....	5
1.1 Introducción	5
1.2 Sistemas existentes vinculados al campo de acción	5
1.3 Aplicaciones Web. Ventajas y Desventajas	5
1.4 Lenguajes y Tecnologías utilizados	7
1.4.1 Servidores para aplicaciones Web	7
1.4.2 Lenguaje de programación PHP (del lado del servidor).....	8
1.4.3 Lenguajes de programación JavaScript (del lado del navegador).....	9
1.4.4 Lenguaje de programación HTML (del lado del navegador).	11
1.4.5 Framework de desarrollo Web	11
1.4.6 Estilo Arquitectónico.....	13
1.5 Sistema Gestor de Base de Datos.....	15
1.5.1 PostgreSQL (versión 8.0).....	15
1.6 Metodologías de Desarrollo	16
1.6.1 Metodología XP	16
1.7 Herramientas utilizadas	21
1.7.1 Sistema de Información Geográfica (SIG).....	21
1.7.2 Servidor de Mapas Geoserver (versión 2.0.2)	21
1.7.3 Paquete Tecnológico XAMPP v1.7.3	21
1.7.4 Visual Paradigm (versión 5.3)	21
1.7.5 Embarcadero ER/Studio (versión 8.0).....	22
Conclusiones parciales.....	22
Capítulo 2: Planificación y Diseño	24
2.1 Introducción	24
2.2 Lista de reserva.....	24
2.3 Historias de Usuarios	27
2.4 Planificación de entregas.....	28
2.5 Planificación de Iteraciones.....	29
2.6 Plan de duración de iteraciones	30
2.7 Valoración de Riesgo	31



2.8 Tarjetas Clases – Responsabilidades - Colaboración (CRC).....	32
2.9 Modelo de Datos.....	33
Conclusiones parciales.....	34
Capítulo 3 Desarrollo y Prueba	35
3.1 Introducción	35
3.2 Desarrollo de las iteraciones.....	35
3.2.1 Tareas de ingeniería por HU	35
3.3 Prueba	38
3.3.1 Prueba de aceptación.....	38
Conclusiones parciales.....	39
Capítulo 4: Estudio de Factibilidad	40
4.1 Introducción	40
4.2 Evaluación Costo-Beneficio	40
4.3 Efectos Económicos.....	41
4.4 Elementos para identificar los Costos y Beneficios del Proyecto	42
4.5 Factibilidad Económica	43
4.6 Evaluación Económica.....	43
4.6.1 Beneficios Tangibles	45
4.6.2 Beneficios y costos intangibles del proyecto.....	46
Conclusiones Parciales	47
Conclusiones Generales	48
Recomendaciones.....	49
Referencias Bibliográficas.....	50
Bibliografía.....	52
Glosario de Términos.....	53
Anexos	55
Anexos 1 Historia de Usuarios	55
Anexos 2 Tarjetas CRC.....	60
Anexos 3 Tareas de Ingeniería	65
Anexo 4 Pruebas de Aceptación.....	76



Índice de Tablas

Tabla 2.1: Personal relacionado con el sistema.....	24
Tabla 2.2 Lista de reserva del producto	25
Tabla 2.4 Estimación de esfuerzo por HU.....	28
Tabla 2.5 Iteraciones	31
Tabla 2.7 Valoración de riesgos.....	32
Tabla 2.6 Tarjeta CRC de Gestionar Entidad Responsable	32
Tabla 3.1 Distribución de tareas por cada historia de usuarios	35
Tabla 3.2 Historias de Usuarios incluidas en la primera iteración	37
Tabla 3.3 Historias de Usuarios incluidas en la segunda iteración	37
Tabla 3.4 Historias de Usuarios incluidas en la tercera iteración	38
Tabla 3.5 Prueba de Aceptación para la HU: Autenticar Usuario.....	38
Tabla 4.1 Costos en moneda libremente convertible	43
HU No.1 Gestionar Entidad Responsable	55
HU No.2 Gestionar Derrame	55
HU No.3 Gestionar Reporte de Derrame.....	56
HU No.4 Gestionar Reporte de Incidente.....	56
HU No.5 Gestionar Informe Final.....	57
HU No.6 Gestionar Cuentas de Usuario.....	57
HU No.7 Autenticar Usuario.....	58
HU No.8 Mostrar derrame de hidrocarburo en un mapa.....	58
HU No.9 Graficar cantidad de derrames ocurridos por año	59
HU No.10 Graficar volumen de hidrocarburo derramado en un año.	59
Tarjeta CRC No.1 Entidad Responsable	60
Tarjeta CRC No.2 Derrame	60
Tarjeta CRC No.3 Ocurrió.....	61
Tarjeta CRC No.4 Reporte de Derrame.....	61
Tarjeta CRC No.5 Reporte de Incidente.....	62
Tarjeta CRC No.6 Informe Final.....	63
Tarjeta CRC No.7 Usuario.	63
Tarjeta CRC No.8 Artefacto Naval.....	64
Tarjeta CRC No.9 UEB.....	64
Tarjeta CRC No.10 Clasificación del derrame	65
Tarea No.1 Insertar Entidad Responsable.	65



Tarea No.2 Modificar Entidad Responsable.	65
Tarea No 3 Mostar Entidad Responsable	66
Tarea No 4 Insertar Derrame de Hidrocarburo.....	66
Tarea No 5 Modificar Derrame de Hidrocarburo	67
Tarea No 6 Mostrar Derrame de Hidrocarburo.....	67
Tarea No 7 Eliminar Derrame de Hidrocarburo.....	67
Tarea No 8 Insertar Reporte de Derrame.....	68
Tarea No 9 Modificar Reporte de Derrame	68
Tarea No 10 Mostrar Reportes de Derrame.....	69
Tarea No 11 Eliminar Reporte de Derrame.....	69
Tarea No 12 Insertar Reporte de Incidente.....	69
Tarea No 13 Modificar Reporte de Incidente.	70
Tarea No 14 Mostrar Reporte de Incidente.....	70
Tarea No 15 Eliminar Reporte de Incidente.....	71
Tarea No 16 Insertar Informe Final.	71
Tarea No 17 Modificar Informe Final.....	71
Tarea No 18 Mostrar Informe Final.	72
Tarea No 19 Eliminar Informe Final.....	72
Tarea No 20 Insertar Usuario	72
Tarea No 21 Modificar Usuario	73
Tarea No 22 Mostrar Usuario	73
Tarea No 23 Eliminar Usuario	74
Tarea No 24 Autenticar Usuario.....	74
Tarea No.25 Mostrar Derrame en un mapa.....	74
Tarea No.26 Graficar cantidad de derrames de hidrocarburos ocurridos en un año. ...	75
Tarea No.27 Graficar cantidad de derrames de hidrocarburos ocurridos en un año. ...	75
Prueba de Aceptación para la HU: Gestionar Cuenta de Usuario.	76
Prueba de Aceptación para la HU: Gestionar Derrame de Hidrocarburo.	77
Prueba de Aceptación para la HU: Gestionar Reporte de Derrame.	78
Prueba de Aceptación para la HU: Gestionar Reporte de Incidente.....	79
Prueba de Aceptación para la HU: Gestionar Informe Final.	80
Prueba de Aceptación para la HU: Mostrar derrames de hidrocarburo en un mapa. ...	81



Introducción

En la actualidad los hidrocarburos son la principal fuente de energía utilizada. La explotación y posterior transporte de los hidrocarburos constituyen una de las principales causas de contaminación del medio ambiente. Debido a los grandes volúmenes que se manejan en la industria petrolera los derrames tanto en tierra como en el agua son inevitables. Al ocurrir un derrame se genera un volumen de información muy importante ya que a partir de ésta es que las organizaciones implicadas toman las medidas necesarias para controlar el incidente, por lo tanto, el cómo se gestiona esa información se convierte en un proceso de vital importancia ya que incide en la toma de decisiones.

La necesidad de que el flujo de la información generada por el derrame se acelere impulsa a un mayor uso de las TIC dadas las ventajas que estas proporcionan, como la digitalización de los sistemas de gestión de información. La digitalización proporciona una mayor gestión y control efectivo de la información por parte de la empresa, facilita que la información se comparta y se aproveche de forma más eficiente, genera una productividad y valor añadido adicionales originados por el rápido acceso a la información dentro de la empresa y su posterior distribución, sin necesidad de trasladar los documentos.

Un proceso tan importante como el de gestión de información de derrames de hidrocarburos, tanto en tierra como en la bahía, de La Empresa Puerto Moa Comandante Raúl Díaz Arguelles se maneja de forma deficiente de manera que no se garantiza la disponibilidad y fiabilidad de la información generada, lo que propicia la necesidad de una mayor informatización del actual sistema de gestión de información sobre derrames de hidrocarburos en el puerto de Moa. El especialista del medio ambiente se encarga de redactar el reporte de incidente ambiental de forma manual, el cual presenta a los dirigentes del Puerto Moa y luego es enviado al Centro de Investigación y Tecnología del Medio Ambiente (CITMA) provincial. Dicho reporte se redacta cumpliendo con parámetros establecidos por la propia empresa, pero la información expuesta no cumple con los requisitos necesarios para un Sistema Informativo OSDE Derrame de Hidrocarburo exigido por el Ministerio de Energía y Minas.



Esta **situación problemática** permite plantearse el siguiente **problema científico**: ¿Cómo hacer más eficiente la gestión de información sobre derrames de hidrocarburos en el Puerto Moa Comandante Raúl Díaz Arguelles?

Se plantea como **objeto de estudio**: Las TIC para el desarrollo de Sistemas de gestión de información. Este objeto de estudio nos delimita como **campo de acción**: Sistema de gestión de información sobre derrame de hidrocarburos en el Puerto Moa Comandante Raúl Díaz Arguelles.

Se plantea como **Objetivo General**:

Desarrollar una Aplicación Web que mejore la eficiencia del actual sistema de gestión de información para documentar los derrames de hidrocarburos en el puerto de Moa de forma que favorezca la disponibilidad y fiabilidad de los reportes que establecen las normas para el registro de sucesos de este tipo.

Para dar cumplimiento al objetivo general se trazan los siguientes **objetivos específicos**:

1. Establecer el estado del arte acerca de los sistemas de gestión de información en el campo de los derrames de hidrocarburo.
2. Seleccionar las herramientas y la metodología a utilizar en el proceso de desarrollo de la aplicación Web.
3. Documentar el proceso de desarrollo del software.
4. Realizar el estudio de factibilidad que permita mostrar los costos y beneficios del sistema.

Para dar cumplimiento a los Objetivos Específicos se definieron las siguientes **tareas de la investigación**:

1. Estudio de los diferentes Sistemas de Gestión de Información sobre derrames de hidrocarburo.
2. Estudiar las tecnologías seleccionadas para el desarrollo de la aplicación.



3. Diseñar el sistema para garantizar que el producto satisfaga las necesidades del cliente.
4. Estudiar detalladamente el proceso de gestión de información sobre derrames de hidrocarburos en el Puerto Moa Comandante Raúl Díaz Arguelles.
5. Identificar los requerimientos.
6. Diseñar la base de datos.
7. Implementar el sistema de gestión de información sobre derrames de hidrocarburos.
8. Realizar pruebas funcionales.
9. Realizar Manual de Usuario.

Se defiende la **Idea** de que con el desarrollo de una aplicación web para la gestión de información sobre derrames de hidrocarburos en el Puerto Moa Comandante Raúl Díaz Arguelles se favorecerá la disponibilidad y fiabilidad de los reportes e informes, y se garantizará que éstos cumplan con las normas establecidas por el Ministerio de Energía y Minas.

Los **métodos empíricos** utilizados son:

1. **Entrevista:** para conocer en detalles cómo era el trabajo del especialista del Medio Ambiente en el Puerto Moa y además para determinar los requerimientos que debe cumplir el sistema a desarrollar.
2. **Análisis de documentos y materiales:** mediante ésta se supo cuál era la funcionalidad de los mecanismos actuales del proceso de gestión de información en la Empresa Puerto Moa.

Los **métodos teóricos** utilizados son:

1. **Análisis y síntesis** para la recopilación y el procesamiento de la información obtenida en los métodos empíricos y arribar a las conclusiones de la investigación.



2. **Histórico y lógico** se puso en práctica durante el estudio de las herramientas a utilizar.

El presente trabajo de diploma está estructurado en 4 capítulos:

Capítulo 1: “Fundamentación Teórica”: Este capítulo incluye un estado del arte del tema tratado, a nivel internacional y nacional, además se analizarán los diferentes lenguajes, tecnologías y metodologías de desarrollo de software empleadas en el desarrollo de la aplicación propuesta.

