



**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO
EN CIENCIAS INFORMÁTICAS.**

**Sistema de gestión de información de los
procesos ambientales en el Centro de Inspección
y Control Ambiental.**

AUTORES

Deilys Lázara Ulloa Martínez
Darianna Dominico Brito

TUTORES

Ing. Arcadio Abad Márquez
Ing. Kenier Cumbá Bueno

CO-TUTOR

MSc. Roberto López Dosagües

La Habana, junio 2013 "Año 55 de la Revolución"



“He tomado sobre mis espaldas el monopolio de mejorar sólo a una persona, esa persona soy yo mismo y sé, cuán difícil es conseguirlo.”

Mahatma Gandhi



DECLARACIÓN DE AUTORÍA:

Por este medio declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que hagan el uso que estimen pertinente con este trabajo.

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

Firma del Autor
Darianna Dominico Brito

Firma del Autor
Deilys Lázara Ulloa Martínez

Firma del Tutor
Ing. Arcadio Abad Márquez

Firma del Tutor
Ing. Kenier Cumbá Bueno

Firma del Co-Tutor
M.Sc. Roberto López Desagüe



Agradezco:

A mi familia por apoyarme en mis decisiones y por convertirme en la persona que soy hoy en día, sin su cariño, sus consejos y enseñanzas no estaría en estos momentos graduándome de Ingeniera en Ciencias Informáticas (UCI), en especial a mi bisabuela Blanca por su amor incondicional y aceptarme como una hija más.

A mi mamá por siempre estar ahí ayudándome cuando la he necesitado y por ser mi inspiración y modelo a seguir al demostrarme que con esfuerzo todo es posible solo hay que proponérselo y sacrificarse para salir adelante y tener un futuro mejor, a mi abuela Tania, que se ha esforzado por criarme y llenar el vacío que dejó de mi bisabuela al morir y por sobre todo por aceptarme tal cual soy. A mis tíos Yoly y Lule que los considero más que tíos padres ya que siempre están ahí para mí, a mis primas Claudia y Dalia, a mis hermanos Darién, Orlando Javier y Pedro Dariel que son los mejores hermanos del mundo y no los cambiaría por nada.

A Yenisey, Yanet Silva, Yanet García y Zuleydis que más que amigas se han convertido en mis confidentes. Gracias por siempre estar ahí para mí y ser tan buenas amigas, uds. para mí son parte de mi familia.

A mis tutores Arcadio, Kenier y Dosagüe, de veras que no existen palabras en este mundo para expresar lo agradecida que estoy que hayan sido mis tutores, gracias por toda la ayuda, dedicación y entrega que han tenido, de corazón esta tesis no es mía y de Deilys sino de uds. sin vosotros nada de esto hubiese sido posible.

A mi compañera de tesis Deilys que sin su ayuda no estaríamos aquí graduándonos y si tuviese que escoger de nuevo un compañero de tesis sin pensarlo ella sería la indicada.

A mis amistades de la UCI Cuni, Mara, Virgilio, Osiel, Odesney, Odalis, Piti, Yayi, Amado, Roney, Oscar, a mis compañeros de aula, entre otros. Gracias por los momentos inolvidables que pasamos siempre estarán en mi corazón.

A mi novio Alejandro Miguel Pedraza por ser mi apoyo, mi fuerza y mi familia en estos últimos 4 años de la carrera, sin él la universidad sin duda habría sido un poco compleja, pero gracias a él todo fue más fácil. Gracias por siempre estar a mi lado, apoyarme en todas mis decisiones y convertirme en mi familia.

Darianna



Agradezco:

A mis padres por darme la fuerza y el apoyo que siempre necesite para seguir esta batalla hasta el final, de verdad que sin ustedes no hubiera llegado hasta donde estoy, gracias por ser mi sustento en los momentos difíciles y seguir a mi lado pese a cualquier decisión que yo tome en la vida.

A mis abuelos, Silvia, Zaida y Pablo sin ustedes, la voz de la experiencia, me hubiera sido imposible seguir adelante, gracias por sus consejos y por guiarme en todo momento por el camino correcto a pesar de los obstáculos que encuentre.

A mi familia en general, tíos, primos, hermanos, gracias por estar de una forma u otra a mi lado, dándome aliento y deseando que pronto hubiera una ingeniera en la familia que fuera el ejemplo de los que siguen.

A mis amistades de la universidad, los que están y los que se han ido, Cuni, Mara, Lirisandra, Yanet, Darianna, Aimeé, Jessica por ser como unas hermanas para mí todo este tiempo. A Sandy, Andres, Alejandro, Rogney, Yusdel, a todos los compañeros de aula que he tenido a lo largo de estos años de universidad, al equipo de softboll, kikimboll y Voly, gracias por esos momentos tan divertidos que me hicieron pasar y que son realmente inolvidables.

A mi compañera de tesis, Darianna, porque sin ella esta tesis no hubiera alcanzado su fin, gracias por estar siempre al pendiente y si a alguien le debo el estar aquí hoy, ese alguien eres tú.

A mi pareja por estar todo este tiempo a mi lado apoyándome, muchas veces escuchándome solo hablar de programación y de tesis, de verdad que solo tú aguantas eso, gracias por darme las fuerzas que siempre necesite para estar tan lejos y también por ayudarme a superar barreras realmente altas, sin ti no hubiera logrado ni alcanzarlas con la vista.

A mis tutores Arcadio, Kenier y Dosagües, esta tesis más que de Darianna y mía es de ustedes, gracias por ayudarnos en cada momento que los necesitamos, por estar presente cada vez que los buscábamos y ayudarnos sin poner un pero.

Deilys



Darianna:

A la persona más importante en mi vida, a mi bisabuela, por ser mi motor impulsor y la fuerza necesaria para seguir adelante cuando pensé que ya no podía más, si hoy estoy graduándome es todo por pensar en lo orgullosa que estaría de mí si me estuviese viendo.

A mis hermanos pequeños para que les sirva de ejemplo e inspiración y para que tengan presente que si se esfuerzan pueden llegar lejos y lograr sus sueños.

Deilys:

A mi abuela Zaida porque ella es el motor impulsor que me ubicó donde estoy ahora y mi mayor deseo es que este orgullosa de mí.

A mis padres que han dejado sus huellas bien marcadas en la vida para que yo las pueda seguir con claridad y que me han servido de guía para llegar a la cúspide de mi carrera y de mi vida.

A mi sobrina para que siga el ejemplo y vea que en la vida sin los estudios realizados completamente, no somos nada.



Por la necesidad de mejorar el control y manejo de la información por los expertos en el Centro de Inspección y Control Ambiental (CICA) mediante el uso de las TIC, se realiza un estudio para seleccionar las herramientas informáticas más factibles, teniendo en cuenta las tendencias del desarrollo web, se analizan algunos de los sistemas similares definiendo posibles funcionalidades para la solución propuesta, agregándole las necesidades del cliente. El desarrollo se divide en las fases propuestas por la metodología seleccionada, documentando cada paso y cada artefacto generado. Para verificar el cumplimiento de las funcionalidades propuestas se realizan las pruebas de aceptación. Como resultado se obtiene un producto funcional con una documentación que sirve de base para futuras investigaciones o modificaciones a la propuesta de solución. El resultado de la investigación deja constancia documental de la metodología empleada, que permitió cumplir el objetivo general propuesto y la satisfacción de las necesidades del cliente.



Introducción	1
Capítulo 1: Fundamentación teórica	6
1.1. Marco conceptual.....	6
1.1.1. Gestión de la información	6
1.1.2. Inspección Ambiental Estatal	7
1.1.2.1. Objetivos de la IAE	7
1.1.3. Licencias Ambientales	7
1.1.4. Evaluación de Impacto Ambiental	8
1.1.5. Contravenciones	9
1.1.6. Control ambiental	10
1.2. Soluciones similares	10
1.2.1. ZOEa Difusión e Investigación del Medio Marino S.L.	10
1.2.2. Red Nacional de Manejo Integrado de Zonas Costeras en Colombia – RedCostera.....	11
1.2.3. Sistema Nacional de Áreas Protegidas de Cuba	12
1.2.4. Red Cubana de la Ciencia	13
1.2.5. Aportes del estudio de las aplicaciones a la propuesta de solución.....	14
1.3. Lenguajes de programación	15
1.3.1 Lenguajes de programación del lado del servidor.....	15
1.3.1.1. PHP 5.3.8	15
1.3.2. Lenguajes de programación del lado del cliente	16
1.3.2.1. Javascript 1.8.5	16
1.3.2.2. CCS 3	17
1.3.2.3. XHTML 1.1	17
1.4. Sistema de gestión de contenidos	18
1.5. Frameworks de desarrollo web	19
1.5.1. <i>Frameworks</i> del lado del cliente	20
1.5.1.1. jQuery 1.9	21
1.5.1.2. ExtJS 3.4	21
1.5.2. <i>Frameworks</i> del lado del servidor	21



1.5.2.1. Symfony 2.0.....	22
1.5.2.2. CodeIgniter 2.1.3.....	22
1.5.2.3. Doctrine ORM 1.2.....	23
1.6. Metodologías de desarrollo de <i>software</i>	24
1.6.1. Metodologías tradicionales	24
1.6.2. Metodologías ágiles	25
1.6.2. Scrum.....	25
1.6.2.2. Extreme programming (XP)	26
1.7. Sistemas gestores de bases de datos	28
1.7.1 MySQL 5.6.9.....	28
1.7.2. PostgreSQL 9.1.....	28
1.8. Entorno de desarrollo integrado	30
1.8.1. Zend studio 9.0.1.....	30
1.8.2. Netbeans 7.3	30
1.9. Servidores web	31
1.9.1 Servidor IIS 7.0.....	31
1.9.2. Apache 2.2.22	31
Conclusiones.....	33
Capítulo 2: Propuesta de solución	34
2.1. Introducción.....	34
2.2. Sistemas de control de acceso.....	34
2.2.1. RBAC	35
2.3. Historias de usuario	36
2.4. Estimación de esfuerzos por HU	38
2.5. Plan de iteraciones	39
2.6. Plan de duración de las iteraciones	41
2.7. Plan de entrega.....	42
2.8. Prototipo de interfaz de usuario no funcional.....	43
2.9. Tarjetas CRC	44
Conclusiones.....	46



Capítulo 3: Implementación y prueba	47
3.1. Introducción.....	47
3.2. Estructura de CodeIgniter	47
3.3. Estructura de la carpeta apps.....	49
3.4. Módulos de la aplicación	49
3.5. Tareas de ingeniería	50
3.6. Patrones de diseño usados en la solución.....	53
3.7. Estándares de codificación.....	55
3.8. Nomenclatura.....	55
3.8.1. Nomenclatura de las clases según el tipo	55
3.8.2. Nomenclatura de los métodos o funciones	55
3.8.3. Nomenclatura de las variables.....	56
3.9. Creación de las tablas en la base de datos	56
3.10. Pantallas de la aplicación	58
3.11. Pruebas	59
3.11.1. Pruebas de aceptación de la iteración 1	60
3.11.2. Pruebas de aceptación de la iteración 2	63
Conclusiones.....	71
Conclusiones generales	72
Recomendaciones	73
Trabajos citados	74
Anexo I.....	80
Anexo II.....	89
Anexo III.....	106
Anexo IV	128
Glosario de términos.....	149



Índice de figuras

Figura 1: ZOEА Difusión e Investigación del Medio Marino S.L.	11
Figura 2: Red Nacional de Manejo Integrado de Zonas Costeras – RedCostera.	12
Figura 3: Sistema Nacional de Áreas Protegidas de Cuba.	13
Figura 4: Red Cubana de la Ciencia.	14
Figura 5: Ciclo de trabajo de Scrum.....	26
Figura 6: Fases del ciclo de vida de la metodología XP.....	27
Figura 7: Prototipo de interfaz de usuario no funcional de la página principal.	44
Figura 8: Estructura general de Codeigniter.	47
Figura 9: Estructura de la aplicación.....	48
Figura 10: Flujo de datos de codeIgniter.....	48
Figura 11: Estructura de la carpeta apps.	49
Figura 12: Estructura de la carpeta web.	50
Figura 13: Estructura de las tablas en la base de datos.	57
Figura 14: Interfaz principal.	58
Figura 15: Interfaz gestionar inspección.	59
Figura 16: No conformidades significativas, no significativas y las recomendaciones.....	71

Índice de tablas

Tabla 1: Formato general de las HU	36
Tabla 2: HU de Gestionar Inspección	37
Tabla 3: HU de Gestionar Licencia Ambiental	38
Tabla 4: Estimación de esfuerzos por historia de usuario.....	38
Tabla 5: Plan de duración de las iteraciones	41
Tabla 6: Plan de Entrega	42
Tabla 7: Tarjeta CRC Inspeccionmodel.....	45
Tabla 8: Tarjeta CRC Inspeccion.plan.Cronograma.Vista.....	45
Tabla 9: Tarjeta CRC Inspeccion	46
Tabla 10: Formato general de la tarea de ingeniería.....	51
Tabla 11: Tarea de ingeniería adicionar inspección.....	51
Tabla 12: Tarea de ingeniería modificar inspección.....	52
Tabla 13: Tarea de ingeniería eliminar inspección	53
Tabla 14: Caso de prueba de aceptación HU2_P1	60
Tabla 15: Caso de prueba de aceptación HU2_P2	60
Tabla 16: Caso de prueba de aceptación HU2_P3	61
Tabla 17: Caso de prueba de aceptación HU4_P2	62
Tabla 18: Caso de prueba de aceptación HU4_P3	63
Tabla 19: Caso de prueba de aceptación HU5_P1	64
Tabla 20: Caso de prueba de aceptación HU5_P2	64



Tabla 21: Caso de prueba de aceptación HU5_P3	65
Tabla 22: Caso de prueba de aceptación HU10_P1	66
Tabla 23: Caso de prueba de aceptación HU10_P2	66
Tabla 24: Caso de prueba de aceptación HU10_P3	67
Tabla 25: Caso de prueba de aceptación HU11_P1	68
Tabla 26: Caso de prueba de aceptación HU11_P2	69
Tabla 27: Caso de prueba de aceptación HU11_P3	70



Introducción

Actualmente es imposible pensar en un mundo sin tecnologías de la información y de las comunicaciones (TIC). Las TIC con el paso del tiempo han estado cada vez más presentes en las actividades humanas. Las sociedades modernas son un reflejo del modo en el que la tecnología ha ingresado en los distintos ámbitos de la realidad social, histórica, industrial y cultural repercutiendo directamente en la vida cotidiana, situando a la mente del ser humano como la fuerza productiva de mayor importancia en la actualidad, al generar pensamientos en función a servicios, y administrarlos más allá de las fronteras geográficas, sociales y culturales; ya que se fragua en el centro mismo de los movimientos globalizantes que exigen un alto nivel de competitividad y designan a la vez un conjunto de innovaciones tecnológicas y herramientas que permiten una redefinición radical del funcionamiento de la sociedad.

La construcción del socialismo no se pudo sustraer de un enfoque antropocéntrico de la problemática ambiental por la necesidad de elevar a un primer plano las acciones encaminadas a la supervivencia de sus pueblos. A pesar de ello, los marxistas nunca perdieron la visión de lo global e interdisciplinario de la problemática ambiental, desde Engels, quien en carta a Jules Guedes escribió “...*porque, al fin y al cabo, la naturaleza y la historia son dos componentes del medio en que vivimos, nos movemos y nos manifestamos*” (1) hasta los más contemporáneos que exponen que “...*revelar la interacción dialéctica de la sociedad y la naturaleza significa mostrar las contradicciones intrínsecas de este proceso, en cuya solución se crea una nueva calidad, se logra la unidad de una y la otra*” (2).

Antes del primero de enero de 1959 no formaba parte de las políticas de los gobernantes preocuparse del impacto que sobre el medio ambiente tenían las empresas a la hora de su instalación o producción, mientras que después del triunfo de la Revolución cambia totalmente la situación en cuanto a la política de las autoridades gubernamentales de preocuparse por la interacción empresas – medio ambiente; con la creación de medidas y/o políticas a todo lo largo y ancho de la nación, proclamándose una serie de leyes y decretos tales como la Ley 81, Decreto Ley 200 y Decreto Ley 99, que contribuye a la obligatoriedad tanto de las autoridades gubernamentales y/o administrativas, como de la población, con el único y marcado fin de mejorar las condiciones ambientales del país.



La preocupación de las ciencias por estudiar y resolver el grave dilema que enfrenta hoy día la humanidad en cuanto a la salvaguarda del entorno socio-natural, obliga a abordar seriamente este tema no solo en lo multidisciplinario sino también en lo transdisciplinario.

La protección del Sistema Ambiental Global ha sido uno de los principales objetivos de la humanidad en los últimos años. Para ello se han realizado variadas y complejas tareas para lograr identificar las causas y los efectos de los problemas ambientales, y poder así trazar estrategias que permitan frenar la acelerada e irracional explotación de los recursos naturales.

En Cuba el Ministerio de Ciencia, Tecnología y Medio Ambiente (CITMA) es el organismo encargado de dirigir, ejecutar y controlar la política del Estado y el Gobierno en la actividad científica, tecnológica y ambiental.

Por la necesidad de contar con una entidad que regule y controle la actividad en el país con respecto a la seguridad ambiental, biológica, química y nuclear se crea por medio de la Resolución 4/2005 la Unidad Presupuestada “Oficina de Regulación Ambiental y Seguridad Nuclear”, en forma abreviada ORASEN, subordinada al CITMA, la que tiene como objetivos esenciales: (3)

- ❖ Organizar y gerenciar el proceso de concesión de autorizaciones, permisos y licencias en las esferas reguladoras de control ambiental, la seguridad biológica, la seguridad nuclear y radiológica y el control de armas químicas, en correspondencia con la legislación vigente.
- ❖ Otorgar, modificar, renovar y suspender o revocar autorizaciones, permisos y licencias en las esferas de su competencia.

El Centro de Inspección y Control Ambiental (CICA) es la entidad adscrita a la ORASEN encargada de elaborar, proponer, controlar y supervisar, según corresponda, y dentro del ámbito de su competencia, las medidas dirigidas a garantizar la protección del medio ambiente y el uso racional de los recursos naturales. El mismo realiza Evaluaciones de Impacto Ambiental, autoriza y habilita a las entidades ejecutoras, organiza, dirige y ejecuta la Inspección Ambiental Estatal; así como la adopción de medidas y el control de su ulterior cumplimiento, además concede, renueva, modifica, suspende y cancela las licencias ambientales según corresponda. En dicho centro se encuentran los procesos de Inspecciones Ambientales Estatales (IAE), Evaluación de



Impacto Ambiental (EIA), la aplicación de Contravenciones, Licencias Ambientales (LA), entre otros, donde cada uno de ellos genera información que es importante archivar para el control y estudio de sus comportamientos. En la actualidad la entidad mencionada, para el registro y control de los datos e información, emplea archivos y carpetas en formato duro¹, las que generan gran volumen de documentación, haciendo muy engorrosa y compleja para los expertos, cualquier búsqueda o consulta de datos para su estudio.

Por una parte el CICA posee personal capacitado y herramientas informáticas que pudiera disponer para facilitar los procesos de inspección y control ambiental, sin embargo no emplean las potencialidades de las TIC puestas a su disposición de manera efectiva para lograr este objetivo. La situación descrita nos conduce a plantear el siguiente **problema de investigación**: ¿Cómo favorecer el control y manejo de la información por los expertos en el CICA?

Objetivo general:

Desarrollar una herramienta informática que favorezca el control y manejo de la información por los expertos del CICA.

El **objeto de investigación** lo constituyen los procesos de gestión de información.

El **campo de acción** se centra en las particularidades de desarrollo de una herramienta informática para la gestión de información de los procesos de inspección y control ambiental en el CICA.

Objetivos específicos:

- ❖ Caracterizar los sistemas de gestión de información referidos a procesos ambientales, así como las tendencias, tecnologías y metodologías más utilizadas en el desarrollo de este tipo de herramientas.
- ❖ Diseñar la herramienta de gestión de información como propuesta de solución del problema planteado.
- ❖ Implementar las funcionalidades para la herramienta de gestión de información que cumpla con los requerimientos definidos.

¹ Por formato duro, en esta investigación se entiende, toda información que no esté contemplada en soporte digital.



La **idea a defender** que sustenta a la investigación es que el desarrollo de un sistema de gestión de información permitirá favorecer el control y manejo de la información por los expertos del CICA.

El presente estudio se desarrolla con el empleo de la combinación dialéctica de los métodos teóricos y empíricos de investigación. Entre los métodos científicos teóricos utilizados se encuentran:

- ❖ **Método Histórico-Lógico:** El uso de este método permitió la realización del estudio del arte de la ciencia que está siendo objeto de investigación y de las tendencias actuales para el desarrollo de la herramienta de gestión de información.
- ❖ **Método Analítico-Sintético:** Este método permitió el análisis de los aspectos estudiados para la elaboración del marco conceptual que sustenta la investigación.
- ❖ **Método Inductivo-Deductivo:** Se realizó durante todo el proceso investigativo con el objetivo de prever y premeditar posibles causas, consecuencias y resultados de los elementos que tengan lugar.

Entre los métodos empíricos se empleó:

- ❖ **Entrevista de tipo abierta** (fundamentalmente a los clientes): Permitted conocer la fenología de los procesos de inspección y control ambiental en el CICA y delimitar funcionalidades a desarrollar.

Para dar cumplimiento a los objetivos trazados se definieron las siguientes **tareas de investigación:**

- ❖ Análisis de los fundamentos teóricos y metodológicos que sustentan la creación de la herramienta de gestión de información de los procesos de inspección y control ambiental en el CICA.
- ❖ Aplicación de técnicas de recopilación de información a los trabajadores del CICA.
- ❖ Estudio de soluciones similares existentes a nivel nacional e internacional.
- ❖ Selección de la metodología de desarrollo de *software* a utilizar para el diseño de la aplicación.



- ❖ Definición de las funcionalidades que debe cumplir el sistema.
- ❖ Realización de los prototipos de interfaz de usuario no funcionales.
- ❖ Investigación y selección de las herramientas de programación y las tecnologías para el desarrollo de herramientas de gestión de información.
- ❖ Realización y análisis de pruebas que se definan y sean necesarias para garantizar el cumplimiento de los objetivos.

El presente trabajo está estructurado en:

Capítulo 1. Fundamentación teórica: En este capítulo se abordan los aspectos teóricos que dan sustento a la investigación y que corresponden al análisis detallado de las metodologías, herramientas y tecnologías a utilizar en el proceso de desarrollo de la herramienta. Se concreta un estudio de las tendencias actuales para el desarrollo de funcionalidades para la herramienta de gestión de información de los procesos de inspección y control ambiental, además de exponerse definiciones claves para una fácil comprensión del trabajo de diploma.

Capítulo 2. Propuesta de solución: En este capítulo se presenta el diseño de la propuesta de solución al problema planteado, que contará con la determinación de los módulos y funcionalidades que brindará la herramienta, definiéndose las funcionalidades que debe cumplir, así como el diseño y la estructuración de la misma.

Capítulo 3. Implementación y prueba: En este capítulo se explica cómo se construyó la herramienta y los módulos de la misma, así como su configuración. Se describen, además, las pruebas realizadas con el objetivo de validar el correcto funcionamiento de la solución.

Al final del documento se incluyen las referencias bibliográficas, un glosario de términos y los anexos referenciados.



Capítulo 1: Fundamentación teórica

1.1. Marco conceptual

Para una mayor comprensión de la problemática se hace necesario un marco conceptual que ayude a comprender y utilizar las ideas de otras personas que han hecho trabajos similares, por qué se escogieron ciertos métodos y no otros para llegar a un punto determinado y explicar por qué se lleva a cabo de cierta manera esta investigación.

Para el desarrollo de este sistema de gestión de información se hicieron énfasis en los conceptos de gestión de la información, inspección ambiental estatal, licencia ambiental, evaluación de impacto ambiental, contravenciones y control ambiental.

1.1.1. Gestión de la información

La bibliografía consultada permite a las autoras de la presente investigación coincidir con la investigadora Lic. Lourdes Aja Quiroga, que define la gestión de información como “... *parte esencial de la infraestructura para la gestión del conocimiento, suministran información, impulsan la generación del conocimiento para la búsqueda de soluciones a los problemas que enfrentan las organizaciones, analizan su impacto sobre los resultados de las empresas e influyen en el comportamiento de los individuos ante la información*”. En otro momento, la investigadora mencionada plantea que “*La gestión de la información se vincula con la generación y la aplicación de estrategias, el establecimiento de políticas, así como con el desarrollo de una cultura organizacional y social dirigida al uso racional, efectivo y eficiente de la información en función de los objetivos y metas de las compañías en materia de desempeño y de calidad*” (4).

La bibliografía examinada permitió además consultar a otros autores que definen un sistema de información como: “...*un conjunto de componentes relacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar a la toma de decisiones y el control de una organización*” (5).

Según la bibliografía consultada existen tres actividades fundamentales dentro de un sistema de gestión de información. En este aspecto y en cumplimiento del objetivo general de la investigación que se presenta, se enuncian las mismas (5):



Capítulo **1** *Fundamentación teórica*

- ❖ Entrada: captura o colecta datos en bruto tanto del interior de la organización como de su entorno externo.
- ❖ Procesamiento: convierte esa entrada de datos en una forma más significativa.
- ❖ Salida: transfiere la información procesada a los usuarios que la usará o a las actividades para las que se utilizará.

Por todo lo anterior descrito, para esta investigación se define un Sistema de Gestión de Información Ambiental como un sistema que recogerá datos en bruto resultantes de los procesos de inspección y control ambiental, los procesará y permitirá mostrarlos de forma que puedan ser analizados de manera más novedosa, así como tomar decisiones con respecto a la protección ambiental, realizar comparaciones en caso de ser necesario, entre otros aspectos.

1.1.2. Inspección Ambiental Estatal

A partir del estudio bibliográfico de los decretos y leyes existentes en Cuba se realiza un análisis detallado de la Ley No. 81 denominada Ley del Medio Ambiente la cual tiene como objeto establecer los principios que rigen la política ambiental y las normas básicas para regular la gestión ambiental del Estado y las acciones de los ciudadanos y la sociedad en general, a fin de proteger el medio ambiente y contribuir a alcanzar los objetivos del desarrollo sostenible del país, en su artículo 8 enuncia que IAE es: “... *una actividad de control, fiscalización y supervisión del cumplimiento de las disposiciones y normas jurídicas vigentes en materia de protección del medio ambiente, con vista a evaluar y determinar la adopción de las medidas pertinentes para garantizar dicho cumplimiento*”(6).

1.1.2.1. Objetivos de la IAE

Después de realizar un análisis de la bibliografía se pudo constatar que las IAE poseen como objetivos: controlar el cumplimiento de la legislación ambiental vigente, contribuir a que las actividades de producción de bienes y de prestación de servicios se realicen de modo que permitan la protección del medio ambiente y al logro de las metas del desarrollo sostenible, además de prevenir delitos en el desarrollo de las actividades mencionadas y disponer las medidas que correspondan (7).

1.1.3. Licencias Ambientales

Dentro de la Ley 81 se formula que una LA es “...*un documento oficial, que sin perjuicio de otras licencias, permisos y autorizaciones que de conformidad con la legislación vigente corresponda*



Capítulo 1 *Fundamentación teórica*

conceder a otros órganos y organismos estatales, es otorgado por el Ministerio de Ciencia, Tecnología y Medio Ambiente para ejercer el debido control al efecto del cumplimiento de lo establecido en la legislación ambiental vigente y que contiene la autorización que permite realizar una obra o actividad” (6).

En los artículos 24, 25 y 26 de la Ley No. 81 del estado cubano plantea que toda actividad susceptible de producir efectos significativos sobre el medio ambiente o que requiera un debido control a los efectos del cumplimiento de lo establecido por la legislación ambiental vigente, estará sujeta al otorgamiento de una LA por el CITMA, de conformidad con lo que al respecto estipule ese organismo, quien establecerá así mismo los tipos y modalidades de dicha licencia y que el otorgamiento de la licencia ambiental está sujeta al pago de los gravámenes que al respecto se establezcan y no exime al licenciatarario de la obligación de proteger de manera efectiva el medio ambiente, ni de las responsabilidades administrativas, civiles y penales en que pueda incurrir, además que los programas, obras o actividades que no cuenten con la licencia ambiental o no cumplan las exigencias y controles que en ésta se fijen, podrán ser suspendidas temporal o definitivamente por el CITMA, sin perjuicio de que se hagan efectivas las responsabilidades correspondientes. Para mayor información consultar (6).

1.1.4. Evaluación de Impacto Ambiental

Dentro de las clasificaciones de evaluación de impacto se encuentra la EIA, donde según la ley 81 la evaluación de impacto ambiental es: “...*procedimiento que tiene por objeto evitar o mitigar la generación de efectos ambientales indeseables, que serían la consecuencia de planes, programas y proyectos de obras o actividades, mediante la estimación previa de las modificaciones del ambiente que traerían consigo tales obras o actividades y, según proceda, la denegación de la licencia necesaria para realizarlos o su concesión bajo ciertas condiciones. Incluye una información detallada sobre el sistema de monitoreo y control para asegurar su cumplimiento y las medidas de mitigación que deben ser consideradas” (6).*

Una vez realizado el proceso de EIA por las empresas consultoras, la información generada ya sea en formato duro o digital es entregada al CICA mediante una ventanilla única. Una vez recepcionada la información por la entidad, un experto es el encargado de verificar la misma y archivarla en un libro de registros de procesos de EIA aceptándola o denegándola en dependencia del contenido. Al ser examinada se procede a convocar expertos los cuales,



Capítulo 1 *Fundamentación teórica*

después de considerar la misma y verificar que esté en su totalidad, dictaminan una serie de aspectos con relación al otorgamiento o denegación de la licencia ambiental, de estar incompletos los datos se detienen los términos de tramitación de la solicitud de LA y se le notifica al beneficiario la necesidad de presentar por escrito la información requerida, en el caso de no recibirse esta información en el término de un año, será dada de baja la solicitud tal como se establece en la Resolución 132/09 del CITMA.

1.1.5. Contravenciones

Según la bibliografía consultada el término contravención es: *“...un término del ámbito del derecho que se utiliza para designar a aquellos actos que van en contra de las leyes o lo legalmente establecido y que por lo tanto pueden representar un peligro tanto para quien lo lleva a cabo como también para otros”* (8).

En el Decreto Ley 200 en su artículo 5 expone que en los procesos de EIA y otorgamiento de la Licencia Ambiental, se consideran contravenciones las siguientes (9):

- ❖ No someter a consideración del Ministerio los nuevos proyectos de obras o actividades que aparecen refrendados en el Artículo 28 de la Ley, previo a su ejecución y para la realización del proceso de evaluación de impacto ambiental.
- ❖ Realizar otras actividades cuya ejecución esté precedida o su desarrollo requerido de una LA, de conformidad con las disposiciones que establezca el Ministerio al amparo del Artículo 24 de la Ley, sin haber obtenido previamente dicha licencia, o habiéndose denegado ésta.

Por otro lado en el Decreto Ley 200 en su artículo 6 son consideradas contravenciones respecto a la inspección ambiental estatal las siguientes (9):

- ❖ Dificultar o impedir el acceso de los inspectores ambientales estatales a las áreas o lugares a ser inspeccionados.
- ❖ Incumplir con las medidas dictadas por la inspección ambiental estatal para la adopción de medidas correctivas de adecuación a las disposiciones ambientales vigentes.
- ❖ No proporcionar la información que le sea debidamente requerida por los inspectores ambientales estatales en el desempeño de sus funciones, entregar información inexacta u



Capítulo 1 *Fundamentación teórica*

ocultar datos u otras informaciones solicitadas para el normal desempeño de la inspección ambiental estatal.