Capítulo 2: “Planeación y Diseño”: Se hace uso de la metodología expuesta en el capítulo anterior para el desarrollo del proyecto, abordando en detalles cada una de sus fases.

Capítulo 3: “Desarrollo y Pruebas”: Se presentan los principales métodos y definiciones dentro de la implementación de los flujos de trabajo. Se describen las pruebas realizadas y sus resultados.

Capítulo 4: “Estudio de Factibilidad”: Se hace referencia a los beneficios tangibles e intangibles y se analizan los costos de desarrollo de la aplicación contra los beneficios para decidir si es factible o no la aplicación y se valora la sostenibilidad del producto.



Capítulo 1

1.1 Introducción

En el presente capítulo se hará referencia a algunos de los sistemas de gestión de información vinculados al campo de acción. También se abordarán conceptos fundamentales sobre el desarrollo de aplicaciones web, las características generales de los sistemas de bases de datos, de los lenguajes de programación Web, se hará referencia a las herramientas a utilizar, también se analizarán las diferentes metodologías existentes para el desarrollo de aplicaciones web, y se seleccionará la más conveniente a utilizar.

1.2 Sistemas existentes vinculados al campo de acción

El manejo y gestión de la información es de suma importancia en caso de producirse un derrame de petróleo, existen sistemas que permiten gestionar este tipo de información como el **Oíl Map**, este sistema permite modelar un derrame, predecir las posibles áreas afectadas por el hidrocarburo derramado, proporciona el recorrido marcha atrás con la finalidad de averiguar dónde inició el derrame, permite valorar el impacto del derrame. También está el **Oil Map Deep**, este sistema permite una rápida predicción del movimiento del derrame en aguas, permite estimar el tamaño de las partículas de petróleo y hereda todas las funcionalidades del **Oíl Map**. Ambos software no se pueden utilizar en Cuba debido a que utilizan información almacenada en servidores a los cuales no podemos acceder. En el ámbito nacional no se encontró ninguna investigación similar. [1]

1.3 Aplicaciones Web. Ventajas y Desventajas

Con la llegada de la web 2.0, propiciada ésta por el desarrollo de las Tecnologías de la Información y las Comunicaciones y los nuevos retos de Internet, se abre paso una nueva idea en la concepción del software orientado a servicios o SaaS (acorde a sus siglas en inglés). SaaS es un modelo de distribución del software que proporciona a los clientes el acceso al mismo a través de la red (generalmente Internet), de manera que les libra del mantenimiento de las aplicaciones, de operaciones técnicas y de soporte. Las aplicaciones distribuidas en la modalidad SaaS pueden llegar a cualquier tipo



de empresa sin importar su tamaño o su ubicación geográfica. Se trata de un modelo que une el producto (software) al servicio, para dotar a las empresas de una solución completa que permita optimizar sus costes y sus recursos.

Ventajas:

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 proactivas, 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 se está trabajando dentro de grandes organizaciones).

Inmediatez de acceso: Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Se accede a la 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.

Menos Bugs: Las aplicaciones basadas en Web deberían ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos.



Precio: Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables y no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios Premium como una opción.

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: A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, 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 brindan amplios servicios de resguardo de datos ya sea como una parte integral del servicio básico o como una opción paga.

Desarrollar aplicaciones en el lenguaje que usted quiera: Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos, pueden también ser escritas en prácticamente cualquier lenguaje de programación. [2]

1.4 Lenguajes y Tecnologías utilizados

1.4.1 Servidores para aplicaciones Web

Un servidor web es un programa que se ejecuta continuamente en una computadora manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar a estas peticiones de una forma adecuada entregando como



resultado una página web o información de todo tipo de acuerdo a los comandos solicitados. [3]

1.4.1.1 Apache (versión 2.2.2)

Apache es un software (libre) servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que originalmente Apache consistía solamente en un conjunto de parches a aplicar al servidor NCSA. Era, en inglés, a patchy server (un servidor parcheado).

El servidor Apache se desarrolla dentro del proyecto HTTP Server (HTTP) de la Apache Software Foundation. Presenta, entre otras características, mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración. Apache tiene amplia aceptación en la red, es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft). [4]

Principales características: [5]

- Multiplataforma
- Gestión avanzada de indexación y alias de directorios
- Sistema de notificación de errores *HTTP* altamente configurable
- Soporte para *HTTP*
- Configuración simple y potente a través de ficheros
- Soporte avanzado de programas *CGI* (*Common Gateway Interface*)
- Soporte para autenticación *HTTP*
- Servidor *proxy* integrado

1.4.2 Lenguaje de programación PHP (del lado del servidor)

Es el lenguaje líder del mundo Open Source y uno de los más potentes, estables y seguros. Ofrece un gran número de herramientas y funciones que



hasta el programador menos experimentado no tendrá problemas en comprender. Hoy en día son muchos los sitios que han decidido migrar a PHP dado su excelente performance en conjunto con la base de datos MySQL. PHP nació como un lenguaje sencillo para páginas personales (de ahí sus siglas en inglés Personal Home Page) y en muy poco tiempo se convirtió en una de las tecnologías más utilizadas en la Web. PHP trae una gran cantidad de funciones predefinidas para trabajar con las tecnologías más comunes como Apache, MySQL, Oracle, ODBC, PDF, GZip, etc. Todo esto hace que programar cualquier tipo de aplicación sea muy sencillo. PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo Web y puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor Web, tomando el código en PHP como su entrada y creando páginas Web como salida. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. [6]

Ventajas [7]

- Es un lenguaje multiplataforma.
- Tiene manejo de excepciones (desde PHP5).
- Biblioteca nativa de funciones.
- Permite técnicas de programación orientada a objetos.
- Amplia documentación en su página oficial.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Es libre por lo que se presenta como una alternativa de fácil acceso para todos.

1.4.3 Lenguajes de programación JavaScript (del lado del navegador)

JavaScript es un lenguaje interpretado, multiplataforma, orientado a eventos con manejo de objetos, cuyo código se incluye directamente en el mismo



documento, usado para el desarrollo de aplicaciones cliente-servidor en páginas HTML.

Originalmente denominado Live Script, fue introducido por Netscape, mediante su navegador 2.0. Poco tiempo después aparece el VBScript (Visual Basic Script) de Microsoft, un buen competidor.

Hasta entonces ya se usaba HTML y Java, pero la aparición del JavaScript produjo una importante revolución, ya que dio al usuario la posibilidad crear aplicaciones "online" es decir modificar páginas web en tiempo real, sin usar CGI's.

Las principales características de JavaScript son:

- Es simple, no hace falta tener unos amplios conocimientos de programación para poder hacer un programa en JavaScript.
- Está orientado a objetos de forma limitada ya que no maneja los conceptos como la herencia, los métodos,... como el C++. Se pueden definir diferentes eventos que producirán una salida, como por ejemplo: presionar un botón, pasar el puntero del mouse sobre un determinado texto o el simple hecho de cargar la página web, eventos que darán una gran versatilidad a la hora de crear programas en JavaScript.
- Es dinámico, se basa en responder a eventos, esos eventos son producidos por el propio usuario y el JavaScript reacciona a ellos en tiempo real. Por ejemplo, podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor un página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario.

Sus principales aplicaciones son:

- Responder a eventos locales dentro de la página, como apretar un botón.
- La realización de cálculos en tiempo real.
- La validación de formularios dentro de una página.



- La personalización de la página por el usuario, que le permitirá tener una página web a su medida.
- La inclusión de datos del propio sistema, como son la hora y la fecha.

Aunque según pasa el tiempo sus aplicaciones se van incrementando. **[8]**

1.4.4 Lenguaje de programación HTML (del lado del navegador).

El HTML (Hyper Text Markup Language) es el lenguaje con el que se escriben las páginas web. Es un lenguaje de hipertexto, es decir, un lenguaje que permite escribir texto de forma estructurada, y que está compuesto por etiquetas, que marcan el inicio y el fin de cada elemento del documento. Un documento hipertexto no sólo se compone de texto, puede contener imagen, sonido, video, etc., por lo que el resultado puede considerarse como un documento multimedia. Los documentos HTML deben tener la extensión html o htm, para que puedan ser visualizados en los navegadores (programas que permiten visualizar las páginas web).

Los navegadores se encargan de interpretar el código HTML de los documentos, y de mostrar a los usuarios las páginas web resultantes del código interpretado. **[9]**

1.4.5 Framework de desarrollo Web

En general, con el término framework, nos estamos refiriendo a una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Un framework se puede considerar como una aplicación genérica incompleta y configurable a la que se le puede añadir las últimas piezas para construir una aplicación concreta. Entre los objetivos principales de un framework se encuentran acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un framework Web, es un conjunto de componentes (por ejemplo clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas web. El propósito de un framework es ayudar y facilitar el proceso de desarrollo



de aplicaciones. Debe permitir desarrollar la aplicación rápido y fácilmente y debe resultar en una aplicación superior finalizada.

Los Frameworks son importantes en todas las fases de desarrollo, desde el diseño hasta el desarrollo y quizá más en el mantenimiento continuo. **[10]**

1.4.5.1 Ext JS (versión 3.3.0)

Ext JS es una librería de JavaScript construida para el desarrollo veloz de aplicaciones Web para los diferentes tipos de navegadores actuales, usando técnicas como Ajax, DHTML y manipulación del DOM. **[11]**

Esta librería incluye:

- Componentes UI del alto performance y personalizables.
- Modelo de componentes extensibles.
- Un API fácil de usar.
- Licencias Open Source (GPL) y comerciales.

Ventajas

- Una de las grandes ventajas de utilizar Ext JS es que permite crear aplicaciones complejas utilizando componentes predefinidos.
- Evita el problema de tener que validar el código para que funcione bien en cada uno de los navegadores (Firefox, IE, Safari, Opera etc.).
- El funcionamiento de las ventanas flotantes lo pone por encima de cualquier otro.
- Relación entre Cliente-Servidor balanceado. Se distribuye la carga de procesamiento, permitiendo que el servidor pueda atender más clientes al mismo tiempo.
- Eficiencia de la red: Disminuye el tráfico en la red pues las aplicaciones cuentan con la posibilidad de elegir qué datos desea transmitir al servidor y viceversa (Criterio este que puede variar con el uso de aplicaciones de precarga).

Comunicación asíncrona. En este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una



acción del usuario, dándole la libertad de cargar información sin que el cliente se dé cuenta. [12]

1.4.5.2 OpenLayers (versión 2.10)

OpenLayers es una librería de JavaScript para la visualización de mapas en el navegador, altamente extensible, que sirve como la base de todas las interfaces de mapeo Web, accede a los datos a través de estándares de la industria, dejándola libre de dependencias de servidor, es liberado bajo una licencia estilo BSD. [13]

1.4.6 Estilo Arquitectónico

Los sistemas o arquitecturas en capas constituyen uno de los estilos que aparecen con mayor frecuencia mencionados como categorías mayores del catálogo o por el contrario, como una de las posibles imágenes de algún estilo envolvente. Definen el estilo en capas como una organización jerárquica tal, que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones de la inmediatamente inferior.

La arquitectura por capas es un estilo de arquitectura en la que el objetivo primordial es la separación de la lógica de negocio de la lógica de diseño, un ejemplo básico es separar la capa de datos, de la capa de presentación al usuario. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio, sólo se ataca al nivel requerido sin tener que revisar el código mezclado. Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de los niveles, simplemente es necesario conocer las API que existen entre niveles.

El diseño de sistemas informáticos suele usar las arquitecturas multinivel o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables, (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

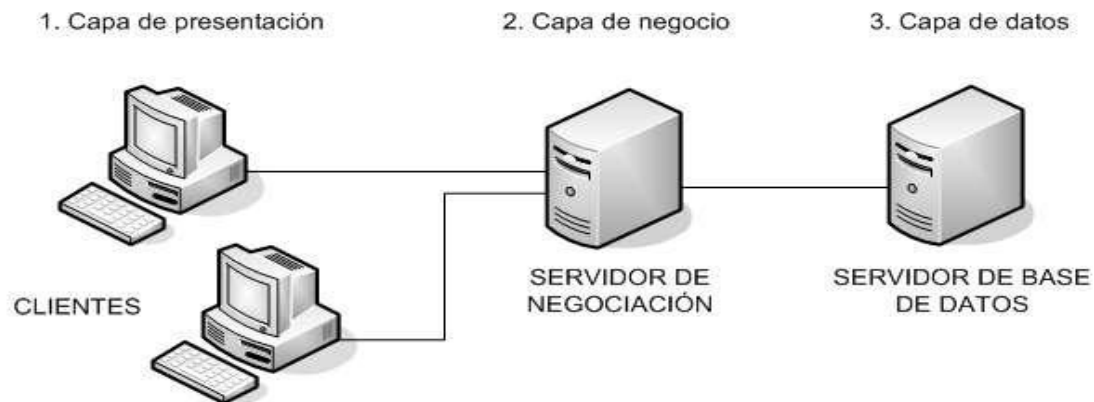


Figura 1: Representación de la arquitectura en tres capas.