1.1.6. Control ambiental

Según la bibliografía analizada el control ambiental es: *“...un instrumento de dirección que le proporciona a la dirección de la compañía la información y medios que ayudan a mejorar los procesos relacionados con el medioambiente”* (10).

Las autoras coinciden con Néstor Julio Jaime Restrepo el cual plantea que el control ambiental es: *“... inspección, vigilancia y aplicación de las medidas legales y técnicas que se aplican y son necesarias para disminuir o evitar, cualquier tipo de afección al medio ambiente en general, y a un ecosistema en particular, producto de las actividades humanas, o por desastres naturales; lo mismo que para disminuir los riesgos para la salud humana. Se incluyen inventarios, muestreo, censo, etc. El caso más conocido de control ambiental es el relacionado con la emisión de contaminantes, provenientes de procesos creados por el hombre al medio ambiente, ya sea al aire, agua o suelo, y aquellos diseñados para disminuir los riesgos sobre la salud humana”* (11).

1.2. Soluciones similares

Para la propuesta de solución se hizo necesario un análisis de la documentación y organizaciones existente a nivel nacional e internacional, referente al desarrollo de aplicaciones de sistemas de gestión de información que permitan dar cumplimiento al objetivo general. El resultado de la misma se describe a continuación.

1.2.1. ZOEa Difusión e Investigación del Medio Marino S.L.

ZOEa Difusión e Investigación del Medio Marino S.L. es una empresa del sector submarinismo en España que ha implantado un sistema de gestión ambiental e incorpora medidas para la protección del medio ambiente, el mismo exige el cumplimiento de la legislación ambiental vigente, la reducción y correcta gestión de residuos generados y el ahorro de energía, se realiza además análisis de pesquería, informes de capturas, evaluación de caladeros, análisis del esfuerzos pesqueros, modelos de transporte de sedimento, cuantificación de variables ambientales de ecosistemas acuáticos, descripción de las actividades asociadas a un proyecto, la identificación de impactos, el diseño de medidas correctoras, elaboración de programas de seguimiento y vigilancia ambiental entre otras. Este sistema cuenta con mapas que muestran la



Capítulo 1 *Fundamentación teórica*

localización de cada una de las áreas marinas a investigar, posee además gráficos que indican la escalabilidad de cada proceso analizado por la aplicación dando como resultado del análisis una muestra de cada uno en cada iteración.

Disponible en: <http://zoemadrid.com/conocenos/sistema-de-gestion-ambiental/>



Figura 1: ZOE A Difusión e Investigación del Medio Marino S.L.

1.2.2. Red Nacional de Manejo Integrado de Zonas Costeras en Colombia – RedCostera

Red Nacional de Manejo Integrado de Zonas Costeras en Colombia – RedCostera tiene como objetivo proveer información general sobre el tema, experiencias relacionadas con el Manejo Integrado de Zonas Costeras (MIZC) y su uso potencial para el desarrollo a nivel nacional y regional, difundir información de las actividades y eventos (entrenamiento, educación y capacitación) en el ámbito nacional e internacional, establecer un directorio de personas e instituciones que estén trabajando en el manejo integrado de zonas costeras y enlazarlo con los directorios internacionales del tema así asesorar sobre los diferentes procesos de planeación y manejo integrado de zonas costeras, a través del intercambio de información y experiencias. Dicha información es manejada mediante perfiles sedimentológico, análisis de componentes, mapas cartográficos, gráficos de comportamiento de indicadores ambientales marinos y costeros de Colombia, entre otras características.

Disponible en: <http://www.invemar.org.co>



Figura 2: Red Nacional de Manejo Integrado de Zonas Costeras – RedCostera.

1.2.3. Sistema Nacional de Áreas Protegidas de Cuba

El Sistema Nacional de Áreas Protegidas de Cuba (SNAP) está constituido por las áreas naturales, terrestres y marinas; más importantes del país. La principal misión del SNAP es garantizar la protección y conservación de los recursos naturales del patrimonio nacional para uso de las actuales y futuras generaciones, como parte del desarrollo sostenible del país. Posee un mapa cartográfico del país donde se ubican las áreas protegidas de las diferentes provincias, un mapa mundial de localización con una escala de colores según las reservas de la biosfera y diferentes mapas del país según el criterio de localización.

Disponible en: <http://www.snap.cu/>



Figura 3: Sistema Nacional de Áreas Protegidas de Cuba.

1.2.4. Red Cubana de la Ciencia

La Red Cubana de la Ciencia brinda información sobre el quehacer científico cubano en los diferentes campos del conocimiento. Este sitio ofrece contenidos y servicios de información concebidos como un espacio de colaboración e interacción con instituciones, científicos y otros profesionales de Cuba y el resto del mundo, posee cursos en línea, Programas Nacionales de Ciencia y Técnica (PNCT) los cuales muestran en un gráfico la cantidad de instituciones ejecutoras principales por los PNCT, enlaces a información de interés, noticias, eventos, entre otros.

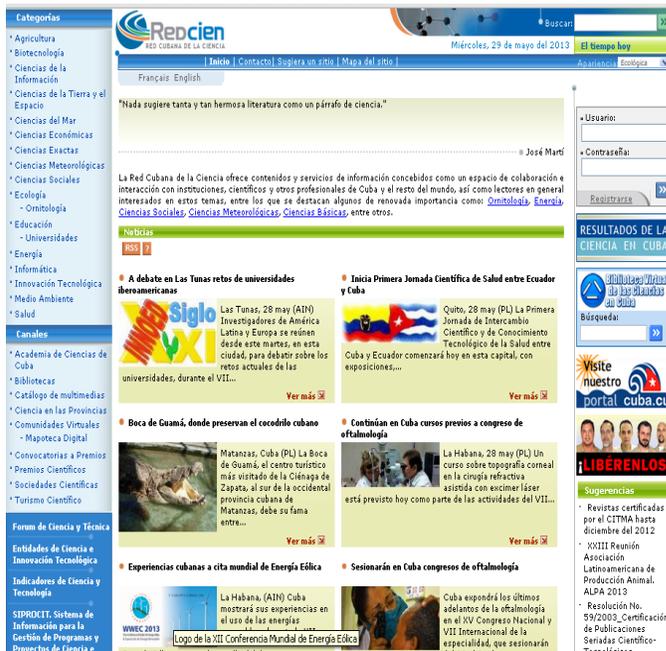


Figura 4: Red Cubana de la Ciencia.

Disponible en: <http://www.redciencia.cu>

1.2.5. Aportes del estudio de las aplicaciones a la propuesta de solución

Después del análisis de varios sistemas similares se pudo constatar que según las tendencias actuales los sistemas de gestión de información están encaminados al desarrollo web. Dichos sistemas no poseen funcionalidades que engloben los procesos de inspección y control ambiental, es por ellos que son descartados y se decide realizar un sistema de gestión de la información sin embargo los sistemas analizados aportan una serie de elementos significativos para la futura propuesta de solución tales como: análisis de componentes, gráficos de comportamiento de indicadores, se pudo analizar la forma en que generan los reportes, la visualización de la información, la generación automática de informes, el manejo de datos mediante estadísticas, elaboración de programas de seguimiento y vigilancia ambiental entre otras. Se concluye que las características mencionadas de los diferentes sistemas similares con respecto a la protección ambiental, brindarán un espacio donde el usuario podrá interactuar con la información necesaria sobre el tema, permitiéndole generar reportes que le sean de gran utilidad y lograr una mejor comprensión de la misma.



1.3. Lenguajes de programación

Un lenguaje de programación “...es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar” (12).

Un lenguaje de programación “...es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente” (13).

Por todo lo antes planteado las autoras concuerdan que un lenguaje de programación es el conjunto de reglas que se introducen en una computadora para controlar su comportamiento.

Los lenguajes de programación para el desarrollo de aplicaciones web se apilan en dos grupos: lenguajes del lado del cliente y lenguajes del lado del servidor.

1.3.1 Lenguajes de programación del lado del servidor

Los lenguajes del lado del servidor son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Este tipo de lenguaje es independiente del cliente por lo que es mucho menos rígido respecto al cambio de un navegador a otro o con relación a las versiones del mismo.

Dentro de los lenguajes de programación del lado del servidor se encuentran: CGI², PHP³, PERL, Java Server Pages (JSP), Microsoft Active Server Pages (ASP) entre otros. Siendo escogido PHP para el desarrollo de la propuesta de solución por presentar varias ventajas de gran utilidad en la implementación del sistema, a continuación se describe su esencia.

1.3.1.1. PHP 5.3.8

PHP es el acrónimo de *Hipertext Preprocesor* (inicialmente se llamó Personal Home Page); surgió en 1995, desarrollado por PHP Group. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación (14).

² Common Gateway Interface (CGI). Es de las primeras formas de programación web dinámica. En sí, es un método para la transmisión de información hacia un compilador instalado en el servidor.

³ PHP Lenguaje de programación del lado del servidor, sus siglas significan *HypertextPre-Processor*.



Es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas *HTML*⁴ y ejecutadas en el servidor, no necesita ser compilado para ejecutarse, para su funcionamiento necesita tener instalado los servidores Apache o IIS (*Internet Information Server*) con las librerías de *PHP*, la mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas y los archivos cuentan con la extensión “php”. Además soporta la orientación a objeto, clases y herencia, es un lenguaje multiplataforma, tiene la capacidad de expandir su potencial utilizando módulos, no requiere definición de tipos de variables ni manejo detallado de bajo nivel y posee la capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras (15).

1.3.2. Lenguajes de programación del lado del cliente

Los lenguajes del lado del cliente son aquellos que pueden ser directamente "digeridos" por el navegador y no necesitan un pre-tratamiento. Existen varios lenguajes de programación del lado del cliente, como son: *Applets*⁵ de Java, Flash, VBScript, Javascript, Jscript, HTML, *DHTML*⁶, CSS⁷, entre otros; donde en la presente investigación se analizan Javascript, *XHTML*⁸ y CSS.

1.3.2.1. Javascript 1.8.5

Es un lenguaje interpretado, no requiere compilación, fue creado por Brendan Eich en la empresa Netscape Communications, es utilizado principalmente en páginas web, es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Javascript, este código puede ser integrado dentro de nuestras páginas y para evitar incompatibilidades el *World Wide Web Consortium* (W3C) diseñó un estándar denominado DOM (15).

⁴ HTML es el acrónimo de *HyperText Markup Language* (Lenguaje de Marcado de Hipertexto) y es el lenguaje que se utiliza para crear páginas web.

⁵ Applet es un componente de software que corre en el contexto de otro programa.

⁶ DHTML en inglés Dynamic HTML, y significa HTML dinámico.

⁷ Hojas de Estilo en Cascada, CSS por sus siglas en inglés, *Cascading Style Sheets*.

⁸ Lenguaje Extensible de Marcado de Hipertexto, por sus siglas en inglés, *eXtensible HyperText Markup Language*.



Capítulo 1 *Fundamentación teórica*

Javascript es un lenguaje de programación que realiza acciones dentro del ámbito de una página web. Su compatibilidad con la mayoría de los navegadores modernos lo posiciona como el lenguaje de programación del lado del cliente más utilizado.

1.3.2.2. CCS 3

Las Hojas de Estilo en Cascada, CSS por sus siglas en inglés, (*Cascading Style Sheets*), es *“...un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos”* (16).

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Este permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo y que cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a la misma en las que aparezca el mismo. Además funciona a base de reglas, es decir, declaraciones sobre el estilo de uno o más elementos, las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML. La regla posee dos partes: un selector y la declaración y a su vez dicha declaración está compuesta por una propiedad y el valor que se le asigne (16).

1.3.2.3. XHTML 1.1

El Lenguaje Extensible de Marcado de Hipertexto, XHTML por sus siglas en inglés, *eXtensible HyperText Markup Language* es: *“... un lenguaje de programación pensado para sustituir a HTML. XHTML es la versión XML de HTML con las mismas funcionalidades, pero cumple las especificaciones, más estrictas de XML.”* (17)

XHTML presenta un gran número de mejoras con respecto a HTML como:

- ❖ Los documentos XHTML son conformes a XML, son fácilmente visualizados, editados y validados con herramientas XML estándar.
- ❖ Los documentos XHTML pueden escribirse para que funcionen igual o mejor que como lo realizaban antes.
- ❖ Los documentos XHTML pueden usar aplicaciones (por ejemplo scripts y applets) que se basen ya sea en el Modelo del Objeto Documento de HTML o XML [DOM].



Capítulo 1 *Fundamentación teórica*

El lenguaje XHTML es una familia de módulos y tipos de documentos que reproduce, engloba y extiende HTML 4.0. Los tipos de documentos de la familia XHTML están basados en XML y diseñados fundamentalmente para trabajar en conjunto con aplicaciones de usuario basados en XML (18).

Algunas de las ventajas que presenta este lenguaje es la facilidad de edición directa del código y mantenimiento, el formato abierto compatible con los nuevos estándares que está desarrollando el W3C y la compatibilidad parcial con navegadores antiguos. El principal inconveniente que presenta es que pueden aumentar mucho el tamaño del documento, por lo que en general se utilizan etiquetas con nombres muy cortos.

1.4. Sistema de gestión de contenidos

El análisis de la bibliografía describe un Sistema de Gestión de Contenido o CMS (*Content Management System*) como: “...término genérico que abarca un amplio conjunto de soluciones cuya funcionalidad y alcance depende del tipo de contenidos que gestionen y del ámbito de aplicación de la solución” (19).

Un Sistema de Gestión de Contenido Web (WCM) es una herramienta que permite crear, editar, modificar y publicar contenidos Web e incluye de forma más habitual funcionalidades avanzadas como gestión de permisos, sistemas de búsquedas, entornos colaborativos, entre otras.

A partir del método científico análisis y síntesis se identificaron algunas de las características más importantes de un CMS: puede implicar la revisión de multitud de páginas y la generación de códigos que aportan las funcionalidades. Brinda mantenimiento a gran cantidad de páginas, reutiliza objetos o componentes, permite la interactividad entre las páginas, entre otras características.

Después del análisis de la bibliografía consultada se arriba a la conclusión que no se escogen para el desarrollo de la propuesta de solución los CMS por presentar varios inconvenientes, tales como: vulnerabilidades o agujeros que contribuyen a la acción de los hackers, por lo tanto poseen menor seguridad, las consultas a la base de datos son altas por ser plataformas genéricas, por tanto cuanto mayor información haya almacenada más lento será el sistema, presentan una ralentización excesiva de la velocidad de carga, ya que tiene que recorrer mayor



número de procesos, tienen menor control sobre los datos. En la comunidad es mucho más difícil de integrar cualquier funcionalidad que se requiera ya que el desarrollo debe ser adaptado al CMS y muchas veces sucede que la plataforma impide dicha adaptación; siendo esto de gran importancia para el desarrollo de la aplicación y por ende para el cumplimiento de su objetivo general, así como de la satisfacción de las necesidades del cliente. (20)

1.5. Frameworks de desarrollo web

Según Ralph Johnson y Brian Foote en el año 1988 un “*Software Framework*”⁹ es: “*Un diseño reusable de un sistema (o subsistemas). Está expresado por un conjunto de clases abstractas y el modo en que sus instancias colaboran para un tipo específico de software. Todos los frameworks de software son orientados a objetos*” (21).

La bibliografía consultada enuncia que un *framework* se puede considerar como: “... *una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta*” (22).