Capas o niveles:

- **Capa de presentación o interface:** es la capa de 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 comprobar que no hay errores de formato). Esta capa se comunica únicamente con la del negocio.
- **Capa de lógica o de negocio:** es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio e incluso lógica del negocio, pues es aquí donde se establecen 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 solicitar al gestor de bases de datos para almacenar o recuperar datos de él.
- **Capa de datos:** es donde se ubican los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de los mismos, reciben solicitudes de almacenamiento o de recuperación de información desde la lógica del negocio.

Todas estas capas pueden residir en un único ordenador, esto no sería lo normal, 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 negocio 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 la base de datos aumenta, pueden separarse en varios ordenadores los cuáles recibirán las peticiones del ordenador en que 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. [2]

1.5 Sistema Gestor de Base de Datos

Un Sistema Gestor de Bases de Datos (SGBD) o DBMS (Database Management System) 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. Algunos ejemplos de SGBD son Oracle, DB2, PostgreSQL, MySQL, MS SQL Server, etc. [14]

1.5.1 PostgreSQL (versión 8.0)

Es un servidor de base de datos relacional orientada a objetos de software libre, publicado bajo la licencia BSD40. Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

Principales características:

- Alta concurrencia:

Mediante un sistema denominado MVCC (Acceso concurrente Multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo “commit”.



Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

- Amplia variedad de tipos nativos.

PostgreSQL provee nativamente soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.

Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL. Algunos ejemplos son los tipos de datos GIS creados por el proyecto PostGIS. **[15]**

1.6 Metodologías de Desarrollo

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayuda a la documentación para el desarrollo de software. Pueden ser comparadas con un plan de contingencia en el que se va indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además quiénes deben participar en el desarrollo de las actividades y qué papel deben de tener. Detallan además la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

1.6.1 Metodología 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, preocupándose por el aprendizaje de los desarrolladores, y propiciando 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. [16]

Características De La Metodología 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 en 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 escrito de esta manera es más importante que la posible pérdida de productividad inmediata.
- Frecuente integración del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- Corrección de todos los errores antes de añadir nueva funcionalidad. Hacer entregas frecuentes.
- **Refactorización del código**, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad 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 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. La simplicidad y la comunicación son extraordinariamente complementarias. Con una buena 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 completa, especialmente si se puede reducir el equipo de programadores. [17]

Fases de la metodología XP

Planificación

Historias de usuario: El primer paso de cualquier proyecto que siga la metodología X.P es definir las historias de usuario con el cliente. Las historias de usuario tienen la misma finalidad que los casos de uso pero con algunas diferencias: Constan de 3 ó 4 líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalles; no se debe hablar ni de posibles algoritmos para su implementación ni de diseños de base de datos adecuados, etc. Son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que especifica la historia de usuario.

Reléase planning: Después de tener ya definidas las historias de usuario es necesario crear un plan de publicaciones, en inglés "Reléase plan", donde se indiquen las historias de usuario que se crearán para cada versión del programa y las fechas en las que se publicarán estas versiones. Un "Reléase plan" es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa.

Iteraciones: Todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el "Reléase planning" que serán implementadas.



Diseño

Diseños simples: La metodología XP sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

Glosarios de términos: Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

Riesgos: Si surgen problemas potenciales durante el diseño, XP sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.

Tarjetas C.R.C. El uso de las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedural clásica.

Codificación

- Como ya se dijo en la introducción, el cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de XP. A la hora de codificar una historia de usuario su presencia es aún más necesaria. Los clientes son los que crean las historias de usuario y negocian los tiempos en los que serán implementadas. Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los test (pruebas) que verifiquen que la historia implementada cumple la funcionalidad especificada.
- La codificación debe hacerse atendiendo a estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.
- Crear test que prueben el funcionamiento de los distintos códigos implementados nos ayudará a desarrollar dicho código. Crear estos test



antes nos ayuda a saber qué es exactamente lo que tiene que hacer el código a implementar y se sabrá que una vez implementado pasará dichos test sin problemas. Se puede dividir la funcionalidad que debe cumplir una tarea a programar en pequeñas unidades, de esta forma se crearán primero los test para cada unidad y a continuación se desarrollará dicha unidad, así poco a poco conseguiremos un desarrollo que cumpla todos los requisitos especificados.

- La optimización del código siempre se debe dejar para el final. Hay que hacer que funcione y que sea correcto, más tarde se puede optimizar.
- XP afirma que la mayoría de los proyectos que necesiten más tiempo extra que el planificado para ser finalizados no podrán ser terminados a tiempo se haga lo que se haga, aunque se añadan más desarrolladores y se incrementen los recursos. La solución que plantea XP es realizar un nuevo "Reléase plan" para concretar los nuevos tiempos de publicación y de velocidad del proyecto.

Pruebas

- Se deben crear las aplicaciones que realizarán los test con un entorno de desarrollo específico para test.
- Un punto importante es crear test que no tengan ninguna dependencia del código que en un futuro evaluará. Hay que crear los test abstrayéndose del futuro código, de esta forma aseguraremos la independencia del test respecto al código que evalúa.

Test de aceptación: sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear "Test de aceptación"; estos test son creados y usados por los clientes para comprobar que las distintas historias de usuario cumplen su cometido. **[18]**



1.7 Herramientas utilizadas

Hoy en día, llevar a cabo el desarrollo de un buen software depende de un gran número de actividades y etapas donde elegir la herramienta adecuada para el equipo influye directamente en el futuro éxito del producto.

1.7.1 Sistema de Información Geográfica (SIG)

Un Sistema de Información Geográfica (SIG o GIS, en su acrónimo inglés [Geographic Information System]) es una integración organizada de hardware, software y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y de gestión. [19]

1.7.2 Servidor de Mapas Geoserver (versión 2.0.2)

Es un servidor de código abierto escrito en Java que permite a los usuarios compartir y editar datos geoespaciales. Diseñado para la interoperabilidad, publica los datos de cualquier fuente importante de datos espaciales usando estándares abiertos. Geoserver es la implementación de referencia de los estándares Open Geospatial Consortium (OGC), Web Feature Service (WFS) y Web Coverage Service (WCS), y está certificado como implementación de alto rendimiento del estándar Web Map Service (WMS). Geoserver es uno de los componentes esenciales de la Web Geoespacial. [20]

1.7.3 Paquete Tecnológico XAMPP v1.7.3

Xampp es un servidor independiente de la plataforma en la cual se esté ejecutando, es un software totalmente libre y se encuentra bajo la licencia GNU/GPL (GNU General Public License). Básicamente Xampp consiste en una recopilación de aplicaciones y servidores, tal es el caso que encontramos el servidor web Apache, el motor de Bases de Datos MySQL, los lenguajes de programación PHP y Perl y otros servicios más. El nombre Xampp proviene del acrónimo **X** (para cualquiera de los Sistema Operativos en la que se está ejecutando), **A**pache, **M**ySQL, **P**HP, **P**erl [21]

1.7.4 Visual Paradigm (versión 5.3)

Visual Paradigm es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño, construcción, pruebas y



despliegue. Está dotada de una buena cantidad de productos o módulos para facilitar el trabajo durante la confección de un software, lo cual garantiza la calidad del producto final. Permite dibujar todos los tipos de diagramas de clases, generar código desde diagramas, código inverso y generar documentación. Se prefiere su utilización en el desarrollo del módulo por ser una herramienta no propietaria, multiplataforma, amigable en su entorno, lo que facilita su interoperabilidad con otras aplicaciones además de que tiene un uso de un lenguaje estándar común a todo el equipo de desarrollo lo que facilita la comunicación. [22]

1.7.5 Embarcadero ER/Studio (versión 8.0)

Es una herramienta líder para el modelado de datos y el mantenimiento de bases de datos transaccionales, su diseño multinivel permite controlar, documentar y desplegar rápidamente cambios en el diseño en las principales plataformas de bases de datos, da soporte para completo para el ciclo de vida de la base de datos y ayuda a las organizaciones en la toma de decisiones.

[23]

Conclusiones parciales

En este capítulo se hizo referencia a los sistemas existentes hoy en día relacionados a los derrames de hidrocarburos, a las tecnologías utilizadas para el desarrollo de Aplicaciones Web. Se realizó un estudio detallado de las principales herramientas con que se desarrollará el sistema, seleccionándose XAMPP como paquete tecnológico, utilizándose de éste el Apache como servidor web ya que es de código abierto, es multiplataforma, es altamente configurable y PHP como lenguaje de programación del lado del servidor ya que es Open Source, es decir, se tiene acceso al código fuente, ofrece soporte para conectarse con una gran variedad de base de datos como MySQL, PostgreSQL, Oracle, Microsoft SQL Server, entre otras y es un lenguaje de programación orientado a objetos; se utilizará PostgreSQL porque es un servidor de base de datos de código abierto, propone un tamaño ilimitado para el almacenamiento de los datos, lo que da la medida de un gestor de bases de datos robusto, es estable, flexible, se puede extender su funcionalidad y tiene gran compatibilidad con diversos sistemas operativos,



permite que mientras un proceso escribe en una tabla, otros accedan a la misma sin necesidad de bloqueos.

Se seleccionó la metodología XP ya que permite adaptarse a las necesidades de cualquier equipo de desarrollo, durante su ciclo de vida se le realizan pruebas contantes al sistema, el cliente se integra a la realización del proyecto, es apropiada para entornos volátiles; se utilizó Ext JS ya que es una librería avanzada que permite el desarrollo rápido de aplicaciones web con un aspecto novedoso y una arquitectura flexible utilizando componentes predefinidos, no es necesario validar el código para que funcione bien en los diferentes navegadores (Firefox, IE, Google Chrome, Opera); se utilizó OpenLayers ya que facilita poner un mapa dinámico en cualquier página web, permite mostrar fragmentos del mapa y marcadores cargados de cualquier fuente, se ha desarrollado para promover el uso de la información geográfica de todo tipo, completamente libre, implementa una API de JavaScript para la construcción de ricas aplicaciones basadas en web geográficas, de forma similar a la de Google Maps y MSN Virtual Earth API, con una diferencia importante, OpenLayers es libre, desarrollado por y para la comunidad de software Open Source; se empleó Geoserver como servidor de mapas empleado para publicar los datos geoespaciales ya que es un servidor de código abierto escrito en Java que implementa los estándares Open Geospatial Consortium (OGC), Web Feature Service (WFS) y Web Coverage Service (WCS), y está certificado como implementación de alto rendimiento del estándar Web Map Service (WMS).



Capítulo 2: Planificación y Diseño

2.1 Introducción

En este capítulo, se establece la fase de planificación y diseño, se especifican las necesidades del cliente, las funcionalidades que serán objeto de automatización 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 posterior culminación.

Tabla 2.1: Personal relacionado con el sistema

Personal Relacionado con el sistema	Justificación
Especialista del Medio Ambiente	Es la persona que se encarga de elaborar los reportes e informes relacionados con los derrames de hidrocarburos
Director de Tecnología y Desarrollo	Es la persona que se encarga de supervisar los reportes e informes
Administrador	Es la persona que se encarga de gestionar los usuarios del sistema
Desarrollador	Es la persona responsable de llevar a cabo la implementación del sistema

2.2 Lista de reserva

La Lista de Reserva del Producto es el primer artefacto generado en la etapa de Captura de Requisitos. Cuando un proyecto comienza es muy difícil tener claro todos los requerimientos sobre el producto, debido a eso la metodología XP nos brinda la posibilidad de que esta lista pueda crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente, con la restricción de que sólo puede cambiarse entre iteraciones. EL objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y



Capítulo 2 Planificación y Diseño

competitivo posible, para esto la lista debe acompañar los cambios en el entorno y el producto. A continuación, se enumeran las funcionalidades que el sistema debe ser capaz de cumplir.

Tabla 2.2 Lista de reserva del producto

Número	Descripción de las historias de usuarios	Prioridad
1	Insertar Entidad Responsable	Alta
2	Modificar Entidad Responsable	Alta
3	Mostrar Entidad Responsable	Alta
4	Insertar Derrame	Alta
5	Modificar Derrame	Alta
6	Mostrar Derrame	Alta
7	Eliminar Derrame	Alta
8	Insertar Reporte de Derrame	Alta
9	Modificar Reporte de Derrame	Alta
10	Mostrar Reporte de Derrame	Alta
11	Eliminar Reporte de Derrame	Alta
12	Insertar Reporte de Incidente	Alta
13	Modificar Reporte de Incidente	Alta
14	Mostrar Reporte de Incidente	Alta
15	Eliminar Reporte de Incidente	Alta
16	Insertar Informe Final	Alta
17	Modificar Informe Final	Alta



Capítulo 2 Planificación y Diseño

18	Mostrar Informe Final	Alta
19	Eliminar Informe Final	Alta
20	Insertar Usuarios	Alta
21	Modificar Usuarios	Alta
22	Mostar Usuarios	Alta
23	Eliminar Usuarios	Alta
24	Autenticar Usuario	Media
25	Mostrar Derrames en un Mapa de del Puerto	Alta
26	Graficar la cantidad de derrames ocurridos por año	Media
27	Graficar el volumen de hidrocarburos derramados en un año	Media
RNF(Requisitos no funcionales)		
Usabilidad		
RNF1	Facilidad de uso por parte de los usuarios: el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada, y debe ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.	
RNF2	Especificación de la terminología utilizada: el sistema debe adaptarse al lenguaje y términos utilizados por los clientes en la rama abordada con vista a una mayor comprensión por parte del cliente de la herramienta de trabajo.	
RNF3	Emplear perfiles de usuario: diferenciar las interfaces y opciones para los usuarios que accedan al sistema según los diferentes roles que estos tengan dentro del sistema.	



Capítulo 2 Planificación y Diseño

Fiabilidad	
RNF4	Seguridad de las bases de datos: la seguridad de la base de datos está a nivel de roles, con el fin de mantener la integridad de los datos en función del acceso de cada uno de ellos, trayendo consigo además la protección de la información.
RNF5	Servicios web restringidos: los servicios web que brinde el sistema deben estar restringidos a grupos de usuarios definidos y aprobados previamente.
RNF6	Políticas de seguridad por usuario y rol: el sistema debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del mismo en dependencia del nivel de autorización que presente un usuario determinado.
Eficiencia	
RNF7	Cantidad de conexiones: el sistema debe soportar una conexión simultánea.

2.3 Historias de Usuarios

La HU, es la técnica utilizada en XP para detallar los requisitos del software; es el resultado directo del intercambio entre el cliente y desarrolladores a través de reuniones donde las conocidas tormenta de ideas arrojan no solo los requerimientos, sino también las posibles soluciones; representa una forma rápida de administrar las necesidades del cliente 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.

Historia de Usuario	
Número: 7	Usuarios: Administrador, Especialista y Director.



Capítulo 2 Planificación y Diseño

Nombre: Autenticar usuarios	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Aurelio Calvo Burke	
Descripción: Los usuarios del Sistema ingresan sus datos para entrar al sistema (nombre de usuario y contraseña). El sistema verifica que los datos estén correctos, en caso de que no sean correctos los mismos la aplicación muestra un mensaje de error.	
Observaciones: Confirmado por el cliente.	

Ver resto de las HU en [Anexo 1](#).

2.4 Planificación de entregas

Se establece la prioridad de cada HU 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 de usuario 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.

Tabla 2.4 Estimación de esfuerzo por HU

Historias de Usuarios	Puntos de Estimación (semanas)
Gestionar Entidad Responsable	1 semana
Gestionar Derrame	1 semana
Gestionar Reportes de Derrame	1 semana



Capítulo 2 Planificación y Diseño

Gestionar Reportes de Incidente	1 semana
Gestionar Informe Final	1 semana
Gestionar Cuenta de Usuario	1 semana
Autenticar Usuario	1 semanas
Mostrar Derrames en un mapa	2 semana
Graficar cantidad de derrames por año	1 semana
Graficar cantidad de hidrocarburo derramado por año	1 semana

2.5 Planificación de Iteraciones

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se procede a realizar la planificación de la etapa de implementación del sistema, apoyándose en el tiempo e intentando concentrar las funcionalidades relacionadas en una misma iteración. En este plan se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su terminación. El plan de iteraciones puede contener indicaciones sobre cuáles HU se incluirán en un release, lo cual debe ser consistente con el contenido de una o dos iteraciones. En relación con lo antes tratado se decide realizar el sistema en 3 iteraciones, las cuales se explican de forma detalla a continuación.

Primera iteración: Esta iteración tiene como objetivo darle cumplimiento a las HU que se consideraron de importancia inicial para el desarrollo del sistema. Al concluir esta iteración se contará con las funcionalidades descritas en las historias de Usuario 1 y 2 las cuales hacen alusión:

1. A la gestión de la entidad responsable brindando la posibilidad de insertar, modificar y mostrar las entidades responsables del derrame.
2. A la gestión de derrames brindando la posibilidad de insertar, modificar, mostrar y eliminar un derrame.



Capítulo 2 Planificación y Diseño

Segunda iteración: Esta iteración tiene como objetivo darle cumplimiento a las HU que se consideraron de importancia capital para el desarrollo del sistema. Al concluir esta iteración se contará con las funcionalidades descritas en las Historias de Usuario 3, 4, 5 y 6, las cuales hacen alusión:

1. A la gestión de los reportes de derrames brindando la posibilidad de crear, modificar, mostrar y eliminar reportes de derrame.
2. A la gestión de los reportes de incidentes brindando la posibilidad de insertar, modificar, mostrar y eliminar reportes de incidentes.
3. A la gestión de los informes finales de cada derrame ocurrido brindando la opción de insertar, modificar, mostrar y eliminar un informe final.
4. A la gestión de las cuentas de usuario del sistema brindando la posibilidad de crear, modificar, mostrar y eliminar una cuenta de usuario del sistema.

Esta iteración será integrada al resultado de las anteriores iteraciones obteniendo una versión del sistema que incluye las principales funcionalidades.

Tercera iteración: Esta iteración tiene como objetivo final dar cumplimiento a las HU 7, 8, 9 y 10. Dichas historias brindan las funcionalidades de autenticación de usuario en el sistema, mostrar en un mapa los derrames ocurridos, mostrar mediante graficas la cantidad de derrames ocurridos por años y la cantidad de hidrocarburo derramado por año. Esta iteración será integrada al resultado de las anteriores iteraciones obteniendo una versión final del sistema que a partir de este momento será puesto a prueba para evaluar el desempeño del mismo.

2.6 Plan de duración de iteraciones

Con respecto a las Historias de Usuario previamente presentadas se realizó una planificación en 3 iteraciones basándose en el tiempo y procurando obtener la funcionalidad relacionada en la misma iteración.



Tabla 2.5 Iteraciones

iteración	Orden de implementación de las historias de usuario	Duración Total
1	Gestionar Entidad Responsable	1+1=2
	Gestionar Derrame	
2	Gestionar Reporte de Derrame	1+1+1+1=4
	Gestionar Reporte de Incidente	
	Gestionar Informe Final	
	Gestionar Cuentas de Usuario	
3	Autenticar Usuario	1+2+1+1=5
	Mostrar derrames en un mapa	
	Graficar cantidad de derrames por año	
	Graficar volumen de hidrocarburo derramado en un año	

2.7 Valoración de Riesgo

La planilla de lista de riesgos, es el documento que se genera de la actividad de valoración de riesgos. En ella se definen los posibles riesgos que actuarán sobre el proceso de desarrollo de software, así como la estrategia trazada para mitigarlos, además de un plan de contingencia que describe qué curso seguirán las acciones si el riesgo se materializa. Para comprender cuáles son los datos que se recogerán en esta planilla, se muestra la misma gráficamente.



Capítulo 2 Planificación y Diseño

**Tabla 2.7 Valoración de riesgos
Plan de Contingencia.**

Riesgos	Tipos de Riesgos	Impacto	Descripción	Efectos	Mitigación de Riesgos
Poco dominio de las herramientas a utilizar	Tecnológico	Retraso en el plazo de entrega del sistema	Al estudiar la herramientas se llegó a la conclusión de que son las más adecuadas	Serias	Amplio estudio de la bibliografía referente a la tecnología a utilizar
Poca comunicación con el cliente.	Organización	Pobre captura de requisitos.	El cliente dispone de poco tiempo para realizar reuniones y determinar los requisitos del sistema	Serias	Planificación de reuniones con el cliente

Riesgo 1

En caso de que no se logre mitigar este riesgo se debe tomar como solución utilizar herramientas y tecnologías que se conozcan.

Riesgo 2

Si no se logra el contacto con el cliente se deberá considerar estudiar el negocio a través del Sistema Informativo OSDES Derrame HC.

2.8 Tarjetas Clases – Responsabilidades - Colaboración (CRC).

Las tarjetas CRC se realizan para facilitar la comunicación y documentación de los resultados. Permiten una total participación y contribución del equipo de desarrollo en el diseño. Cada tarjeta CRC representa clases, donde el nombre de cada clase se ubica en forma de título en la parte superior de la tarjeta, sus atributos y 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.

Tabla 2.6 Tarjeta CRC de Gestionar Entidad Responsable

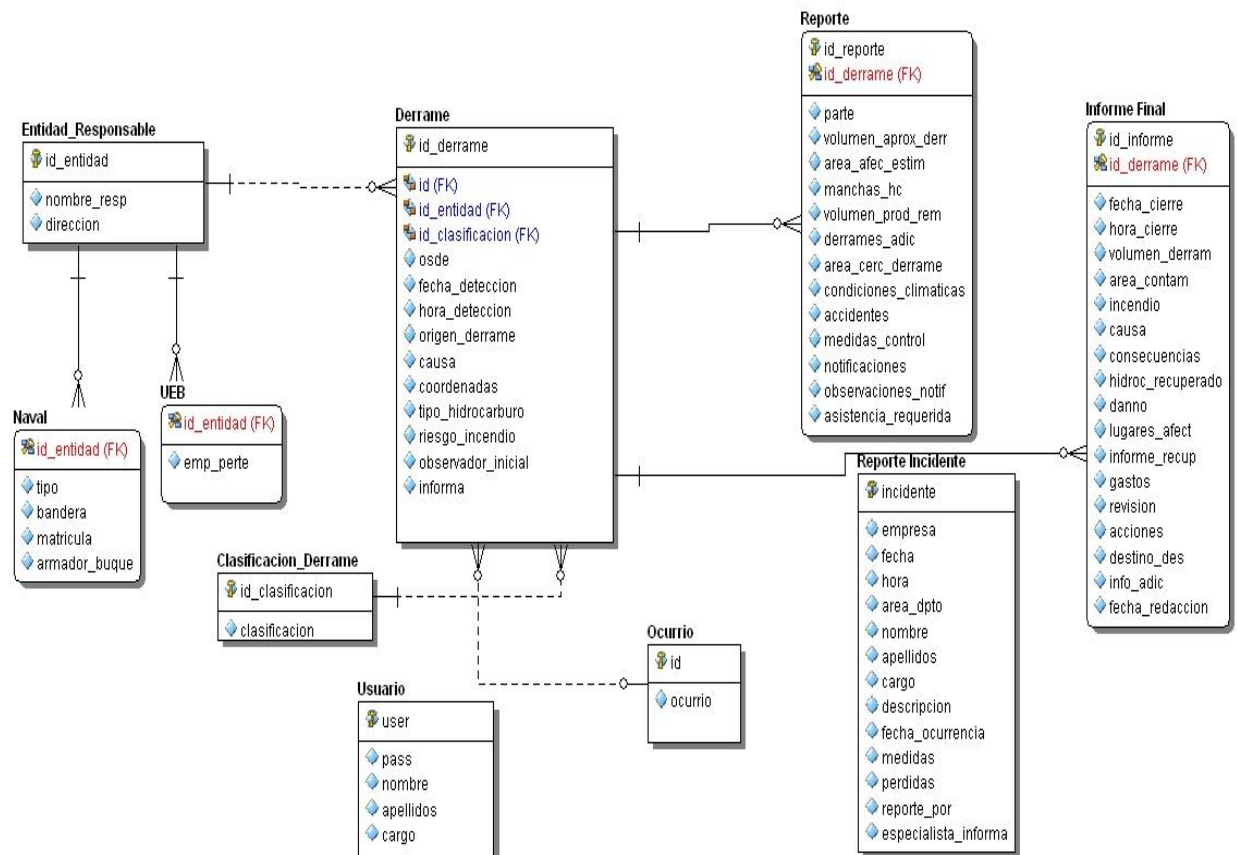


Clase: entidad_responsable	
Description: Guarda los datos de la entidad responsable	
Attributes:	
Name	Description
id_entidad	identificador de entidad responsable
nombre	
direccion	
Responsibilities:	
Name	Collaborator
Insertar entidad responsable	
Modificar entidad responsable	
Mostrar entidad responsable	

Ver Tabla CRC en [Anexos 2](#)

2.9 Modelo de Datos

En este modelo se presentan todas las tablas con la que interactúa el sistema propuesto en esta investigación; se recogen y modelan todos los datos con lo que dispondrá la aplicación.