Se coincide con la bibliografía analizada la cual expresa que un *framework web* es “...*una estructura definida, reusable en el que los componentes facilitan la creación de aplicaciones web, provee de una capa de abstracción sobre la arquitectura original ocultándola o adaptándola para no tener que utilizar el protocolo http de manera nativa y así acelerar los tiempos de desarrollo y mantenimiento*” (23).

Por lo que puede concluirse que un *framework* es un diseño reusable de un sistema compuesto de componentes personalizables e intercambiables para el desarrollo de una aplicación.

Los objetivos principales que persigue un *framework* son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas como el uso de patrones. Para desarrollar una herramienta con características similares a la que se propone, se necesitaría la integración de varios *frameworks* especializados, por lo que es de trascendental importancia el grado de acoplamiento entre ellos.

Para comprender como trabajan los *frameworks web* existentes es imprescindible conocer el patrón Modelo-Vista-Controlador (MVC), el cual es una guía para el diseño de arquitecturas de

⁹La traducción de *framework*(o en plural *frameworks*) en idioma español es marco de trabajo.



Capítulo 1 *Fundamentación teórica*

aplicaciones que ofrezcan una fuerte interactividad con usuarios. Este patrón organiza la aplicación en tres modelos separados, el primero es un modelo que representa los datos de la aplicación y sus reglas de negocio, el segundo es un conjunto de vistas que representa los formularios de entrada y salida de información y el tercero es un grupo de controladores que procesa las peticiones de los usuarios y controla el flujo de ejecución del sistema.

Existen varios tipos de *frameworks web* dentro de los que se destacan CodeIgniter, Cake PHP, Yii, Zend Framework y Symfony, además están los orientados a la interfaz de usuario, como Java Server Faces, a aplicaciones de publicación de documentos, como Cocoon al control de eventos, como Struts y algunos que incluyen varios elementos como Tapestry, aunque la mayoría de estos se encargan de ofrecer una capa de controladores ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación ((22) (24) (25)).

Según la bibliografía analizada se arriba a la conclusión que un *framework web* asiste el trabajo de un programador en dos niveles: en el lado del cliente, ofreciendo funciones javascript para enviar peticiones al servidor y en el lado del servidor, el cual procesa las peticiones, busca información, y la transmite al navegador. Los *frameworks javascript* se ejecutan en el navegador y son ampliamente usados en el desarrollo de aplicaciones AJAX, siendo esto de gran importancia para el desarrollo de la aplicación y por ende para el cumplimiento de su objetivo general, así como la satisfacción de las necesidades del cliente.

1.5.1. Frameworks del lado del cliente

Existen varios *frameworks* del lado del cliente entre los que se destacan ExtJS el cual es una biblioteca que expande Prototype, YUI y jQuery el cual provee un *framework* y muchas otras utilidades, estos pueden liberar al programador de tener que tratar con las complejidades de bajo nivel de Ajax, o proporcionar *widgets*¹⁰ preexistentes como un *textbox*¹¹ con terminaciones accionadas por el mismo, además sirven para programar con lenguaje javascript de una manera rápida y compatible con todos los navegadores.

¹⁰ Widgets es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños.

¹¹ Representa un control que se puede utilizar para mostrar o modificar el texto sin formato.



1.5.1.1. jQuery 1.9

Según Aaron Newton jQuery es una biblioteca de Javascript rápida y concisa que simplifica el recorrido, manejo de eventos, animación e interacciones Ajax en el documento HTML para un rápido desarrollo web (26).

jQuery posee inconvenientes que no le permiten ser una opción factible para la aplicación tal es el caso que no ofrece un sistema de herencia ni ofrece ninguna mejora a los objetos nativos (Function, String, etc), da la imposibilidad de establecer ciertos *headers*¹² a los pedidos AJAX.

1.5.1.2. ExtJS 3.4

ExtJS es una librería javascript que permite construir aplicaciones complejas en Internet utilizando tecnologías como AJAX y DHTML. Esta librería incluye los componentes de interfaz de usuario de alto performance y personalizables, modelo de componentes extensibles, una Aplicación para la Interfaz de Programación (API) fácil de usar y licencias *código abierto* y comerciales.

Para la propuesta de solución se decide utilizar como *framework* del lado del cliente ExtJS debido a que posibilita crear aplicaciones complejas utilizando componentes predefinidos, proporcionando una experiencia consistente sobre cualquier navegador, evitando el tedioso problema de validar que el código escrito funcione bien en cada uno. Existe un balance entre Cliente – Servidor ya que la carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo. Posee una comunicación asíncrona donde en este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un comando o una acción del usuario, dándole la libertad de cargar información sin que el cliente se dé cuenta y permite además una gran eficiencia de la red debido a que el tráfico de red puede disminuir al permitir que la aplicación elija la información que desea transmitir al servidor y viceversa. (27)

1.5.2. Frameworks del lado del servidor

Existen varios *frameworks* del lado del servidor, para el desarrollo de la propuesta de solución se analizarán para la lógica del negocio los *frameworks* php Codeigniter y Symfony y para el acceso a datos Doctrine ORM.

¹² En programación, se suele denominar *Headers* (traducido al castellano como cabeceras), al conjunto de archivos fuente en los cuales se provee una interface para determinado tipo de dato.



1.5.2.1. Symfony 2.0

Symfony es un *framework* desarrollado en php 5 que posee desventajas que lo descartan para el desarrollo de la propuesta de solución tales como: su empinada curva de aprendizaje, muchos conceptos nuevos de golpe (si no se conoce otro *framework* MVC), gran consumo de memoria, búsqueda de datos un poco lenta, esto sucede cuando se desea realizar una búsqueda de datos muy específica, se debe hacer en forma manual ya que la utilización de la interfaces propel genera código que puede ser lento, la integración entre diferentes aplicaciones de un mismo proyecto todavía no se encuentra muy depurada, es decir se presentan varios inconvenientes si se desea utilizar un módulo que se encuentra en otra aplicación, posee muchos archivos de configuración, depende de algunas librerías internas. Además está pensado para el desarrollo de aplicaciones robustas y con muchas prestaciones no siendo este el entorno. Por todo lo antes planteado se decide rechazar symfony para el desarrollo de la solución (28).

1.5.2.2. CodeIgniter 2.1.3

Según Miguel Ángel Álvarez CodeIgniter es: “...un programa o aplicación web desarrollada en PHP para la creación de cualquier tipo de aplicación web bajo PHP. Es un producto de código libre, libre de uso para cualquier aplicación” (29).

Este *framework* contiene una serie de librerías que sirven para el desarrollo de aplicaciones web, marca una manera específica decodificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener, además está creado para que sea fácil de instalar en cualquier servidor.

Algunos de los puntos más interesantes de este *framework*, sobre todo en comparación con otros productos similares, son los siguientes (29):

- ❖ **Versatilidad:** CodeIgniter es capaz de trabajar la mayoría de los entornos o servidores.
- ❖ **Compatibilidad:** CodeIgniter, es compatible con la versión PHP 4 o superior siendo la PHP 4.3.2. la mínima versión a utilizar por dicho *framework*, lo que hace que se pueda utilizar en cualquier servidor, incluso en algunos antiguos.
- ❖ **Ligereza:** El núcleo de CodeIgniter es bastante ligero, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. La mayoría de los módulos o clases que ofrece se pueden cargar de manera opcional, sólo cuando se van a utilizar realmente.



- ❖ **Documentación tutorializada:** La documentación de CodeIgniter es fácil de seguir y de asimilar, porque está escrita en modo de tutorial. Esto facilita la referencia rápida.

Sin duda, lo más destacable de CodeIgniter es su accesibilidad. Por todo lo antes planteado es escogido este *framework* para el desarrollo de la aplicación.

1.5.2.3. Doctrine ORM 1.2

Doctrine es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un Sistema Gestor de Base de Datos (SGBD), es además una librería para PHP que permite trabajar con un esquema de base de datos como si fuese un conjunto de objetos, y no de tablas y registros.

Alguna de las características más significativas de Doctrine ORM son (30):

- ❖ **Generación automática del modelo:** Cuando se trabaja con ORM, es necesario crear el conjunto de clases que representa el modelo de la aplicación, luego estas clases serán vinculadas al esquema de la base de datos de forma automática con un motor ORM. Aunque son cosas diferentes, cuando se diseña un modelo relacional y un modelo de clases, suelen ser muy parecidos. Doctrine se aprovecha de esta similitud y permite generar de forma automática el modelo de clases basándose en el modelo relacional de tablas.
- ❖ **Posibilidades de trabajar con YAML¹³:** Como se comenta en el apartado anterior, Doctrine ORM puede generar de forma automática el modelo, pero también deja la posibilidad que se pueda definir por uno mismo el mapeo de tablas y sus relaciones. Esto se puede hacer con código PHP o con YAML que es un formato de serialización de datos legibles muy usado para este fin.
- ❖ **Buscadores mágicos (*Magic finders*):** En Doctrine ORM, se pueden buscar registros basándose en cualquier campo de una tabla. Es importante decir que existe el método `findAll()`, que obtiene todos los registros de una tabla.

¹³ YAML es un formato de serialización de datos legible inspirado en lenguajes como XML, C, Python, Perl, entre otros.



- ❖ **Relaciones entre entidades:** En Doctrine ORM una vez definido el modelo (o se ha creado de forma automática) con las tablas y sus relaciones, resulta fácil acceder y moverse por entidades relacionadas.

Por todo lo antes planteado es escogido este *framework* para el desarrollo de la propuesta de solución.

1.6. Metodologías de desarrollo de *software*

El análisis de la bibliografía describe que las metodologías de desarrollo de *software* son: “...un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos *software*” (31).

Son muchas las ventajas que puede aportar el uso de una metodología. Estas pueden ser agrupadas desde diferentes perspectivas, desde el punto de vista de gestión permite facilitar la tarea de planificación, la tarea del control y el seguimiento de un proyecto, desde el punto de vista de los ingenieros del *software* posibilita ayudar a la comprensión del problema, optimizar el conjunto, facilitar el mantenimiento del producto final y permitir la reutilización de partes del mismo y desde el punto de vista del cliente o usuario garantizar un determinado nivel de calidad en el producto final así como la confianza en los plazos de tiempo fijados en la definición de estos (32).

El análisis de la bibliografía consultada, muestra que existen dos tipos de metodologías de desarrollo del *software*, las metodologías robustas o tradicionales y las metodologías ágiles.

1.6.1. Metodologías tradicionales

Entre las metodologías robustas se encuentran, RUP (por sus siglas en inglés, Rational Unified Process), MSF (por sus siglas en inglés, Microsoft Solution Framework), MÉTRICA 3, entre otras. Estas metodologías son denominadas de forma peyorativa, como metodologías pesadas, se centran en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requieren una extensa documentación, ya que pretende prever todo de antemano. Estas son más eficaces y necesarias cuanto mayor es el proyecto que se pretende realizar respecto a tiempo y recursos que son necesarios emplear, donde una gran organización es requerida (33).



No se escogen estas metodologías para el desarrollo de la aplicación debido a que poseen cierta resistencia al cambio durante el desarrollo, estas fueron diseñadas para grandes proyectos, se centran en la definición detallada de los procesos, tareas y herramientas a utilizar; y por esto generan una extensa documentación. Además, al no demandar un contacto sistemático con el cliente son descartadas en la presente investigación y se propone el empleo de una metodología ágil.

1.6.2. Metodologías ágiles

Las metodologías ágiles promueven generalmente un proceso de gestión de proyectos que fomenta el trabajo en equipo, la organización y responsabilidad propia, un conjunto de mejores prácticas de ingeniería que permiten la entrega rápida de *software* de alta calidad, y un enfoque de negocio que alinea el desarrollo con las necesidades del cliente y los objetivos de la compañía. Entre las metodologías ágiles más notables están Scrum, Crystal Clear, XP (*Extreme Programming*), ASD (*Adaptative Software Development*), *Feature Driven Development* y DSDM (*Dynamic Systems Development method*) concibiéndose como metodologías ágiles desde que se publicó el manifiesto ágil en 2001 (32).

1.6.2. Scrum

*Scrum*¹⁴ es una metodología ágil que se puede usar para gestionar y controlar desarrollos complejos de *software* y productos usando prácticas iterativas e incrementales. Dicha metodología posee desventajas que no le permiten ser una opción factible para la propuesta de solución tales como: plantea un problema si el desarrollo está restringido por una fecha de entrega, presupone que los requerimientos cambian, pero no de forma que el cliente acepte un diseño funcional/técnico, se requiere de un experto en la metodología que monitorice su cumplimiento, es un inconveniente para personas con muy poca experiencia en el desarrollo de *software* y no posee ningún tipo de prácticas de ingeniería (34).

¹⁴Término que hace referencia a como se devuelve un balón que ha salido fuera del campo al terreno de juego de una manera colectiva.



Figura 5: Ciclo de trabajo de Scrum.

Disponible en: www.publicaciones.urbe.edu

1.6.2.2. Extreme programming (XP)

La programación extrema (XP) es un enfoque de la ingeniería del *software* formulado por Kent Beck. En 1996 este expresó: “*La programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad*”.

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de *software*, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo, se basa en cuatro principios fundamentales, la realimentación continua entre el cliente y el equipo de desarrollo, la comunicación fluida entre todos los participantes, la simplicidad en las soluciones implementadas y el coraje para enfrentar los cambios. También se define como adecuada para proyectos con requisitos imprecisos y donde existe un alto riesgo técnico. Esta metodología consta de doce prácticas fundamentales destacando entregas pequeñas, diseño simple, programación en parejas, pruebas e integración continua donde el mayor beneficio de las prácticas se consigue con su aplicación conjunta y equilibrada puesto que se apoyan unas en otras, (32) además se realizan pruebas continuas durante el proyecto y es mejor utilizada en la implementación de

nuevas tecnologías donde los requerimientos cambian rápidamente y el cliente tiene el control sobre las prioridades (35).

XP cuenta con las siguientes seis fases: exploración, en la cual los clientes plantean sus necesidades y el equipo de desarrollo revisa las tecnologías, prácticas y herramientas que van a ser utilizadas durante el proyecto, planificación donde los programadores y clientes se ponen de acuerdo para priorizar las historias de usuario y el alcance de la primera versión, desarrollo o iteración hacia la primera versión, aquí el cliente decide las historias de usuario (HU) que se realizarán dividiéndose en tareas que serán desarrolladas en la siguiente versión, en la fase de producción se deciden los errores, las nuevas funcionalidades o las modificaciones, la de mantenimiento implica iteraciones más largas, incorpora nuevos desarrolladores e incluso reestructura el equipo de desarrollo y la última fase denominada cierre del proyecto finaliza cuando el cliente no tenga más historias de usuario para ser incluidas en el sistema (36).

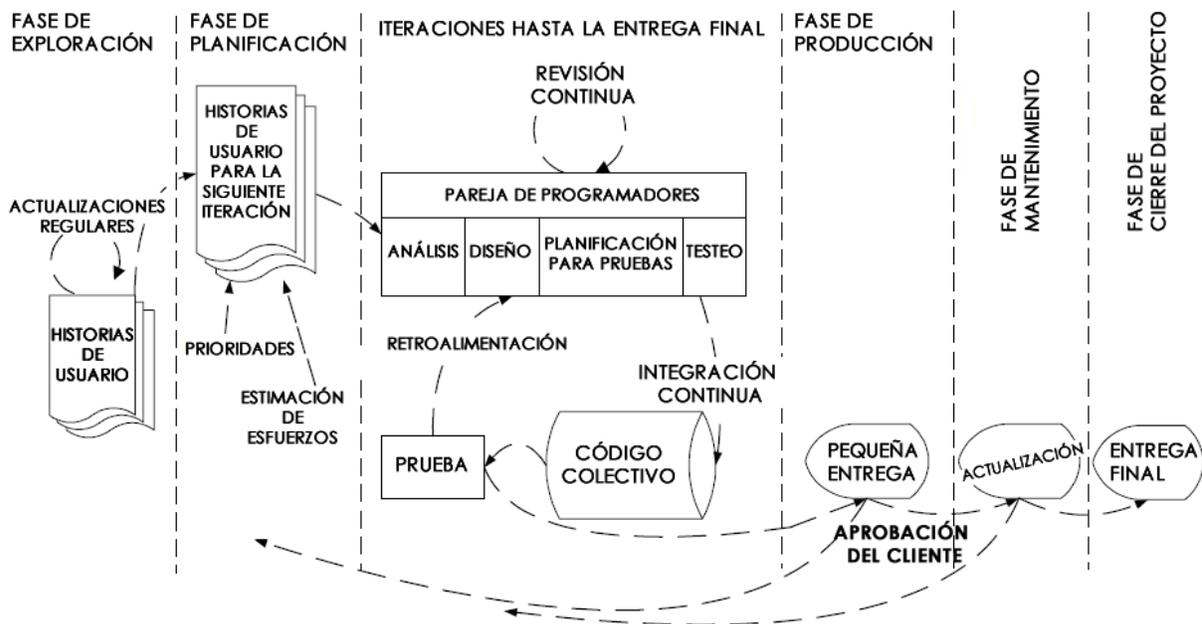


Figura 6: Fases del ciclo de vida de la metodología XP.