Conclusiones parciales

Con la culminación de este capítulo se han desarrollado las bases con las que se sustentarán las necesidades del cliente, se identificaron las HU con la participación conjunta del cliente y usuarios, destacando la planificación de cada HU por la prioridad de sus iteraciones, o sea, a partir del esfuerzo de las mismas. También se crearon las tarjetas CRC las cuales van a facilitar y documentar los resultados.



Capítulo 3 Desarrollo y Prueba

3.1 Introducción

En este capítulo abordaremos lo relacionado con la fase de implementación y pruebas siguiendo la Metodología XP. Se describen cada una de las tareas de ingeniería para cumplir con el desarrollo de cada una de las HU definidas, se presentan las tarjetas clases, responsabilidades y colaboradores, y se mostrarán las pruebas de aceptación elaboradas por el cliente para comprobar que la aplicación funcione correctamente.

3.2 Desarrollo de las iteraciones

Durante la fase planificación y diseño fueron detalladas las historias de usuario correspondientes a cada una de las iteraciones a desarrollar, teniendo en cuenta las prioridades y restricciones de tiempo, previstas por el cliente.

3.2.1 Tareas de ingeniería por HU

Las HU se descomponen en tareas de programación o de ingeniería, y a su vez, estas son asignadas al equipo de desarrollo para su implementación. Las tareas no tienen que ser entendidas necesariamente por el cliente, pues las mismas, sólo son utilizadas por los miembros del equipo de desarrollo, por lo que pueden ser escritas en lenguaje técnico. Estas se representan mediante las tarjetas de tareas.

Tabla 3.1 Distribución de tareas por cada historia de usuarios

Historias de Usuario	Tarea de Ingeniería
Gestionar Entidad Responsable	Insertar Entidad Responsable. Modificar Entidad Responsable. Mostrar Entidad Responsable.



Capítulo 3 Implementación y Prueba

Gestionar Derrame	Insertar Derrame. Modificar Derrame. Mostrar Derrame. Eliminar Derrame.
Gestionar Reporte de Derrame	Insertar Reporte de Derrame. Modificar Reporte de Derrame. Mostrar Reporte de Derrame. Eliminar Reporte de Derrame.
Gestionar Reporte de Incidente	Insertar Reporte de Incidente. Modificar Reporte de Incidente. Mostrar Reporte de Incidente. Eliminar Reporte de Incidente.
Gestionar Informe Final	Insertar Informe Final. Modificar Informe Final. Mostrar Informe Final. Eliminar Informe Final.
Gestionar Cuenta de Usuario	Insertar Usuario. Modificar Usuario. Mostrar Usuario. Eliminar Usuario.
Autenticar Usuario	Insertar los datos para entrar al sistema (nombre de usuario y



Capítulo 3 Implementación y Prueba

	contraseña).
Mostrar derrame en un mapa	Mostrar en un mapa del puerto los derrames ocurridos.
Graficar cantidad de derrames de hidrocarburos ocurridos en un año	Mostrar en una gráfica la cantidad de derrames ocurridos en un año.
Graficar el volumen de hidrocarburo derramado en un año	Mostrar en una gráfica la cantidad de hidrocarburo derramados en un año.

Tabla 3.2 Historias de Usuarios incluidas en la primera iteración

Iteración	Orden de Implementación por Historias de Usuario	Semanas
1	Gestionar Entidad Responsable	1
	Gestionar Derrame	1

Tabla 3.3 Historias de Usuarios incluidas en la segunda iteración

Iteración	Orden de Implementación por Historias de Usuario	Semanas
2	Gestionar Reporte de Derrame	1
	Gestionar Reporte de Incidente	1
	Gestionar Informe Final	1
	Gestionar Cuenta de Usuario	1



Capítulo 3 Implementación y Prueba

Tabla 3.4 Historias de Usuarios incluidas en la tercera iteración

Iteración	Orden de Implementación por Historias de Usuario	Semanas
3	Autenticar Usuario	1
	Mostrar los derrames de hidrocarburos en un mapa	2
	Graficar cantidad de derrames de hidrocarburos ocurridos en un año	1
	Graficar volumen de hidrocarburo derramado en un año	1

Ver Tareas de Ingeniería en los [Anexos 3](#).

3.3 Prueba

En la metodología XP las pruebas juegan un papel importante, pues estas permiten la comprobación continua del código. El desarrollo constante de las pruebas da lugar a que se desarrolle un software con mayor calidad dando una mayor seguridad de lo que se está haciendo, dividiendo las pruebas en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y las pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñada por el cliente.

3.3.1 Prueba de aceptación.

En esta planilla se escriben las pruebas ejecutadas según la historia de usuario seleccionada para realizar la comprobación y validar las funcionalidades del sistema, y de esta forma saber si está apto para ser entregado. A continuación se muestran pruebas de aceptación de las historias de usuarios.

Tabla 3.5 Prueba de Aceptación para la HU: Autenticar Usuario

Prueba de Aceptación
HU: Autenticar Usuario.
Nombre: Prueba para comprobar la Autenticación de Usuario.
Descripción: Validación de entrada de datos de los usuarios.



Capítulo 3 Implementación y Prueba

Condiciones de ejecución: El usuario debe escribir su nombre de usuario y contraseña.

Entrada / Pasos de ejecución: El usuario escribe su nombre de usuario y contraseña y luego da clic en el botón Aceptar.

Resultado esperado:

- Si el usuario tiene acceso a la aplicación e inserta correctamente sus datos accederá sin problemas al Sistema.

Se produce un error en caso de que:

- Se inserten los datos de un usuario o contraseña no válida para el sistema.
- Se dé clic en el botón Aceptar y los campos estén vacíos.

Evaluación de la prueba: Aceptada

Ver las demás pruebas de aceptación en los [Anexos 4](#).

Conclusiones parciales

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



Capítulo 4: Estudio de Factibilidad

4.1 Introducción

En la actualidad es muy importante realizar un estudio de factibilidad antes de empezar a desarrollar un proyecto. En general, ningún producto informático está exento de riesgos durante la concepción del proyecto. Por tanto, es necesario minimizar recursos humanos, materiales y financieros; de ahí que es de vital importancia estimar la relación costo-beneficio, así como el esfuerzo, capital humano y el tiempo de desarrollo que se emplea en la ejecución de estos.

En este capítulo se expone el estudio de factibilidad del proyecto, centrado en estimaciones de esfuerzo humano, tiempo de desarrollo para su ejecución y costo. Se estiman los beneficios tangibles e intangibles que representan para el sistema propuesto, un análisis de costos y beneficios.

4.2 Evaluación Costo-Beneficio

La mayoría, por no decir todos los proyectos de informática, son evaluados según el criterio de Costo-Beneficio. Esta Metodología, plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores:

- El costo, que involucra la implementación de la solución informática, adquisición y puesta en marcha del sistema hardware/software y los costos de operación asociados.
- La efectividad, que se entiende como la capacidad del proyecto para satisfacer la necesidad, solucionar el problema o lograr el objetivo para 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 de cumplimiento del objetivo). El desarrollo de un producto informático, siempre tiene un costo.

Este puede estar justificado por los beneficios tanto tangibles como intangibles que origina el mismo. En este proceso, se necesita de una selección adecuada de los elementos más convenientes para su evaluación.



4.3 Efectos Económicos

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

Efectos Directos

Positivos:

- Se agilizará el proceso de obtención del informe final y de los reportes de derrames y de incidentes ambientales.
- Se facilitará el proceso de impresión de reportes y del informe final.
- El proceso de cualquier tipo de búsqueda que se desee realizar será de manera sencilla y personalizada.
- El sistema es compatible con todos los navegadores más usados en la actualidad (Mozilla Firefox, Google Chrome).

Negativos:

- El sistema no es compatible con el navegador Internet Explorer.

Efectos Indirectos

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

Efectos Externos

Se obtuvo un producto que le facilitará al especialista del medio ambiente en el Puerto de Moa la gestión de información sobre derrames de hidrocarburo, beneficiar la disponibilidad y fiabilidad de los reportes e informes y que estos cumplan con lo establecido por el ministerio de Energía y Minas.

Intangibles



Capítulo 4 Estudio de Factibilidad

En la valoración económica siempre hay elementos como perjuicio o beneficio, pero al momento de ponderar en unidades monetarias esto resulta difícil o prácticamente imposible.

4.4 Elementos para identificar los Costos y Beneficios del Proyecto

Para la identificación de los costos y beneficios del proyecto que son pertinentes para su evaluación, es necesario definir una situación base o situación sin proyecto; la comparación de lo que sucede con proyecto versus lo que hubiera sucedido sin proyecto, definirá los costos y beneficios pertinentes del mismo.

Estos escenarios, resultan ser una herramienta determinante, puesto que ayudan en gran medida en la definición de los elementos necesarios para la evaluación. A continuación se analizan ambas situaciones.

Situación sin Proyecto

El especialista del medio ambiente en el puerto de Moa, cuando ocurre un derrame, redacta un reporte de incidente ambiental y luego lo presenta a la dirección de la empresa y envía al CITMA provincial, el cual es redactado cumpliendo los requisitos establecidos por la empresa. En dicho reporte la información expuesta es escasa y no cumple con los requisitos establecidos para este tipo de documento por parte del Ministerio de Energía y Minas. El sistema actual de gestión de información sobre derrames de hidrocarburo no garantiza la disponibilidad y fiabilidad del reporte de incidente.

Situación con Proyecto

Se cuenta con un sistema que agiliza la gestión de la información sobre derrames de hidrocarburo, garantiza la disponibilidad y fiabilidad de los reportes de incidentes ambientales, además de incorporar los reportes de derrame e informe final exigidos por el Ministerio de Energía y Minas. El sistema facilita visualizar en un mapa los derrames ocurridos y muestra mediante gráficas información tan importante como la cantidad de derrames y el volumen de hidrocarburo derramado por año.



4.5 Factibilidad Económica

El análisis de factibilidad económica identifica los costos y beneficios asociados con el proyecto. El mismo incluye cuatro categorías:

- Costo de desarrollo.
- Costos operacionales.
- Beneficios tangibles.
- Beneficios intangibles.

Esta investigación, englobará los costos de desarrollo y los costos operacionales en Evaluación Económica.

4.6 Evaluación Económica

Los principales factores a considerar para el cálculo de los costos son los relacionados al personal, hardware y software, los que se pueden calcular de diversas maneras, que muchas veces se limitan al buen criterio y a la experiencia.

Para determinar el costo económico del proyecto, se desglosaron los costos en moneda libremente convertible y en moneda nacional.

Tabla 4.1 Costos en moneda libremente convertible

Ficha de Costo	
	Precio(\$)
Costos moneda libremente convertible	
Costos directos	
Compra de equipos de cómputo	0.00
Alquiler de equipos de cómputo	0.00
Compra de licencia de software	0.00
Depreciación de equipos	25.00
Materiales directos	0.00



Capítulo 4 Estudio de Factibilidad

Subtotal	25.00
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	0.00
Gastos de distribución y ventas	
Participación en ferias o exposiciones	0.00
Gastos en transportación	0.00
Compra de materiales de propaganda	0.00
Subtotal	0.00
Total	25.00

Tabla 4.2 Costos en moneda nacional

Ficha de Costo	
	Precio(\$)
Costos moneda nacional	
Costos directos	
Salario del personal que laborará en el proyecto.	100.00
12,5% del total de gastos por salarios se dedica a la seguridad social.	0.00
9,09% de salario total, por concepto de vacaciones a acumular.	0.00
Gasto por consumo de energía eléctrica.	0.00



Capítulo 4 Estudio de Factibilidad

Gastos administrativos.	0.00
Subtotal	100.00
Costos Indirectos	
Know How	0.00
Subtotal	0.00
Total	100.00

La evaluación económica se efectúa conjuntamente con la que se puede llamar evaluación técnica del proyecto, que consiste en cerciorarse de la factibilidad técnica del mismo. En el análisis de la Factibilidad Técnica del proyecto, se pudo apreciar que:

- Se cuenta con hardware mayor a los requisitos mínimos.
- Se cuenta con la disponibilidad de software.