Además según Billy Reinoso en una conferencia sobre metodologías ágiles de desarrollo de *software*, plantea que XP ha demostrado ser la metodología ágil que más estudios dispone y la de mayor número de artículos escritos (37).



Dada las condiciones, facilidades que brinda y la idea de desarrollo que se tiene de la solución, las autoras acordaron que las características de la metodología de desarrollo de *software* XP se asocian más al tipo de proyecto que se lleva a cabo durante la construcción de la solución informática ya que es empleada para proyectos de corto plazo, con un equipo de trabajo pequeño, propone una realimentación continua entre el cliente y el equipo de desarrollo, define un modelo de 40 horas semanales para no trabajar horas extras, entre otros aspectos mencionados con anterioridad.

1.7. Sistemas gestores de bases de datos

La bibliografía consultada permite definir para esta investigación un sistema gestor de base de datos (SGBD, en inglés DBMS: *DataBase Management System*) como: “... un sistema de *software* que permite la definición de bases de datos; así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación” (38).

En otras palabras, un SGBD relacional es un modelo de datos que facilita a los usuarios describir los datos que serán almacenados en la base de datos vinculado con un conjunto de operaciones para manejar los mismos. Estos son una herramienta efectiva que permite a varios usuarios acceder a los datos al mismo tiempo, ofrecen facilidades eficientes y un grupo de funciones con el objetivo de garantizar la confidencialidad, la calidad, la agilidad, la seguridad y la integridad de los datos que contienen.

Al ser escogido con anterioridad el *framework* CodeIgniter se tiene que este es compatible con algunas bases de datos dentro de las que se encuentran: MySQL (4.1 o posterior), MySQLi, MS SQL, Postgres, Oracle, SQLite.

1.7.1 MySQL 5.6.9

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. El objetivo que persigue la empresa de código abierto MySQL AB consiste en que el mismo cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad. Dicho SGBD posee desventajas que lo descartan para la propuesta de solución como el hecho de que es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

1.7.2. PostgreSQL 9.1



Capítulo 1 *Fundamentación teórica*

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos y libre surgido en 1986 con el lanzamiento de su primera versión y es publicado bajo la licencia BSD¹⁵; es un potente motor de bases de datos, que tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales y permite métodos almacenados, restricciones de integridad, vistas, entre otras (39).

Algunas de las características más significativas de PostgreSQL son ((40) (41)):

- ❖ PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, optimización de consultas, herencia y *arrays*¹⁶ siendo de gran importancia para el trabajo con la aplicación.
- ❖ Soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- ❖ Incluye características avanzadas tales como las uniones (*joins*).
- ❖ Soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- ❖ Posee claves ajenas también denominadas llaves ajenas o claves foráneas (*foreignkeys*).
- ❖ Incorpora disparadores (*triggers*): Un disparador o *trigger* se define en una acción específica basada en algo ocurrente dentro de la base de datos. En PostgreSQL esto significa la ejecución de un procedimiento almacenado basado en una determinada acción sobre una tabla específica.

Por las características que presenta PostgreSQL se decide utilizar como SGBD para la propuesta de solución ya que está considerado como la base de datos de código abierto y con orientación a objetos más avanzada del mundo porque proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2, Oracle o SQLServer. Además puede operar sobre distintas plataformas, como Linux,

¹⁵ licencia BSD (Berkeley Software Distribution), pertenece al grupo de licencias de software Libre pero tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público.

¹⁶ Un *array* es una colección ordenada de elementos de un mismo tipo de datos, agrupados de forma consecutiva en memoria. Cada elemento del *array* tiene asociado un índice, que no es más que un número natural que lo identifica inequívocamente y permite al programador acceder a él.



UNIX, AIX, BSD, HP-UX, Mac OS X, Solaris y Windows, tiene buena seguridad gracias a la correcta gestión de usuarios, grupos de usuarios y contraseñas, así como los permisos asignados a cada uno de ellos mediante sentencias SQL, incorpora una estructura de datos *array* y permite la declaración de funciones propias.

1.8. Entorno de desarrollo integrado

Un entorno de desarrollo integrado (IDE) (también conocido como entorno de diseño integrado, entorno integrado de depuración o entorno de desarrollo interactivo) “...es una aplicación de software que proporciona servicios integrales a los programadores de computadoras para el desarrollo de software” (42).

Un IDE normalmente se compone de un editor de código fuente, un compilador y / o un intérprete, una automatización de generación de herramientas y un depurador. Estos proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación, tales como C++, Java, C#, Basic, Object Pascal, entre otros (43).

Algunos ejemplos de entornos integrados de desarrollo (IDE) son Eclipse, NetBeans, MS Visual Studio .NET de Microsoft, Delphi de Borland, JDeveloper de Oracle, Aptana, KDevelop, Zend Studio, entre otros.

1.8.1. Zend studio 9.0.1

Zend Studio es un ambiente integrado de desarrollo (IDE PHP) disponible para los desarrolladores profesionales que ofrecen las capacidades necesarias para desarrollar aplicaciones de negocio, posee características como refactorización, generación de código, asistente de código y análisis semántico (44), sin embargo es descartado por ser *software* propietario.

1.8.2. Netbeans 7.3

Netbeans es un entorno de desarrollo integrado de código abierto, libre, y tiene una comunidad mundial de usuarios y desarrolladores. Posee multilenguaje completo y modular, incorpora gran cantidad de módulos de terceros (*plugins*), crea ventanas, menús, barras de herramientas y acciones, además de que permite la utilización de varios lenguajes, incluido *PHP*. Netbeans en su versión 7.3 aumenta el rendimiento y la experiencia de programar, escanea de forma inteligente el proyecto para corregir cualquier tipo de fallo ((45) (46)).



Existen otros IDEs de desarrollo como Aptana que posee características similares al netbeans aunque consume menor cantidad de recursos, sin embargo no es escogido para el desarrollo de la propuesta de solución debido a que el equipo de desarrollo poseía una base del trabajo con netbeans y el mismo cumple con las características necesarias para el desarrollo del sistema que se desea realizar. Por todo lo antes mencionado se decide utilizar netbeans como IDE para la propuesta de solución.

1.9. Servidores web

Un servidor web se define como *“...un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de internet. El servidor web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados”* (47).

1.9.1 Servidor IIS 7.0

Según la bibliografía consultada los servicios de *Internet Information Server* (o IIS) son *“...los servicios de software que admiten la creación, configuración y administración de sitios Web, además de otras funciones de Internet. Los servicios de IIS incluyen el protocolo de transferencia de noticias a través de la red (NNTP), el protocolo de transferencia de archivos (FTP) y el protocolo simple de transferencia de correo (SMTP)”* (48).

Este servidor posee desventajas tales como: cantidad de fallos de seguridad, la mayoría de las funcionalidades extras deben ser comprada separadamente, sólo funciona en Windows NT/2000; es decir no es multiplataforma y tiende a limitarse en las versiones que no son de la familia “Server” (49). Por todo lo antes planteado es descartado este servidor para el desarrollo de la aplicación.

1.9.2. Apache 2.2.22

El **Servidor HTTP Apache** es *“...un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual”* (50).



Capítulo 1 *Fundamentación teórica*

Apache presenta un conjunto de características que le permiten ser un servidor seguro, eficiente y extensible que proporcione servicios HTTP en sincronización con los estándares HTTP actuales. A continuación se exponen algunas de ellas (51) (52):

- ❖ Es multiplataforma, lo que lo hace prácticamente universal.
- ❖ Es un servidor de web conforme al protocolo HTTP/1.1.
- ❖ Es una tecnología gratuita de código fuente abierta.
- ❖ Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y permite su instalación cuando es necesario.
- ❖ Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.
- ❖ Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.
- ❖ Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- ❖ Tiene una alta configurabilidad en la creación y gestión de *logs*¹⁷. Permite la creación de ficheros de *log* a medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

El servidor web a utilizar para el desarrollo de la aplicación será Apache ya que su código fuente es abierto, su arquitectura modular, construida sobre un pequeño núcleo, se adapta a las necesidades específicas de cada usuario. Trabaja con gran cantidad de código PHP teniendo todo el soporte que se necesita para tener páginas dinámicas, es un servidor web excelente, por su configurabilidad y robustez, además es casi universal al estar en una multitud de sistemas operativos.

¹⁷Los logs son usados para registrar datos o información.



Capítulo **1** *Fundamentación teórica*

Conclusiones

Los métodos científicos y las técnicas de recopilación de datos empleadas en el estudio que se presenta del capítulo I, permitieron desarrollar la teoría que sustenta esta investigación, además posibilitaron realizar un estudio detallado de las herramientas, tecnologías, metodologías y lenguajes de programación existentes para el desarrollo de una herramienta web, siendo seleccionadas las opciones más factibles para dar cumplimiento al objetivo general.



Capítulo 2: Propuesta de solución

2.1. Introducción

El presente capítulo detalla las fases de Exploración y Planeación definidas en el ciclo de vida de la metodología de desarrollo XP que posibilita describir la solución propuesta.

De igual forma se muestran los principios, prácticas y técnicas que sirven de guía para los desarrolladores, entre las que se destacan: HU, plan de iteraciones, plan de entrega y las tarjetas CRC (Clases, Responsabilidad y Colaboración).

Uno de los pasos fundamentales en la fase de Exploración de la metodología XP es definir una metáfora del sistema. En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora. El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo (53).

Una metáfora es una historia compartida que describe cómo debería funcionar el sistema. Martin Fowler en 2001 explica que la práctica de la metáfora consiste en: “... un conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema. Este conjunto de nombres ayuda a la nomenclatura de clases y métodos del sistema” (54).

Para lo siguiente, la metáfora que se propone de la solución que se quiere alcanzar es: “un espacio donde se ‘recolecte’ información relevante en el terreno por parte de los expertos y garantice la conservación de la misma.”

2.2. Sistemas de control de acceso

Uno de los elementos esenciales en el desarrollo de la solución es definir el control de acceso de los usuarios a la aplicación, para determinar si el usuario tiene permitido el acceso a los recursos existen modelos de control de acceso como el control de acceso discrecional (DAC), el control de acceso obligatorio (MAC) y el control de acceso basado en roles (RBAC), donde DAC y MAC por sí solos son inadecuados para cubrir las necesidades de la mayor parte de las organizaciones. El modelo DAC es demasiado débil para controlar el acceso a los recursos de información de forma efectiva, en tanto que el MAC es demasiado rígido. El modelo RBAC, es un intento de unificar los modelos anteriores, consiguiendo un sistema que impone el control de



accesos, pero sin las restricciones rígidas impuestas por las etiquetas de seguridad, para esta investigación se definió el RBAC.

2.2.1. RBAC

El análisis de la bibliografía describe que un RBAC básicamente consiste en la creación de roles para los trabajos o funciones que se realizan en la organización, su arquitectura está ideada para asignar permisos a operaciones más que a objetos concretos, siendo por tanto su principal ventaja el permitir alinear la infraestructura de seguridad de la información con los objetivos de negocio de una forma natural. Este recopila además las capacidades de superusuario (usuario root) en perfiles de derechos, los mismos se asignan a cuentas de usuario especiales denominadas roles. Luego, un usuario puede asumir un rol para realizar un trabajo que requiere algunas de las capacidades de superusuario.

RBAC actualmente es considerado uno de los modelos más generales, debido a su neutralidad respecto a las políticas de control de acceso y a su flexibilidad. A causa de la existencia de múltiples variantes de modelos e implementaciones del mismo se destacan las siguientes características (55):

- ❖ Administración de autorizaciones
 - La asignación de permisos a usuarios tiene dos partes: asociar usuarios a roles y asignar permisos para objetos a roles.
 - Si un usuario cambia de tareas, solo basta con cambiarle el rol.
- ❖ Jerarquía de roles
 - Los roles también poseen relaciones de jerarquía.
 - Pueden heredar privilegios y permisos de otros roles de menor jerarquía, simplificando la administración de las autorizaciones.
- ❖ Menor privilegio
 - Permite implementar la política del menor privilegio posible la cual consiste en que un usuario dispone exactamente de la cantidad de privilegios necesaria para realizar un trabajo.
 - Si una tarea no va a ser ejecutada por un usuario, entonces su rol no tendrá los permisos para hacerla, de esta manera se minimizan riesgos de daños.



- ❖ Separación de responsabilidades
 - Se basa en el principio de que ningún usuario tenga suficientes privilegios para usar el sistema en su propio beneficio.

2.3. Historias de usuario

Las HU son la técnica utilizada en XP para especificar los requisitos del *software*. Se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las HU es muy dinámico y flexible, en cualquier momento las mismas pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (56). XP propone fichas para registrar las mismas y también tareas asociadas a ellas, sin dar mayores guías respecto a la información que debe ser recolectada ni cómo debe de gestionarse. Por otra parte, además de la restricción máxima de esfuerzo asociado (3 semanas de programación) no hay más pautas respecto a la granularidad de una historia de usuario, con lo cual, tratándose de requisitos funcionales y no funcionales, el número puede ser considerable incluso para sistemas pequeños. (57) A continuación se muestra el formato general de una HU propuesto por R. Jeffries, A. Anderson y C. Hendrickson en *Extreme Programming Installed* y las HU que representan las funcionalidades críticas que fueron implementadas para la propuesta de solución. En los anexos (Ver Anexo I) se encuentran las HU de las restantes funcionalidades junto con las características del sistema:

Tabla 1: Formato general de las HU

Historia de Usuario	
No. Número consecutivo a partir del 1.	Nombre: Identifica la HU.
Usuario: ¿Quién ejecuta la HU?	
Prioridad de Negocio: Define la relevancia e impacto de la historia de usuario para el negocio de acuerdo con las necesidades del	Puntos Estimados: Permite estimar la duración de la implementación, representando



Capítulo 2 *Propuesta de solución*

usuario. Nivel de Complejidad: Define la dificultad técnica que supone desarrollar la historia de usuario desde el punto de vista del programador.	con 1, una semana de trabajo. Iteración Asignada: Precisa la iteración en la que será desarrollada la HU.
Descripción: Explica en qué consiste la HU, teniendo en cuenta las acciones realizadas por el usuario y la respuesta brindada por el sistema.	
Observaciones: Brinda información extra que se estime agregar para hacer más comprensible la HU. Por ejemplo: conceptos, post-condiciones, relación con otros requisitos, entre otras.	

Tabla 2: HU de Gestionar Inspección

Historia de Usuario	
No. 11	Nombre: Gestionar Inspección
Usuario: Jefe de grupo de Inspección	
Prioridad de Negocio: Alta	Puntos Estimados: 1
Nivel de Complejidad: Alta	Iteración Asignada: 1
Descripción: Permite crear, modificar y eliminar las inspecciones en la BD. En caso de agregar estos permisos lo asigna el rol que posea permisos de	



Jefe de grupo de Inspección.