Se llega a la conclusión de que el proyecto es factible técnicamente. Normalmente no se encuentran problemas en relación con el mercado o la tecnología disponible que se empleará en la fabricación del producto; por tanto la decisión de inversión casi siempre recae en la evaluación económica. Los costos de los proyectos de informática son relativamente simples de cuantificar, no así los beneficios, que se presentan como ahorro de costos con respecto a la situación base.

4.6.1 Beneficios Tangibles

Los beneficios tangibles son los ingresos adicionales y/o reducción de costo que el nuevo sistema proveerá. Debido a que este proyecto no es construido con el propósito de comercializarse, no se posee ingresos perceptibles, no así la reducción de costos, pues el sistema proveerá mejoras en la gestión de información.

Estos elementos son de muy difícil cuantificación, por lo que en esta investigación se consideran solo como intangibles.



4.6.2 Beneficios y costos intangibles del proyecto.

Costo

- Resistencia al cambio.

Beneficios

- Mayor agilidad en la gestión de los reportes e informes sobre derrames de hidrocarburos.
- Menor tiempo empleado en el proceso.

La técnica seleccionada para evaluar la factibilidad del proyecto es la Metodología Costo-Efectividad. Dentro de la misma, 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, tomaremos como costo el tiempo en minutos empleados para resolver la gestión académica, y la variable sería la complejidad de las pruebas que se realizan durante este proceso.

Valores de la variable (solución manual)

- Crear Reporte de Incidente Ambiental. (10 min.).

Valores de la variable (solución con el sistema)

- Crear Reporte de Incidente Ambiental. (2 min).
- Crear Reporte de Derrame de Hidrocarburo. (2 min).
- Crear Informe Final. (2 min).

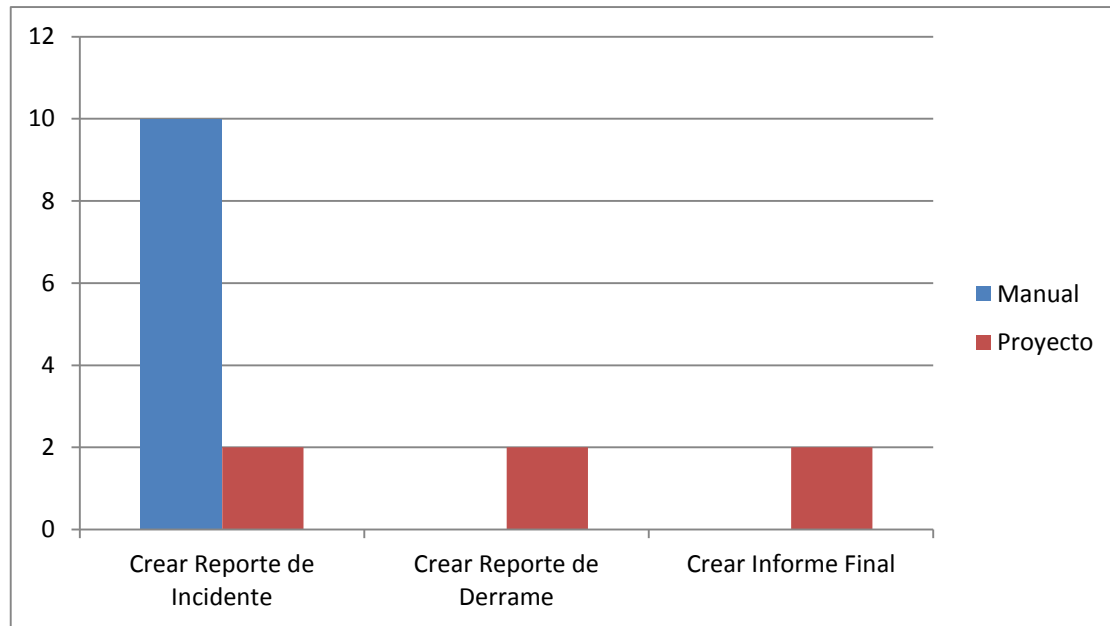


Figura 1 Gráfica de solución sin el producto y solución con el producto.

Teniendo en cuenta los resultados reflejados en la gráfica en cuanto al Punto de Equilibrio queda demostrada la factibilidad del sistema evidenciado por la relación entre la complejidad del problema (cantidad de variables) y el tiempo que demora la solución del mismo de forma manual y automatizada.

Conclusiones Parciales

En este capítulo se realizó el estudio de factibilidad del proyecto mediante la Metodología Costo-Efectividad (Beneficio). En este se analizaron los efectos económicos, los beneficios y costos intangibles, y además se calculó el costo de ejecución del proyecto mediante la ficha de costo, la cual arrojó como resultado \$25.00 CUC y \$100,00 CUP por mes, demostrándose así la factibilidad del proyecto.



Conclusiones Generales

Con el desarrollo de la aplicación Web para la gestión de la información sobre derrames de hidrocarburos en el puerto de Moa se dio cumplimiento a los objetivos propuestos en esta investigación, pues se obtuvo como resultado un sistema informático en el que se aplican los resultados de la investigación realizada arribándose a las siguientes conclusiones:

- Se encontraron dos sistemas que gestionan la información sobre derrames de hidrocarburos y ninguno se puede utilizar en Cuba debido a que no tenemos acceso a los servidores a los que estos se conectan.
- Se seleccionaron las herramientas más adecuadas para el desarrollo de la aplicación web: como framework de desarrollo se utilizaron Ext JS y OpenLayers, como gestor de base de datos se utilizó PostgreSQL, como servidor de mapas se empleó Geoserver, como lenguajes de programación se emplearon PHP, HTML y JavaScript.
- Durante el desarrollo de la metodología XP se generaron 10 historias de usuario, una lista de reserva de 27 requisitos funcionales, 10 tarjetas CRC, 5 pruebas de aceptación y 27 tareas de ingeniería.
- Se realizó el estudio de factibilidad del proyecto mediante la Metodología Costo-Efectividad (Beneficio). En este se analizaron los efectos económicos, los beneficios y costos intangibles, y además se calculó el costo de ejecución del proyecto mediante la ficha de costo, la cual arrojó como resultado \$ 25.00 CUC y \$ 100,00 CUP por mes, demostrándose así la factibilidad del proyecto.

Como resultado de este trabajo se logró desarrollar una aplicación web que da cumplimiento a las especificidades de los objetivos propuestos, agilizando la gestión de la información y garantizando la disponibilidad y fiabilidad de los reportes e informes exigidos por el Ministerio de Energía y Minas.



Recomendaciones

- Poner en explotación el software en el puerto de Moa Comandante Raúl Días Arguelles.
- Agregar funcionalidad que le permita a los usuarios del sistema saber las condiciones meteorológicas.
- Realizar un estudio más profundo de este sistema en vista a perfeccionarlo en versiones futuras.
- Ejecutar la aplicación preferentemente en los navegadores Mozilla Firefox y Google Chrome ya que traducen los estilos de diseño con mayor claridad.

Con la puesta en práctica de las recomendaciones se obtendrá una aplicación con mayores funcionalidades. Esto permitirá al especialista del medio ambiente en el puerto de Moa contar con una herramienta informática que agilice la gestión de información sobre derrames de hidrocarburo en el puerto de Moa.



Referencias Bibliográficas

1-ASA | Applied Science Associates (En línea) (Citado: febrero 13,2013)

<http://www.asascience.com/>

2- CALA HINOJOSA, Alexei. Sistema de gestión energética asistida para el ISMMM-Módulo de control del consumo energético. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2011.

3-¿Qué es un servidor web? (En línea) (Citado: febrero 13,2013)

<http://www.misrespuestas.com/que-es-un-servidor-web.html>

4- Servidor httpd (Apache) (En línea) (Citado: febrero 12,2013)

<http://www.profesionalhosting.com/servidores-dedicados/definicion/servidor-httpdapache-70.html>

5-1C_C15010: Características Principales (En línea) (Citado: febrero 12,2013)

<http://ocw.uniovi.es/mod/resource/view.php?id=1242>

6- HERNÁN RUIZ, Marcelo. Programación Web Avanzada [Libro]. La Habana, Cuba. Editorial Félix Varela, 2006.

7- Diseño web y multimedia (En línea) (Citado el 13/2/2013)

<http://blogs.utpl.edu.ec/disenowebymultimedia/2009/07/23/ventajas-y-desventajas-de-php-2/>

8-¿Qué es JavaScript? (En línea) (Citado el 13/2/2013)

<http://www.pablin.com.ar/computer/cursos/cursojs/js1.htm>

9- aC Tema 01 HTML (En línea) (Citado el 13/2/2013)

http://www.aulaclie.es/html/t_1_1.htm

10- Jorge Naula ¿Qué es un framework de desarrollo web?

(En línea) (Citado el 13/2/2013)

<http://kokiing.blogspot.com/2010/12/que-es-un-framework-de-desarrollo-web.html>

11- ¿Qué es Ext JS? (En línea) (Citado el 13/2/2013)

<http://www.extjs.mx/2011/11/27/post-con-cursos-generales/>

12- Ext JS lo bueno, lo malo y lo feo |Desarrollo en Web (En línea) (Citado: febrero 13,2013)

<http://blogs.antartec.com/desarrolloweb/2008/10/extjs-lo-bueno-lo-malo-y-lo-feo/>

13- OpenGeo: OpenLayers (En línea) (Citado: febrero 17,2013)

<http://opengeo.org/technology/openlayers/>



14-¿Qué es un Sistema Gestor de Bases de Datos o SGBD? (En línea)
(Citado el 16/2/2013)

<http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd>

15- XP-Extreme Programing Ingeniería de Software (En línea) (Citado el 18/2/2013)

http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html

16- Metodología XP Vs. Metodología Rup. (En línea) (Citado el 16/2/2013)

<http://metodologiaxpvsmetodologiarup.blogspot.com/>

17- Fases de la Programación Extrema (En línea) (Citado el 16/2/2013)

<http://programacionextrema.tripod.com/fases.htm>

18- ¿Qué es un SIG? | Sistemas de Información Geográfica (En línea)
(Citado el 16/2/2013)

<http://langleruben.wordpress.com/%C2%BFque-es-un-sig/>(En línea) (Citado el 16/2/2013)

19- Geoserver –OSGeo-Live 6.5 Documentation (En línea) (Consultado el 14 febrero del 2013)

http://live.osgeo.org/es/overview/geoserver_overview.html

20- Xampp, el servidor web listo para ser usado (En línea) (Consultado el 14 febrero del 2013)

<http://www.aplicacionesempresariales.com/xampp-el-servidor-web-listo-para-ser-usado.html>

21- Garcés Valdés Alexander, Sistema de Gestión Académica Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2012.

22- Rodríguez Pérez, Agustín, Sistema para la Gestión del Presupuesto en el ISMM, Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2012.

23- Embarcadero ER/Studio, Software – Herramientas de Desarrollo de Software [En línea] [Consultado el 2 de mayo del 2013] Disponible en:

<http://www.guiadesolucionestic.com/software-del-sistema/herramientas-de-desarrollo/herramientas-de-desarrollo-de-software/972-embarcadero-erstudio>



Bibliografía

- ✓ Erik Hazzard: OpenLayers 2.10 Beginners Guide.
- ✓ Antonio Santiago Pérez: OpenLayers. Cookbook.2012.
- ✓ <http://www.quizzpot.com/>
- ✓ <http://www.forosdelweb.com/>
- ✓ <http://www.geoserver.org/>
- ✓ Cuaresma S. Metodología de Desarrollo
<http://www.marblestation.com>
- ✓ Figueroa R. G, Solís C. J, Cabrera A. A. Metodologías tradicionales vs. metodologías ágiles. Disponible en:
<http://adonisnet.files.wordpress.com>
- ✓ Denzer P (2002). PostgreSQL.
<http://es.scribd.com/doc/36596149/Postgres-informe>
- ✓ Achour Mehdi y otros (2005). Manual de PHP:
<http://php.net/manual/es/>
- ✓ Dondo A. PHP en castellano (2005) . ¿Por qué elegir PHP?
<http://www.programacion.com/php/articulo/porquephp/>



Glosario de Términos

Cliente: Persona, organización o grupo de personas que solicita la construcción de un sistema, ya sea empezando desde cero, o mediante el refinamiento de versiones sucesivas.

Glosario de términos: Términos comunes que se utilizan para describir el sistema.

HTML: Hyper Text Markup Language, o simplemente HTML, es un lenguaje de etiquetas de programación.

Interfaz: Conjunto de representaciones de operaciones públicas.

Metodología ágil: Nuevo enfoque metodológico orientado a la gente y los resultados.

Metodologías de desarrollo de software: Son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software.

Herramientas: Son los ambientes de apoyo necesario para automatizar las prácticas de Ingeniería de Software.

Servidor: Computadora central de un sistema de red que provee servicios y recursos (programas, comunicaciones, archivos, etc.) a otras computadoras (clientes) conectadas a ella.

Proyecto: Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

Requisitos: Son las funciones, servicios y restricciones operativas del sistema.

Iteraciones: En el contexto de un proyecto se refieren a la técnica de desarrollar y entregar componentes incrementales de funcionalidades de un negocio. Una iteración resulta en uno o más paquetes atómicos y completos del trabajo del proyecto que pueda realizar alguna función tangible del negocio. Múltiples iteraciones contribuyen a crear un producto completamente integrado.



Glosario de Términos

Procedimiento: Son los mecanismos de gestión que soportan a los métodos: el control de los proyectos y el control de la calidad.

Sistema: Delimita el mundo sobre el cual se está construyendo el modelo.

Software: Se refiere a los programas y datos almacenados en un ordenador.

Usuario: Persona que interactúa con el sistema.

Validación: No es más que verificar que un producto determinado cumple con los requisitos que fueron pactados con el cliente.

**Anexos****Anexos 1 Historia de Usuarios**

Historias de Usuario que se abordan en el dominio del sistema que se desea automatizar.

HU No.1 Gestionar Entidad Responsable

Historia de Usuario	
Número: 1	Usuarios: Especialista
Nombre: Gestionar Entidad Responsable	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Aurelio Calvo Burke	
Descripción: El especialista debe insertar, modificar y mostrar las entidades responsables	
Observaciones: Confirmado por el cliente.	

HU No.2 Gestionar Derrame

Historia de Usuario	
Número: 2	Usuarios: Especialista
Nombre: Gestionar Derrame	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Aurelio Calvo Burke	
Descripción: El especialista debe insertar, modificar, eliminar y mostrar los derrames ocurridos.	
Observaciones: Confirmado por el cliente.	



HU No.3 Gestionar Reporte de Derrame

Historia de Usuario	
Número: 3	Usuarios: Especialista
Nombre: Gestionar Reporte de Derrame	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Aurelio Calvo Burke	
Descripción: El especialista debe insertar, modificar, eliminar y mostrar los reportes de derrames ocurridos.	
Observaciones: Confirmado por el cliente.	

HU No.4 Gestionar Reporte de Incidente

Historia de Usuario	
Número: 4	Usuarios: Especialista
Nombre: Gestionar Reporte de Incidente	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Aurelio Calvo Burke	
Descripción: El especialista debe insertar, modificar, eliminar y mostrar los reportes de incidentes ocurridos.	
Observaciones: Confirmado por el cliente.	



HU No.5 Gestionar Informe Final

Historia de Usuario	
Número: 5	Usuarios: Especialista
Nombre: Gestionar Informe Final	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Aurelio Calvo Burke	
Descripción: El especialista debe insertar, modificar, eliminar y mostrar los informes redactados.	
Observaciones: Confirmado por el cliente.	

HU No.6 Gestionar Cuentas de Usuario

Historia de Usuario	
Número: 6	Usuarios: Especialista
Nombre: Gestionar Cuentas de Usuario	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Aurelio Calvo Burke	
Descripción: El administrador debe crear, modificar, eliminar y mostrar los usuarios del sistema.	
Observaciones: Confirmado por el cliente.	



HU No.7 Autenticar Usuario

Historia de Usuario	
Número: 7	Usuarios: Administrador ,Especialista y Director.
Nombre: Autenticar usuarios	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Aurelio Calvo Burke	
Descripción: Los usuarios del Sistema ingresan sus datos para entrar al sistema (nombre de usuario y contraseña). El sistema verifica que los datos estén correctos, en caso de que no sean correctos los mismos la aplicación muestra un mensaje de error.	
Observaciones: Confirmado por el cliente.	

HU No.8 Mostrar derrame de hidrocarburo en un mapa.

Historia de Usuario	
Número: 8	Usuarios: Especialista y Director.
Nombre: Mostrar derrame de hidrocarburos en un mapa	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Aurelio Calvo Burke	
Descripción: El sistema debe mostrar en un mapa los derrames de hidrocarburos ocurridos.	
Observaciones: Confirmado por el cliente.	



HU No.9 Graficar cantidad de derrames ocurridos por año

Historia de Usuario	
Número: 9	Usuarios: Especialista y Director.
Nombre: Graficar cantidad de derrames de hidrocarburos ocurridos en un año.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Aurelio Calvo Burke	
Descripción: El sistema debe mostrar una gráfica con los derrames de hidrocarburos ocurridos en un año.	
Observaciones: Confirmado por el cliente.	

HU No.10 Graficar volumen de hidrocarburo derramado en un año.

Historia de Usuario	
Número: 10	Usuarios: Especialista y Director.
Nombre: Graficar cantidad de derrames de hidrocarburos ocurridos en un año.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Aurelio Calvo Burke	
Descripción: El sistema debe mostrar una gráfica con el volumen de hidrocarburos derramados en un año.	
Observaciones: Confirmado por el cliente.	



Anexos 2 Tarjetas CRC

Tarjeta CRC No.1 Entidad Responsable

Clase: entidad_responsable	
Description: Guarda los datos de la entidad responsable	
Attributes:	
Name	Description
id_entidad	identificador de entidad responsable
nombre	
direccion	
Responsibilities:	
Name	Collaborator
Insertar entidad responsable	
Modificar entidad responsable	
Mostrar entidad responsable	

Tarjeta CRC No.2 Derrame

Clase:Derrame	
Description: Guarda los datos de los derrames	
Attributes:	
Name	Description
dhc	identificador del derrame
osde	
ocurrio	
fecha_detec	
hora_detec	
origen	
causa	
tipo_hidrocarburo	
riesgo_incendio	
riesgo_incendio	
observador_inicial	
especialista_informa	
Responsibilities:	
Name	Collaborator
Insertar Derrame	entidad_responsable
Modificar Derrame	clasificacion
Mostrar Derrame	
Eliminar Derrame	



Tarjeta CRC No.3 Ocurrió

Clase: ocurrio	
Description: contiene donde ocurrio el derrame	
Attributes:	
Name	Description
id	identificador de donde ocurrio
ocurrio	
Responsibilities:	
Name	Collaborator
Muestra donde ocurrio el derrame	

Tarjeta CRC No.4 Reporte de Derrame

Clase: Reporte de Derrame	
Description: Guarda los datos de los reportes de derrame	
Attributes:	
Name	Description
id_reporte	identificador de los reporte de derrame
parte	
volumen_ad	
area_ce	
manchas_hidrocarburo	
volumen_pr	
derrames_adic	
areas_cercanas	
condiciones_climaticas	
accidentes	
medidas_control	
notificaciones	
observaciones	
Responsibilities:	
Name	Collaborator
Insertar Reporte de Derrame	dhc
Modificar Reporte de Derrame	
Mostrar Reporte de Derrame	
Eliminar Reporte de Derrame	

**Tarjeta CRC No.5 Reporte de Incidente**

Clase:Reporte de Incidente	
Description: Guarda los datos de los reportes de incidentes	
Attributes:	
Name	Description
incidente	identificador de los reportes de incidente
empresa	
fecha	
hora	
area_dpto	
nombre	
apellidos	
cargo	
descripcion	
fecha_ocurrencia	
medidas	
perdidas	
reporte_por	
especialista_informa	
Responsibilities:	
Name	Collaborator
Insertar Reporte de Incidente	
Modificar Reporte de Incidente	
Mostrar Reporte de Incidente	
Eliminar Reporte de Incidente	



Tarjeta CRC No.6 Informe Final.

Clase: informe_final	
Description: Guarda los datos del informe final	
Attributes:	
Name	Description
id_informe	identificador del informe final
fecha_cierre	
hora_cierre	
volumen_derram	
area_contam	
ocurrio_incendio	
causa	
consecuencias	
hidroc_recuperado	
dannos	
lugares_afectados	
informe_recup	
gastos	
revisión	
acciones	
destino_des	
informacion_adic	
fecha_redac	
Responsibilities:	
Name	Collaborator
Insertar informe final	dhc
Modificar informe final	
Mostrar informe final	
Eliminar informe final	

Tarjeta CRC No.7 Usuario.

Clase: Usuario	
Description: Guarda los datos de los usuarios	
Attributes:	
Name	Description
usuario	identificador de usuario
pass	
nombre	
apellidos	
cargo	
Responsibilities:	
Name	Collaborator
Insertar Usuario	
Modificar Usuario	
Mostrar Usuario	
Eliminar Usuario	

**Tarjeta CRC No.8 Artefacto Naval**

Clase: artefacto_naval	
Description: Guarda los datos del artefacto naval	
Attributes:	
Name	Description
tipo	
bandera	
matricula	
nombre_buque	
Responsibilities:	
Name	Collaborator
Insertar artefacto naval	id_entidad
Modificar artefacto naval	
Mostrar artefacto naval	

Tarjeta CRC No.9 UEB.

Clase: ueb	
Description: Guarda los datos de la ueb	
Attributes:	
Name	Description
empresa	
Responsibilities:	
Name	Collaborator
Inserta ueb	id_entidad
Modificar ueb	
Mostrar ueb	



Tarjeta CRC No.10 Clasificación del derrame

Clase: clasificacion_derrame	
Description: guarda la clasificación de los derrames	
Atributes:	
Name	Description
id	identificador de la clasificacion del derrame
clasificacion	
Responsibilities:	
Name	Collaborator
Muestra la clasificación del derrame	

Anexos 3 Tareas de Ingeniería

Tarea No.1 Insertar Entidad Responsable.

Tarea de Ingeniería	
Numero tarea: 1	Numero de historia: 1
Nombre tarea: Insertar Entidad Responsable.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 22/2/2013	Fecha de fin: 23/2/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará insertar una entidad responsable	

Tarea No.2 Modificar Entidad Responsable.

Tarea de Ingeniería	
Numero tarea: 2	Numero de historia: 1
Nombre tarea: Modificar Entidad Responsable	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 24/2/2013	Fecha de fin: 27/2/2013



Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará modificar una entidad responsable	

Tarea No 3 Mostar Entidad Responsable

Tarea de Ingeniería	
Numero tarea: 3	Numero de historia: 1
Nombre tarea: Mostrar Entidad Responsable.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha de inicio: 28/2/2013	Fecha de fin: 28/2/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará mostrar las entidades responsable	

Tarea No 4 Insertar Derrame de Hidrocarburo

Tarea de Ingeniería	
Numero tarea: 4	Numero de historia: 2
Nombre tarea: Insertar Derrame de hidrocarburo	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 1//3/2013	Fecha de fin: 2/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará insertar un derrame de hidrocarburo.	



Tarea No 5 Modificar Derrame de Hidrocarburo

Tarea de Ingeniería	
Numero tarea: 5	Numero de historia: 2
Nombre tarea: Modificar Derrame de hidrocarburo	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 3/3/2013	Fecha de fin: 4/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará modificar un derrame de hidrocarburo.	

Tarea No 6 Mostrar Derrame de Hidrocarburo

Tarea de Ingeniería	
Numero tarea: 6	Numero de historia: 2
Nombre tarea: Mostrar Derrame de hidrocarburo	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 5/3/2013	Fecha de fin: 6/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará mostrar los derrames de hidrocarburo ocurridos.	