Observaciones: Debe haberse realizado la HU 10 con anterioridad, es decir debe haberse planificado la inspección en el cronograma.

Tabla 3: HU de Gestionar Licencia Ambiental

Historia de Usuario	
No. 14	Nombre: Gestionar Licencia Ambiental
Usuario: Director	
Prioridad de Negocio: Alta	Puntos Estimados: 1.5
Nivel de Complejidad: Alta	Iteración Asignada: 2
Descripción: Permite crear, modificar y eliminar las licencias ambientales en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de director.	
Observaciones: Debe haberse realizado la HU 15 con anterioridad.	

2.4. Estimación de esfuerzos por HU

La estimación de esfuerzos es establecida por los programadores utilizando las semanas como medida. Las HU deben ser programadas en un tiempo estimado hasta 3 semanas. Si la estimación supera las 3 semanas la HU deberá ser dividida hasta que pueda ser desarrollada en un tiempo factible. En caso de que el esfuerzo sea menor de 1 semana la HU será combinada con otra.

Tabla 4: Estimación de esfuerzos por historia de usuario

No.	Historia de Usuario	Estimación
-----	---------------------	------------



Capítulo 2 *Propuesta de solución*

		(semanas)
1	Gestionar Usuario	0.5
2	Gestionar Rol	1
3	Autenticar Usuario	0.5
4	Gestionar Funcionalidades	1
5	Gestionar Empresa	0.5
6	Gestionar Persona	0.5
7	Gestionar DPT ¹⁸	0.5
8	Gestionar Cargo	0.3
9	Gestionar Legislación	0.7
10	Gestionar Cronograma	1
11	Gestionar Inspección	1
12	Generar Reportes	2
13	Gestionar Contravenciones	0.5
14	Gestionar Licencia Ambiental	1.5
15	Gestionar Dictámenes	0.5

2.5. Plan de iteraciones

El plan de iteración se compone de las historias de usuario definidas en el pan de entrega, más las pruebas de aceptación que hayan fallado en el ciclo anterior los cuales son analizados

¹⁸ DPT: División Política Territorial



Capítulo 2 *Propuesta de solución*

para evaluar su corrección, así como para prever que no vuelvan a ocurrir, dichas iteraciones poseen una duración de 1 a 3 semanas y no permiten realizar ninguna tarea no programada para la iteración.

Iteración 1: En la primera iteración se entregarán algunas de las funcionalidades que tienen prioridad alta para el desarrollo de la aplicación, correspondiendo a las HU:

- ❖ HU 1: Gestionar Usuario.
- ❖ HU 2: Gestionar Rol.
- ❖ HU 3: Autenticar Usuario.
- ❖ HU 4: Gestionar Funcionalidades.

Iteración 2: En esta iteración se realizarán las HU con prioridad alta para el cliente siendo estas:

- ❖ HU 5: Gestionar Empresa.
- ❖ HU 6: Gestionar Persona.
- ❖ HU 10: Gestionar Cronograma.
- ❖ HU 11: Gestionar Inspección.

Iteración 3: En esta iteración se realizarán las restantes HU que son importantes para el cliente siendo estas:

- ❖ HU 7: Gestionar DPT.
- ❖ HU 8: Gestionar Cargo.
- ❖ HU 9: Gestionar Legislación.
- ❖ HU 14: Gestionar Licencia Ambiental.

Iteración 4: En esta iteración se implementan las HU de baja prioridad para el cliente pero no menos importante que las anteriores para los desarrolladores, las que se definen por:

- ❖ HU 12: Generar Reportes.
- ❖ HU 13: Gestionar Contravenciones.
- ❖ HU 15: Gestionar Dictámenes.



2.6. Plan de duración de las iteraciones

El plan de duración de las iteraciones se realiza luego de tener el estimado en días que demora implementar cada HU. Se tendrá en cuenta la prioridad que el cliente le asigna a cada historia y el nivel de complejidad que estas poseen.

Tabla 5: Plan de duración de las iteraciones

Iteración	Orden de las HU	Duración total
1	Gestionar Usuario Gestionar Rol Autenticar Usuario Gestionar Funcionalidades	3 semanas
2	Gestionar Empresa Gestionar Persona Gestionar Cronograma Gestionar Inspección	3 semanas
3	Gestionar DPT Gestionar Cargo Gestionar Legislación Gestionar Licencia Ambiental	3 semanas
4	Generar Reportes Gestionar Contravenciones Gestionar Dictámenes	3 semanas



2.7. Plan de entrega

El plan de entrega es un plan dividido en varios planes de iteración donde en la primera iteración se obtiene el sistema con toda su arquitectura, en dicho plan el usuario selecciona las HU en cada iteración y las pruebas funcionales de estas historias son validadas al final de la misma, además el cronograma de entregas establece qué HU serán agrupadas para conformar una entrega, y su orden; este cronograma será el resultado de una reunión entre todos los actores del grupo de desarrollo incluyendo el cliente permitiendo alcanzar un mayor entendimiento en la implementación del sistema.

Tabla 6: Plan de Entrega

Historia de Usuario	Primera Iteración	Segunda Iteración	Tercera Iteración	Cuarta Iteración
Gestionar Usuario	V 1.0	Finalizado	Finalizado	Finalizado
Gestionar Rol	V 1.0	Finalizado	Finalizado	Finalizado
Autenticar Usuario	V 1.0	Finalizado	Finalizado	Finalizado
Gestionar Funcionalidades	V 1.0	Finalizado	Finalizado	Finalizado
Gestionar Empresa	-	V 1.0	Finalizado	Finalizado
Gestionar Persona	-	V 1.0	Finalizado	Finalizado
Gestionar Cronograma	-	V 1.0	Finalizado	Finalizado
Gestionar Inspección	-	V 1.0	Finalizado	Finalizado
Gestionar DPT	-	-	V 1.0	Finalizado
Gestionar Cargo	-	-	V 1.0	Finalizado
Gestionar Legislación	-	-	V 1.0	Finalizado



Capítulo 2 Propuesta de solución

Gestionar Licencia Ambiental	-	-	V 1.0	Finalizado
Generar Reportes	-	-	-	V 1.0
Gestionar Contravenciones	-	-	-	V 1.0
Gestionar Dictámenes	-	-	-	V 1.0

2.8. Prototipo de interfaz de usuario no funcional

Un prototipo en *software*: “...es un modelo del comportamiento del sistema que puede ser usado para entenderlo completamente o ciertos aspectos de él y así clarificar los requerimientos... un prototipo es una representación de un sistema, aunque no es un sistema completo, posee las características del sistema final o parte de ellas.” (58)

Para mostrar una vista preliminar del sistema a desarrollar se crearon prototipos de interfaz de usuario no funcionales con apoyo de la herramienta Balsamiq Mockups, permitiendo al cliente garantizar el entendimiento de los requisitos del sistema y poder comenzar con la implementación del mismo. A continuación se muestra el prototipo de interfaz de usuario no funcional de la pantalla de inicio del sistema, para consultar los restantes prototipos (ver Anexo II).

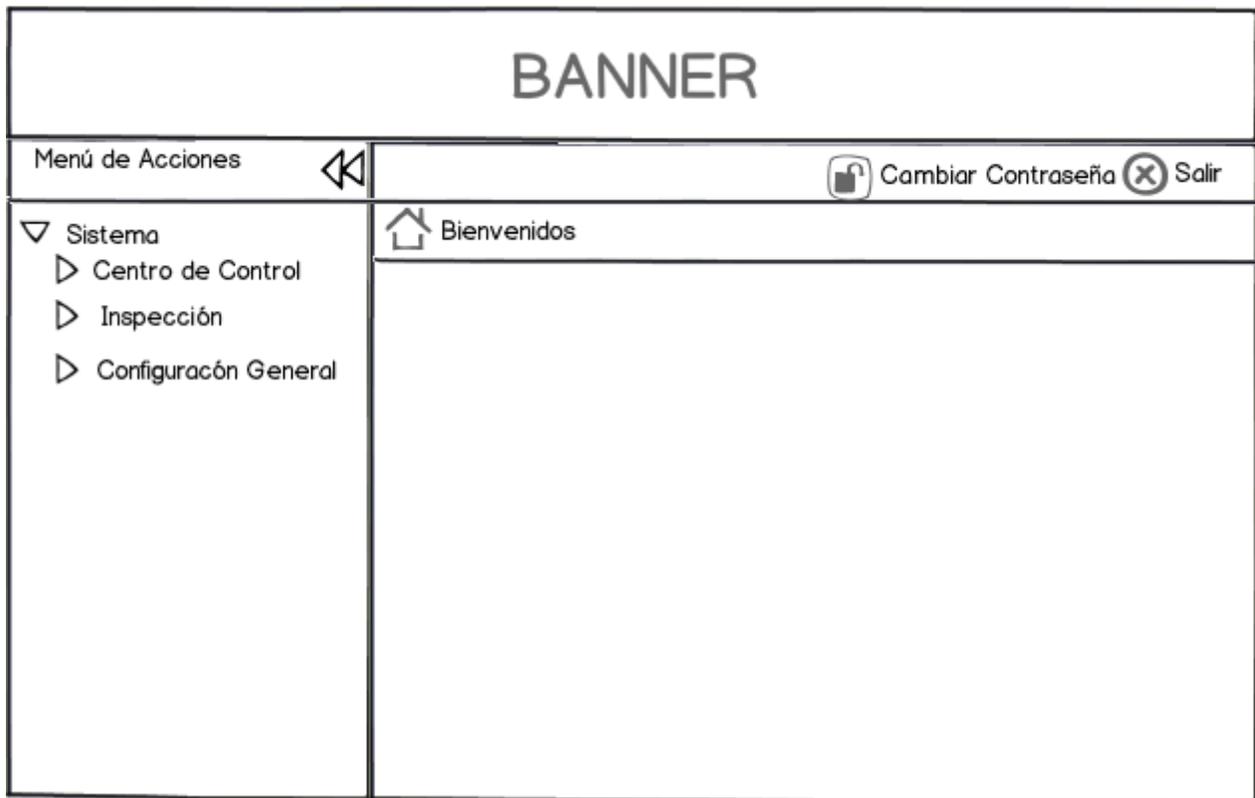


Figura 7: Prototipo de interfaz de usuario no funcional de la página principal.

2.9. Tarjetas CRC

Según la bibliografía consultada XP estimula el uso de tarjetas CRC como un mecanismo eficaz para pensar en el *software* en un contexto orientado a objetos, creadas por Kent Beck y Ward Cunningham, las cuales identifican y organizan las clases orientadas a objetos, siendo este el único producto de trabajo de diseño que se generan como parte del proceso XP. Las tarjetas están compuestas por el nombre de la clase colocado como título, en la parte izquierda se colocan las responsabilidades (funcionalidades) y en la parte derecha las clases que se implican en cada responsabilidad. (59)

En Programación Orientada a Objetos (POO), cada tarjeta representa una clase y define sus responsabilidades (lo que hace) y las colaboraciones con las otras clases (cómo se comunica con ellas), por lo que frecuentemente cada tarjeta CRC se convierte en un objeto, sus responsabilidades en métodos públicos, y sus colaboradores en llamadas a otras clases. Por



esta razón es posible afirmar que las mismas se utilizan para llevar la realidad al modelo orientado a objetos (como si fuera un diagrama UML). (56)

Tabla 7: Tarjeta CRC Inspeccionmodel

Inspeccionmodel	
Responsabilidades	Colaboradores
✓ __construct	✓
✓ saveDocumentoPadre ✓ salvarDocumento	✓ Doctrine
✓ eliminarDocumento	✓ Doctrine_Query
✓ cargarPlanes	✓ DatPlaninspeccion

Tabla 8: Tarjeta CRC Inspeccion.plan.Cronograma.Vista

Inspeccion.plan.Cronograma.Vista	
Responsabilidades	Colaboradores
✓ Init	✓ Xtiaris.Lib.Util ✓ Ext.Button ✓ Ext.grid.RowSelectionModel ✓ Ext.grid.ColumnModel ✓ Ext.grid.RowNumberer ✓ Ext.grid.GridPanel ✓ Ext.grid.GroupingView ✓ Ext.PagingToolbar ✓ Ext.Viewport



✓ ventanaPlan	✓ Ext.Button
	✓ Ext.Window
	✓ Ext.form.FormPanel

Tabla 9: Tarjeta CRC Inspeccion

Inspeccion	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ __construct ✓ gestplan ✓ Devolverdpt ✓ Adicionarplan ✓ Modificarplan ✓ Eliminardocumento 	✓
<ul style="list-style-type: none"> ✓ Cargarplanes 	✓ stdClass

Conclusiones

La metodología seleccionada para el desarrollo de las funcionalidades del sistema de gestión de información posibilitó realizar la propuesta del sistema y describirla, y permitió además diseñar el prototipo de interfaz de usuario no funcional de la solución. Los métodos empíricos posibilitaron, junto a los clientes realizar el levantamiento de las HU y su prioridad, así como el orden de su implementación, en una visión futura de las funcionalidades a implementar y se instituyó el plan de entrega del proyecto, determinando un cronograma en conjunto con el cliente que especifica las entregas que deben hacerse.



Capítulo 3: Implementación y prueba

3.1. Introducción

Este capítulo se enmarca en las fases de Iteración y Producción en el cual la programación es guiada por parejas, una de las buenas prácticas que propone la metodología seleccionada. En la fase de iteración al finalizar la misma, el cliente habrá realizado las pruebas funcionales para asegurarse de que todo funcione correctamente, en caso de existir errores, éstos se solucionarán en las siguientes iteraciones, mientras que en la fase de producción las iteraciones pueden disminuir su duración para aumentar la velocidad y el ritmo con el que se converge al final de la versión, siendo el riesgo el elemento más importante en estos casos.

A continuación se abordarán los elementos esenciales relacionados con la implementación del sistema, los estándares de codificación y los patrones de diseño aplicados. Además, se muestran las pruebas de aceptación diseñadas para cada HU que se proponen en la fase de iteración.

3.2. Estructura de CodeIgniter

Según la bibliografía analizada CodeIgniter posee una estructura principal compuesta por un archivo `index.php` el cual permite que se inicie la aplicación web y tres carpetas adicionales tal cual se muestra en la siguiente imagen:

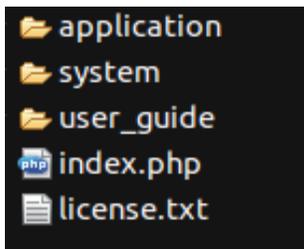


Figura 8: Estructura general de CodeIgniter.

En la carpeta “`user_guide`” se encuentra la documentación completa del *framework*, en la carpeta “`system`” está el núcleo del *framework* y la carpeta “`application`” donde se localizan las funcionalidades implementadas (60).

Para el desarrollo de la propuesta de solución y poseer así una mayor seguridad en la aplicación se modificó la estructura general de CodeIgniter. Tal como se muestra en la siguiente imagen:

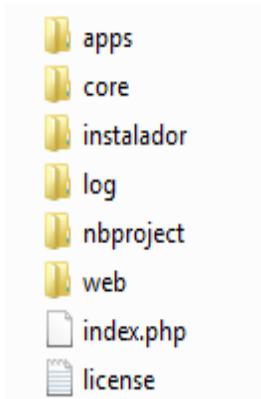


Figura 9: Estructura de la aplicación.

Mediante el estudio bibliográfico se pudo constatar que en codeIgniter la página index.php sirve como el controlador frontal, inicializando los recursos básicos que necesita el mismo para ejecutar las acciones, el ruteador examina la solicitud HTTP para determinar que debería hacer con ella, donde si existe el archivo de caché, se lo envía directamente al navegador, sin pasar por la ejecución normal del sistema, antes que se cargue el controlador de la aplicación, por razones de seguridad se filtran la solicitud HTTP y cualquier otro dato enviado por los usuarios, el controlador carga el modelo, las bibliotecas del núcleo y cualquier otro recurso requerido para procesar una solicitud específica y la vista terminada se procesa y se envía al navegador para que se pueda ver, si el caché está habilitado, se limpia primero la vista para que las siguientes solicitudes que la necesiten puedan ser servidas. (29)

El siguiente gráfico ilustra como los datos fluyen a través del sistema:

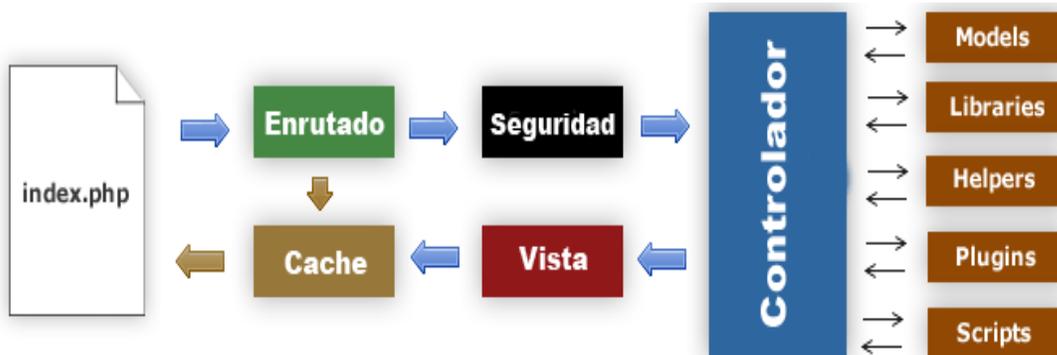


Figura 10: Flujo de datos de codeIgniter.



3.3. Estructura de la carpeta apps

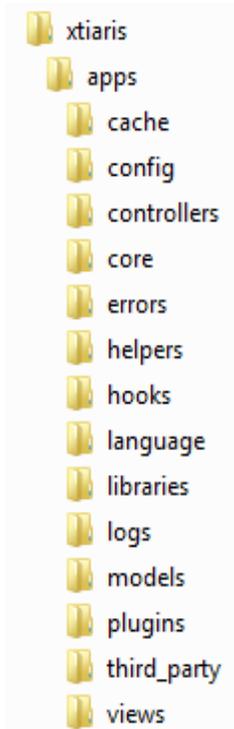


Figura 11: Estructura de la carpeta apps.

3.4. Módulos de la aplicación

La propuesta de solución se divide en módulos, los cuales tienen como función definir las características de la aplicación a la que pertenecen. El sistema implementado está compuesto por los módulos centro de control o seguridad, configuración general, inspección y licencia ambiental. A continuación se muestra la carpeta web en la cual se evidencian los módulos de la aplicación, los ficheros de estilo, los archivos javascript y las imágenes utilizadas. Su estructura es la siguiente:

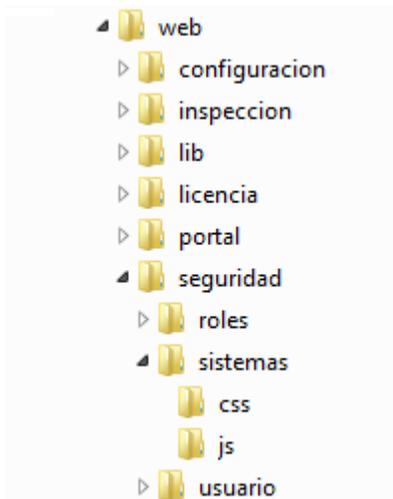


Figura 12: Estructura de la carpeta web.

3.5. Tareas de ingeniería

En la planificación de la iteración según la metodología XP se definen las actividades para las siguientes tres o cuatro semanas, aquí el cliente elige las HU de mayor valor para ser implementadas en la iteración planeada y a continuación los programadores dividen las mismas en tareas más pequeñas, denominadas tareas de ingeniería y eligen las tareas que desean implementar, las analizan en mayor detalle y realizan una estimación de su tiempo de desarrollo. Finalmente, el cliente ordena en función de sus necesidades las HU estimadas, dejando para iteraciones posteriores aquellas que sobrepasan la capacidad productiva de la iteración. (61) A continuación se muestra el formato general de una tarea de ingeniería (57) y las tareas de ingeniería que representan a las funcionalidades críticas que fueron implementadas para la propuesta de solución:



Capítulo 3 Implementación y prueba

Tabla 10: Formato general de la tarea de ingeniería

Tarea	
Número tarea: Número consecutivo a partir del 1.	Número historia: Identifica la HU
Nombre tarea:	
Tipo de tarea: Desarrollo/ Corrección/ Mejora/ Otra	Puntos estimados: Permite estimar la duración de la implementación, representando en días.
Fecha inicio: Comienzo de la tarea	Fecha fin: Fin de la tarea
Programador responsable: Persona encargada de la realización de la tarea de ingeniería	
Descripción: Explica en qué consiste la tarea de ingeniería.	

Tabla 11: Tarea de ingeniería adicionar inspección

Tarea	
Número tarea: 1	Número historia: 11
Nombre tarea: Adicionar inspección	
Tipo de tarea: Desarrollo	Puntos estimados: 2



Capítulo 3 Implementación y prueba

Fecha inicio: 20/05/2013

Fecha fin: 21/05/2013

Programador responsable: Darianna Dominico Brito y Deilys Lázara Ulloa Martínez

Descripción: Se puede adicionar una inspección en el módulo de inspección dándole al botón Adicionar inspección, a continuación aparece un formulario para llenar los datos referente a la inspección y al seleccionar el botón aceptar que se localiza al final del formulario se inserta una nueva inspección en la base de datos.

Tabla 12: Tarea de ingeniería modificar inspección

Tarea	
Número tarea: 2	Número historia: 11
Nombre tarea: Modificar inspección	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha inicio: 22/05/2013	Fecha fin: 24/05/2013
Programador responsable: Darianna Dominico Brito y Deilys Lázara Ulloa Martínez	
Descripción: Se puede modificar una inspección en el módulo de inspección dándole al botón Modificar inspección, a continuación aparece un formulario con los datos que posee la inspección hasta el momento, se seleccionan los datos a modificar y al terminar se acciona el botón aceptar que se localiza al final del formulario y se modifica la inspección en la base de	



datos.

Tabla 13: Tarea de ingeniería eliminar inspección

Tarea	
Número tarea: 3	Número historia: 11
Nombre tarea: Eliminar inspección	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha inicio: 22/05/2013	Fecha fin: 24/05/2013
Programador responsable: Darianna Dominico Brito y Deilys Lázara Ulloa Martínez	
Descripción: Se puede eliminar una inspección en el módulo de inspección dándole al botón Eliminar inspección, a continuación aparece un mensaje de confirmación de accionar el botón aceptar, se eliminará la inspección en la base de datos.	

3.6. Patrones de diseño usados en la solución

Según la bibliografía consultada Christopher Alexander en el año 1977 expresó que: “...un patrón describe un problema que ocurre una y otra vez en nuestro entorno, y describe la esencia de la solución a ese problema, de tal modo que pueda utilizarse esta solución un millón de veces más, sin siquiera hacerlo de la misma manera dos veces” (62).

Un patrón es una solución a un problema en un contexto, codifica conocimiento específico acumulado por la experiencia en un dominio, son soluciones de sentido común que deberían formar parte del conocimiento de un diseñador experto. Además, facilitan la comunicación entre diseñadores, pues establecen un marco de referencia.



Capítulo 3 Implementación y prueba

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de *software* y otros ámbitos referentes al diseño de interacción o interfaces. A continuación se realiza una breve descripción de los patrones utilizados durante la implementación y dónde se encuentran evidenciados en el sistema:

Singleton: Este patrón garantiza que se cree una sola instancia de una clase específica logrando de esta forma que en todos los lugares en las que se requiera se utilice el mismo objeto eliminando de esta forma altos consumos de memoria. En el desarrollo de la aplicación fue usado para proveer una instancia única de la clase conexión usada por Doctrine para acceder a la base de datos.

Creador: Este patrón se manifiesta en gran parte de la solución, al ser creadas instancias de las clases necesarias para ejecutar las funcionalidades.

Experto: Este patrón se manifiesta al crear clases nuevas que heredan atributos y métodos de las ya implementadas y son dotadas de comportamiento afín al propósito para el que fueron diseñadas. Un ejemplo concreto queda reflejado al ver que las clases *DatInspeccion* constituye una especialización de la clase *BaseDatInspeccion*.

Controlador: Este patrón se evidencia cuando las clases controladoras convierten un evento generado por el usuario en una llamada de un método del modelo específico, convirtiéndose así en controladoras del flujo de eventos generados en el sistema.

Bajo acoplamiento: Este patrón se pone de manifiesto al no existir una conexión directa entre las clases controladoras y las clases modelos de dominio creadas por Doctrine y que representa a las tablas de la base de datos.

Builder: Este patrón fue utilizado en la implementación de las interfaces con ExtJS, permitiendo separar la creación de componentes visuales de su utilización, para el logro de este objetivo se desarrolló la clase *Ext.LOG.Util*, encargada de la construcción dinámicamente los elementos visuales más utilizados en la solución. Esta clase brinda funcionalidades que permiten la creación de botones, campos de fecha, ventanas, entre otros.



3.7. Estándares de codificación

Los estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código.

Los cuatro módulos implementados se rigen por los estándares de codificación definidos en el marco de trabajo.

El nombre de las clases se escribe con la primera letra en mayúsculas y las demás en minúsculas. En caso de ser un nombre compuesto es seguido del primer nombre y comienza con minúscula. Ejemplo: Configuracionmodel.

3.8. Nomenclatura

En el desarrollo de la propuesta de solución se hizo necesaria la utilización de las nomenclaturas las cuales permitieran a los programadores la utilización de una misma simbología a la hora de nombrar los códigos a implementar, a continuación se evidencian las nomenclaturas utilizada en la aplicación.

3.8.1. Nomenclatura de las clases según el tipo

Las clases controladoras se encuentran dentro de la carpeta *controller* y a estas clases se le escribe el nombre según su nomenclatura. Ejemplo: Configuracion

Las clases modelos se encuentran dentro de la carpeta *bussines*. A estas clases se le añade la palabra *model*, después de escribir el nombre según su nomenclatura. Ejemplo: Configuracionmodel.

Las clases que representan entidades del dominio se encuentran dentro de la carpeta *domain*. Las clases dominios son generadas por Doctrine, heredan de la clase abstracta *Doctrine_Record* y su nombre se corresponde con el de la tabla de la base de datos a la que representan. Ejemplo: DatEmpresa.

Las clases bases que se encuentran dentro de la carpeta *generated* son generadas por el *framework* Doctrine y delante del nombre de la clase se le añade *Base*. Ejemplo: BaseDatEmpresa.

3.8.2. Nomenclatura de los métodos o funciones



Capítulo 3 Implementación y prueba

El nombre de los métodos de una clase comienzan con minúsculas, en caso de que sea compuesto seguido del primer nombre comienza el segundo con la primera letra en mayúscula. Deben describir el propósito del mismo. Ejemplo: cargarEmpresas(\$params).

3.8.3. Nomenclatura de las variables

El nombre que se le pone a los atributos se escribe con la primera palabra en minúscula, en caso de que sea un nombre compuesto seguido de la primera palabra se escribe la segunda con la inicial en mayúscula. Además en caso de ser un objeto se comienza con: obj y después se escribe el nombre con la primera letra en mayúscula, en caso de ser un nombre compuesto es seguido del primer nombre y comienza con mayúscula. Ejemplo: objFiltro.

3.9. Creación de las tablas en la base de datos

Al ser escogido con anterioridad el *framework* Doctrine ORM se definió un modelo con tablas y sus relaciones, fácil de acceder y moverse por entidades relacionadas entre sí, ya que este Mapeador Relacional de Objetos (ORM) permite acceder a las tablas de una base de datos con la abstracción de la orientación a objetos posibilitando trabajar con objetos en lugar de una tabla directamente con sql.

En la propuesta de solución el modelo de datos hereda de dos clases Doctrine_Record la cual representa una entidad con sus propiedades (columnas) y facilita métodos para insertar, actualizar o eliminar registros, entre otros y Doctrine_Table la cual representa el esquema de una tabla y a través de esta clase se puede obtener información sobre las columnas o buscar registros específicos, entre otros aspectos.

Las estructura de los datos en la base de datos es modelada utilizando la herramienta case Visual Paradigm 8.0, reflejándose así los conceptos, relaciones y reglas del dominio estudiado, con cierto grado de abstracción inherente, con el objetivo de darle respuesta al problema afrontado por el equipo. Es, obviamente, un artefacto indispensable para lograr el éxito definitivo pues se encarga de manejar la persistencia de la información relevante del sistema.



Capítulo 3 Implementación y prueba

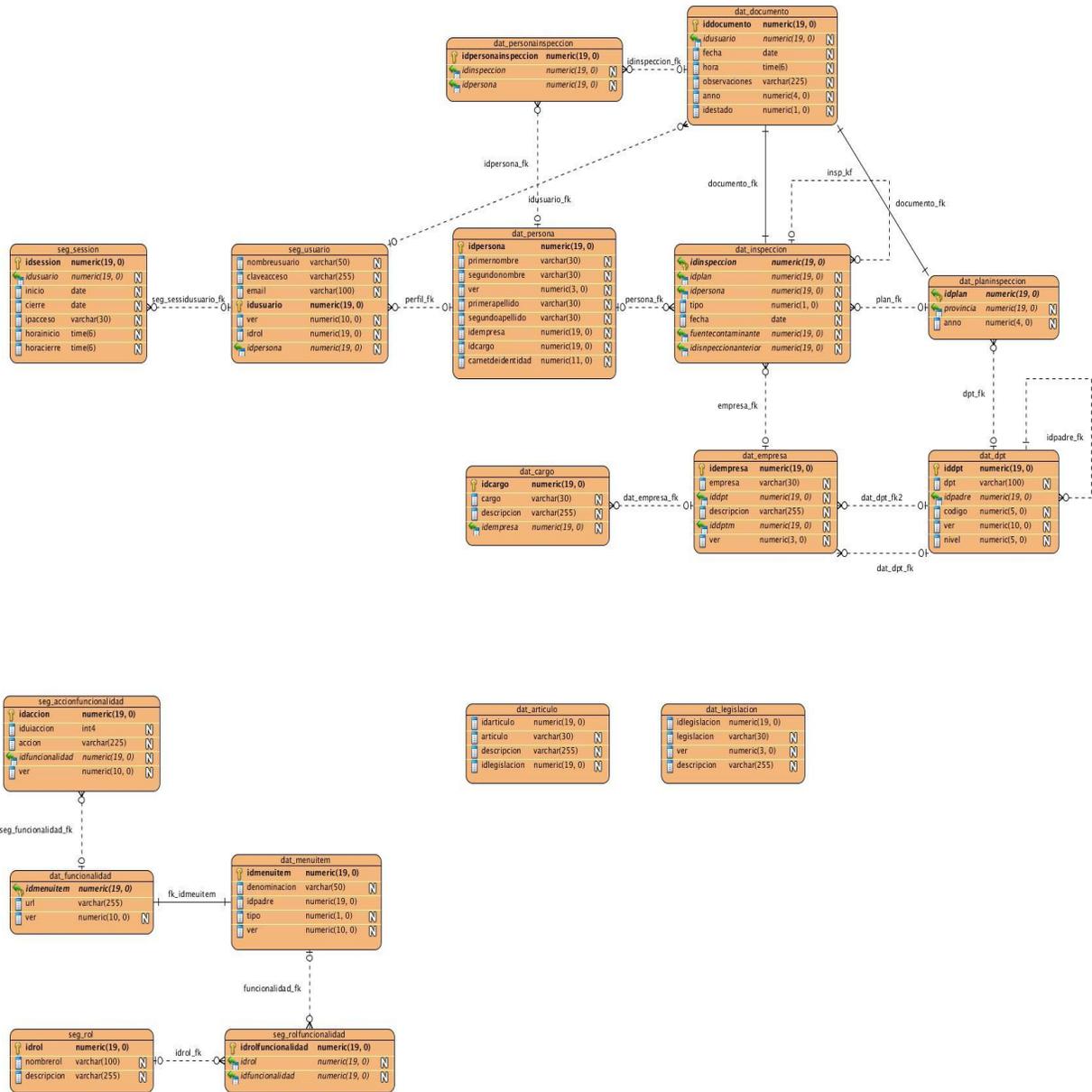


Figura 13: Estructura de las tablas en la base de datos.