Tarea No 7 Eliminar Derrame de Hidrocarburo

Tarea de Ingeniería	
Numero tarea: 7	Numero de historia: 2
Nombre tarea: Eliminar Derrame de hidrocarburo	
Tipo de tarea: Desarrollo	Puntos estimados: 1



Fecha de inicio: 7/3/2013	Fecha de fin: 7/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará eliminar un derrame de hidrocarburo.	

Tarea No 8 Insertar Reporte de Derrame

Tarea de Ingeniería	
Numero tarea: 8	Numero de historia: 3
Nombre tarea: Insertar Reporte Derrame	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 8/3/2013	Fecha de fin: 9/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará insertar un reporte de derrame.	

Tarea No 9 Modificar Reporte de Derrame

Tarea de Ingeniería	
Numero tarea: 9	Numero de historia: 3
Nombre tarea: Modificar Reporte de Derrame	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 10/3/2013	Fecha de fin: 11/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará modificar un reporte de derrame.	

**Tarea No 10 Mostrar Reportes de Derrame**

Tarea de Ingeniería	
Numero tarea: 10	Numero de historia: 3
Nombre tarea: Mostrar los Reportes de Derrame	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 12/3/2013	Fecha de fin: 13/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará mostrar los reportes de derrames.	

Tarea No 11 Eliminar Reporte de Derrame

Tarea de Ingeniería	
Numero tarea: 11	Numero de historia: 3
Nombre tarea: Eliminar Reporte de Derrame	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 14/3/2013	Fecha de fin: 14/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará eliminar reporte de derrame.	

Tarea No 12 Insertar Reporte de Incidente

Tarea de Ingeniería	
Numero tarea: 12	Numero de historia: 4
Nombre tarea: Insertar Reporte de Incidente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 15/3/2013	Fecha de fin: 16/3/2013



Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará insertar un reporte de incidente.	

Tarea No 13 Modificar Reporte de Incidente.

Tarea de Ingeniería	
Numero tarea: 13	Numero de historia: 4
Nombre tarea: Modificar Reporte de Incidente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 17/3/2013	Fecha de fin: 18/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará modificar un reporte de incidente.	

Tarea No 14 Mostrar Reporte de Incidente.

Tarea de Ingeniería	
Numero tarea: 14	Numero de historia: 4
Nombre tarea: Mostrar Reporte de Incidente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 19/3/2013	Fecha de fin: 20/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará mostrar los reportes de incidentes.	



Tarea de Ingeniería	
Numero tarea: 15	Numero de historia: 4
Nombre tarea: Eliminar Reporte de Incidente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 21/3/2013	Fecha de fin: 21/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará eliminar un reporte de incidente.	

Tarea No 15 Eliminar Reporte de Incidente.**Tarea No 16 Insertar Informe Final.**

Tarea de Ingeniería	
Numero tarea: 16	Numero de historia: 5
Nombre tarea: Insertar Informe Final	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 22/3/2013	Fecha de fin: 23/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará insertar un informe final.	

Tarea No 17 Modificar Informe Final.

Tarea de Ingeniería	
Numero tarea: 17	Numero de historia: 5
Nombre tarea: Modificar Informe Final	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 24/3/2013	Fecha de fin: 25/3/2013



Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará modificar un informe final	

Tarea No 18 Mostrar Informe Final.

Tarea de Ingeniería	
Numero tarea: 18	Numero de historia: 5
Nombre tarea: Mostrar Informe Final	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 26/3/2013	Fecha de fin: 27/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará mostrar los informes finales.	

Tarea No 19 Eliminar Informe Final

Tarea de Ingeniería	
Numero tarea: 19	Numero de historia: 5
Nombre tarea: Eliminar Informe Final	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 28/3/2013	Fecha de fin: 28/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el especialista y facilitará eliminar un informe final	

Tarea No 20 Insertar Usuario

Tarea de Ingeniería	
Numero tarea: 20	Numero de historia: 6
Nombre tarea: Insertar Usuario	



Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 29/3/2013	Fecha de fin: 30/3/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el administrador y facilitará insertar un usuario.	

Tarea No 21 Modificar Usuario

Tarea de Ingeniería	
Numero tarea: 21	Numero de historia: 6
Nombre tarea: Modificar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 31/3/2013	Fecha de fin: 1/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el administrador y facilitará modificar un usuario.	

Tarea No 22 Mostrar Usuario

Tarea de Ingeniería	
Numero tarea: 22	Numero de historia: 6
Nombre tarea: Mostrar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 2/4/2013	Fecha de fin: 3/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el administrador y facilitará mostrar los usuarios.	



Tarea de Ingeniería	
Numero tarea: 23	Numero de historia: 6
Nombre tarea: Eliminar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 4/4/2013	Fecha de fin: 5/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el administrador y facilitará eliminar un usuario.	

Tarea No 23 Eliminar Usuario

Tarea No 24 Autenticar Usuario

Tarea de Ingeniería	
Numero tarea: 24	Numero de historia: 7
Nombre tarea: Autenticar Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 6/4/2013	Fecha de fin: 12/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por los usuarios del sistema y les permite acceder a la aplicación.	

Tarea No.25 Mostrar Derrame en un mapa.

Tarea de Ingeniería	
Numero tarea: 25	Numero de historia: 8
Nombre tarea: Mostrar derrames en un mapa	



Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha de inicio: 13/4/2013	Fecha de fin: 26/4/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el sistema y facilitará mostrar los derrames ocurridos en un mapa.	

Tarea No.26 Graficar cantidad de derrames de hidrocarburos ocurridos en un año.

Tarea de Ingeniería	
Numero tarea: 26	Numero de historia: 9
Nombre tarea: Graficar cantidad de derrames de hidrocarburos ocurridos en un año	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 27/4/2013	Fecha de fin: 10/5/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el sistema y facilitará mostrar en una gráfica la cantidad de derrames ocurridos en un año.	

Tarea No.27 Graficar cantidad de derrames de hidrocarburos ocurridos en un año.

Tarea de Ingeniería	
Numero tarea: 27	Numero de historia: 10
Nombre tarea: Graficar volumen de hidrocarburo derramado en un año.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha de inicio: 11/5/2013	Fecha de fin: 17/5/2013
Responsable: Aurelio Calvo Burke	
Descripción: Esta tarea es realizada por el sistema y facilitará mostrar en una gráfica el volumen de hidrocarburo derramado en un año.	



Anexo 4 Pruebas de Aceptación.**Prueba de Aceptación para la HU: Gestionar Cuenta de Usuario.**

Prueba de Aceptación
HU: Gestionar Cuenta de Usuario.
Nombre: Prueba para comprobar la Gestión de Usuario.
Descripción: Validación del Gestionar Cuenta de Usuario.
Condiciones de ejecución: El administrador debe entrar a la aplicación para poder crear, modificar, mostrar y eliminar las cuentas de usuarios existentes.
Entrada / Pasos de ejecución: El administrador escribe el nombre de los usuarios con sus datos para insertar, luego presiona el botón insertar. Después de haber insertado los usuarios es que se pueden modificar sus datos, eliminarlo o mostrarlos en un listado.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrán ver en un listado de usuarios registrados.• Cuando se modifican los usuarios se podrán ver en un listado de usuarios con sus datos modificados.• Cuando se elimina un usuario, se borra del listado de usuarios.
Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar el nuevo usuario.• El usuario ya existe en la base datos.
Evaluación de la prueba: Aceptada

**Prueba de Aceptación para la HU: Gestionar Derrame de Hidrocarburo.**

Prueba de Aceptación
HU: Gestionar Derrame.
Nombre: Prueba para comprobar la Gestión de Derrame.
Descripción: Validación del Gestionar Derrame.
Condiciones de ejecución: El especialista debe entrar a la aplicación para poder insertar, modificar, mostrar y eliminar los derrames de hidrocarburos ocurridos.
Entrada / Pasos de ejecución: El especialista escribe los datos del derrame, luego presiona el botón insertar. Luego de haber insertado los derrames, es que se pueden modificar sus datos, eliminarlo o mostrarlos en un listado.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrán ver en un listado de derrames de hidrocarburos ocurridos.• Cuando se modifican los derrames se podrán ver en un listado de derrames de hidrocarburos con sus datos modificados.• Cuando se elimina un derrame de hidrocarburo, se borra del listado de derrames.
Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar un nuevo derrame.• El derrame ya está registrado en la base datos.
Evaluación de la prueba: Aceptada



Prueba de Aceptación para la HU: Gestionar Reporte de Derrame.

Prueba de Aceptación
HU: Gestionar Reporte de Derrame.
Nombre: Prueba para comprobar la Gestión de Reporte de Derrame.
Descripción: Validación del Gestionar Reporte de Derrame.
Condiciones de ejecución: El especialista debe entrar a la aplicación para poder insertar, modificar, mostrar y eliminar los reportes de derrames de hidrocarburos ocurridos.
Entrada / Pasos de ejecución: El especialista escribe los datos del reporte de derrame, luego presiona el botón insertar. Después de haber insertado los reportes de derrames es que se pueden modificar sus datos, eliminarlo o mostrarlos en un listado.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrán ver en un listado de reportes de derrames de hidrocarburos.• Cuando se modifican los reportes de derrames se podrán ver en un listado de reportes de derrames de hidrocarburos con sus datos modificados.• Cuando se elimina un reporte de derrame de hidrocarburo, se borra del listado de reportes de derrames.
Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar un nuevo reporte de derrame.• El reporte de derrame ya está registrado en la base datos.
Evaluación de la prueba: Aceptada

**Prueba de Aceptación para la HU: Gestionar Reporte de Incidente.**

Prueba de Aceptación
HU: Gestionar Reporte de Incidente.
Nombre: Prueba para comprobar la Gestión de Reporte de Incidente.
Descripción: Validación del Gestionar Reporte de Incidente.
Condiciones de ejecución: El especialista debe entrar a la aplicación para poder insertar, modificar, mostrar y eliminar los reportes de incidentes.
Entrada / Pasos de ejecución: El especialista escribe los datos del reporte de incidente, luego presiona el botón insertar. Después de haber insertado los reportes de incidentes es que se pueden modificar sus datos, eliminarlo o mostrarlos en un listado.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrán ver en un listado de reportes de incidentes.• Cuando se modifican los reportes de incidentes se podrán ver en un listado de reportes de incidentes con sus datos modificados.• Cuando se elimina un reporte de incidente, se borra del listado de reportes de incidentes.
Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar un nuevo reporte de incidente.• El reporte de incidente ya está registrado en la base datos.
Evaluación de la prueba: Aceptada

**Prueba de Aceptación para la HU: Gestionar Informe Final.**

Prueba de Aceptación
HU: Gestionar Informe Final.
Nombre: Prueba para comprobar la Gestión del Informe Final.
Descripción: Validación del Gestionar Informe Final.
Condiciones de ejecución: El especialista debe entrar a la aplicación para poder insertar, modificar, mostrar y eliminar el informe final.
Entrada / Pasos de ejecución: El especialista escribe los datos del informe final, luego presiona el botón insertar. Después de haber insertado el informe final es que se pueden modificar sus datos, eliminarlo o mostrarlos en un listado.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrá ver en un listado los informes finales.• Cuando se modifican los informes finales se podrán ver en un listado de informes finales con sus datos modificados.• Cuando se elimina un informe final, se borra del listado de informes finales.
Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar un nuevo informe final.• El informe final ya está registrado en la base datos.
Evaluación de la prueba: Aceptada



Prueba de Aceptación para la HU: Mostrar derrames de hidrocarburo en un mapa.

Prueba de Aceptación
HU: Mostrar derrames de hidrocarburo en un mapa.
Nombre: Prueba para comprobar el Mostrar derrames de hidrocarburo en un mapa.
Descripción: Validación del Mostrar derrames de hidrocarburo en un mapa..
Condiciones de ejecución: El especialista debe entrar a la aplicación para poder insertar los datos del derrame de hidrocarburo.
Entrada / Pasos de ejecución: El especialista escribe los datos del derrame de hidrocarburo, luego presiona el botón insertar. Después de haber insertado los datos del derrame es que se puede ver la ubicación del derrame en un mapa.
Resultado esperado: <ul style="list-style-type: none">• Si se insertaron los datos correctamente se podrá ver en un mapa la ubicación del derrame de hidrocarburo.• Cuando se elimina un derrame de hidrocarburo, este derrame ya no aparecerá ubicado en el mapa. Se produce un error en caso de que: <ul style="list-style-type: none">• Existan campos vacíos a la hora de insertar un nuevo informe final.• El informe final ya está registrado en la base datos.
Evaluación de la prueba: Aceptada