Capítulo 3 Implementación y prueba

3.10. Pantallas de la aplicación

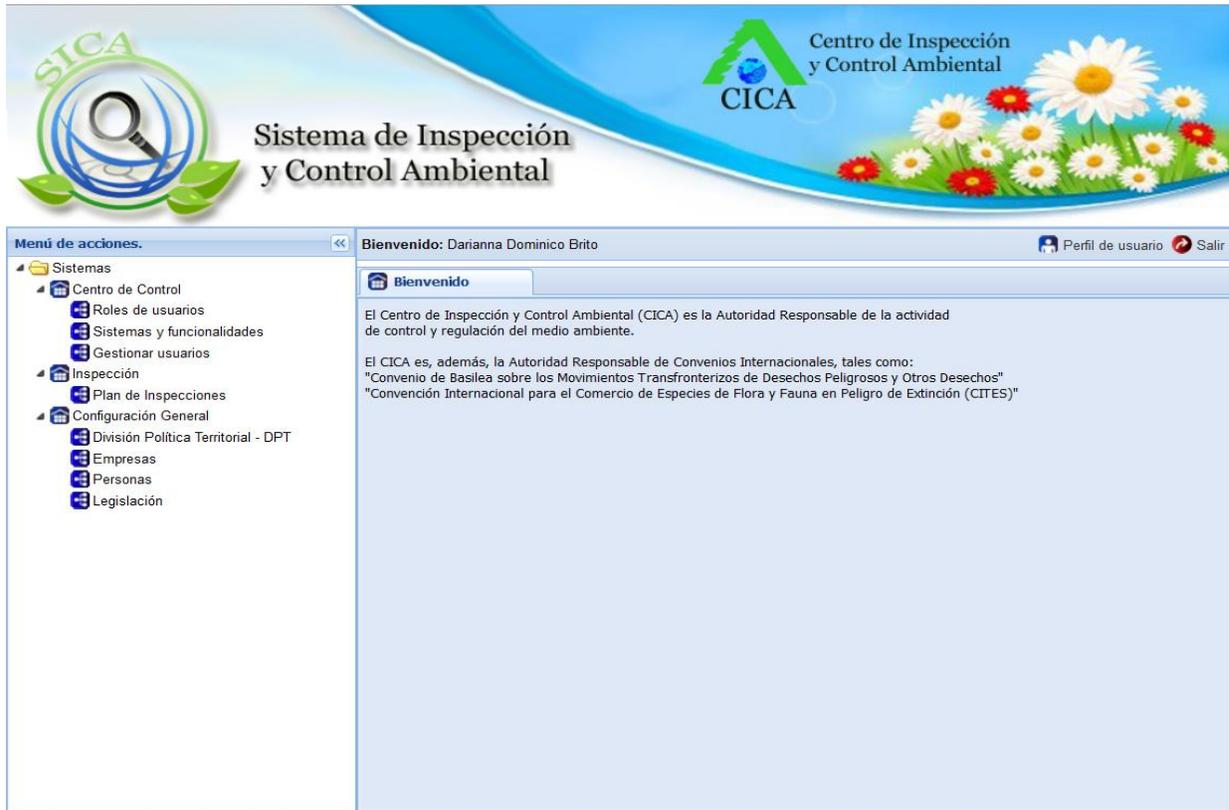


Figura 14: Interfaz principal.



Capítulo 3 Implementación y prueba

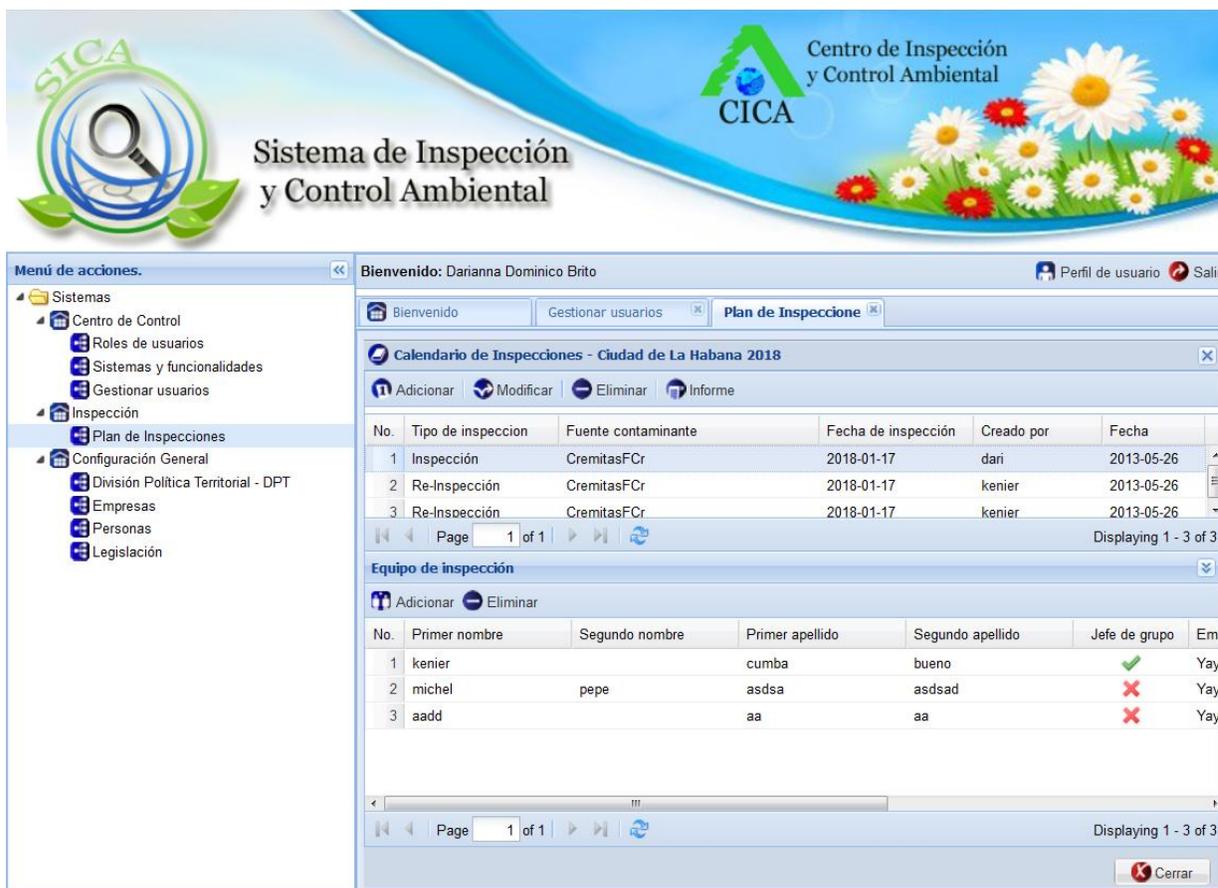


Figura 15: Interfaz gestionar inspección.

Para ver más [Ver Anexo V.](#)

3.11. Pruebas

La metodología XP divide las pruebas del sistema en dos grupos: las 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 diseñadas por el cliente final. (63)

Las pruebas unitarias aseguran que un único componente de la aplicación produce una salida correcta para una determinada entrada. Este tipo de pruebas validan la forma en la que las funciones y métodos trabajan en cada caso particular. Las pruebas unitarias se encargan de un único caso cada vez, lo que significa que un único método puede necesitar varias pruebas unitarias si su funcionamiento varía en función del contexto.



Capítulo 3 Implementación y prueba

Las pruebas funcionales no solo validan la transformación de una entrada en una salida, sino que validan una característica completa. De modo que las pruebas funcionales verifican procesos y requieren de un escenario. A continuación se muestran las pruebas de aceptación de las funcionalidades críticas que fueron implementadas para la propuesta de solución. En los anexos (Ver Anexo IV) se encuentran las restantes pruebas de aceptación:

3.11.1. Pruebas de aceptación de la iteración 1

Tabla 14: Caso de prueba de aceptación HU2_P1

Caso de prueba de aceptación	
Código: HU2_P1	Historia de usuario: 2
Nombre: Adicionar rol.	
Descripción: Prueba para la funcionalidad que permite adicionar los roles.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador.	
Pasos de ejecución: En el módulo Centro de Control del menú principal, el usuario selecciona la opción “Adicionar Rol”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 15: Caso de prueba de aceptación HU2_P2

Caso de prueba de aceptación



Capítulo 3 Implementación y prueba

Código: HU2_P2	Historia de usuario: 2
Nombre: Modificar rol.	
Descripción: Prueba para la funcionalidad que permite modificar un rol existente en el sistema.	
Condiciones de ejecución: ✓ El usuario debe estar autenticado con el rol de administrador.	
Pasos de ejecución: En el módulo Centro de Control del menú principal, el sistema muestra en un campo de selección con los roles existentes. Luego de seleccionar el rol, el usuario selecciona la opción “Modificar Rol” y al accionar el botón modificar se mostrarán los campos con los datos del rol a modificar. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 16: Caso de prueba de aceptación HU2_P3

Caso de prueba de aceptación	
Código: HU2_P3	Historia de usuario: 2
Nombre: Eliminar rol.	
Descripción: Prueba para la funcionalidad que permite eliminar un rol existente en el sistema.	
Condiciones de ejecución:	



Capítulo 3 Implementación y prueba

- ✓ El usuario debe estar autenticado con el rol de administrador.

Pasos de ejecución: Al accionar el módulo Centro de Control del menú principal el sistema muestra en un campo de selección con los roles existentes. Luego de seleccionar el rol, el usuario selecciona la opción “Eliminar rol” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar el rol, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 17: Caso de prueba de aceptación HU4_P2

Caso de prueba de aceptación	
Código: HU4_P2	Historia de usuario: 4
Nombre: Modificar funcionalidades.	
Descripción: Prueba para la funcionalidad que permite modificar una funcionalidad existente en el sistema.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador del sistema.	
Pasos de ejecución: En el módulo Centro de Control del menú principal, el sistema muestra en un campo de selección con las funcionalidades existentes. Luego de seleccionar la funcionalidad, el usuario selecciona la opción “Modificar Funcionalidad” y al accionar el botón modificar se mostrarán los campos con los datos de la funcionalidad a modificar. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada	



satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 18: Caso de prueba de aceptación HU4_P3

Caso de prueba de aceptación	
Código: HU4_P3	Historia de usuario: 4
Nombre: Eliminar funcionalidades.	
Descripción: Prueba para la funcionalidad que permite eliminar una funcionalidad existente en el sistema.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador del sistema.	
Pasos de ejecución: Al accionar el módulo Centro de Control del menú principal al accionar Sistemas y funcionalidades el sistema muestra en un campo de selección con las funcionalidades existentes. Luego de seleccionar la funcionalidad, el usuario selecciona la opción “Eliminar Funcionalidad” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar el rol, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

3.11.2. Pruebas de aceptación de la iteración 2



Tabla 19: Caso de prueba de aceptación HU5_P1

Caso de prueba de aceptación	
Código: HU5_P1	Historia de usuario: 5
Nombre: Adicionar empresa	
Descripción: Prueba para la funcionalidad que permite adicionar una empresa.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de director.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar Empresa”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 20: Caso de prueba de aceptación HU5_P2

Caso de prueba de aceptación	
Código: HU5_P2	Historia de usuario: 5
Nombre: Modificar empresa	
Descripción: Prueba para la funcionalidad que permite modificar una empresa.	
Condiciones de ejecución:	



Capítulo 3 Implementación y prueba

- ✓ El usuario debe estar autenticado con el rol de director.
- ✓ Debe existir al menos una empresa.

Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra en un campo de selección las empresas existentes. Luego de seleccionar la empresa el usuario selecciona la opción “Modificar Empresa” y al accionar el botón modificar se mostrarán los campos con los datos de la empresa. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 21: Caso de prueba de aceptación HU5_P3

Caso de prueba de aceptación	
Código: HU5_P3	Historia de usuario: 5
Nombre: Eliminar empresa	
Descripción: Prueba para la funcionalidad que permite eliminar una empresa.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de director.✓ Debe existir al menos una empresa.	
Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra en un campo de selección las empresas existentes. Luego de seleccionar la empresa el usuario selecciona la opción “Eliminar Empresa” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la empresa de ser afirmativo se le mostrará un mensaje confirmando que la acción fue	



Capítulo 3 Implementación y prueba

ejecutada satisfactoriamente.

Resultado esperado: Los datos son eliminados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 22: Caso de prueba de aceptación HU10_P1

Caso de prueba de aceptación	
Código: HU10_P1	Historia de usuario: 10
Nombre: Adicionar cronograma	
Descripción: Prueba para la funcionalidad que permite adicionar un cronograma.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de planificador.	
Pasos de ejecución: En el módulo Inspección del menú principal, el usuario selecciona la opción “Adicionar Plan”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 23: Caso de prueba de aceptación HU10_P2

Caso de prueba de aceptación



Capítulo 3 Implementación y prueba

Código: HU10_P2	Historia de usuario: 10
Nombre: Modificar cronograma	
Descripción: Prueba para la funcionalidad que permite modificar un cronograma.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de planificador.✓ Debe existir al menos una planificación en el cronograma.	
Pasos de ejecución: Al accionar el módulo Inspección del menú principal el sistema muestra en un campo de selección con las actividades planificadas en el cronograma. Luego de seleccionar la actividad el usuario selecciona la opción “Modificar Cronograma” y al accionar el botón modificar se mostrarán los campos con los datos del cronograma. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 24: Caso de prueba de aceptación HU10_P3

Caso de prueba de aceptación	
Código: HU10_P3	Historia de usuario: 10
Nombre: Eliminar cronograma	
Descripción: Prueba para la funcionalidad que permite modificar un cronograma en el sistema.	



Capítulo 3 Implementación y prueba

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de planificador.
- ✓ Debe existir al menos una planificación en el cronograma.

Pasos de ejecución: Al accionar el módulo Inspección del menú principal el sistema muestra en un campo de selección con las actividades planificadas en el cronograma. Luego de seleccionar la actividad el usuario selecciona la opción “Eliminar Plan” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la inspección del cronograma, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son eliminados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 25: Caso de prueba de aceptación HU11_P1

Caso de prueba de aceptación	
Código: HU11_P1	Historia de usuario: 11
Nombre: Adicionar inspección	
Descripción: Prueba para la funcionalidad que permite adicionar una inspección al cronograma.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de experto.✓ Debe existir planificaciones de inspecciones.	
Pasos de ejecución: En el módulo Inspección del menú principal, el usuario selecciona la opción “Adicionar Cronograma”, seguidamente el sistema muestra los	



Capítulo 3 Implementación y prueba

campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 26: Caso de prueba de aceptación HU11_P2

Caso de prueba de aceptación	
Código: HU11_P2	Historia de usuario: 11
Nombre: Modificar inspección	
Descripción: Prueba para la funcionalidad que permite modificar una inspección en el cronograma.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de experto.✓ Debe existir al menos una planificación de inspección.	
Pasos de ejecución: Al accionar el módulo Inspección del menú principal el sistema muestra en un campo de selección las inspecciones existentes. Luego de seleccionar la inspección el usuario selecciona la opción “Modificar Inspección” y al accionar el botón enviar se mostrarán los campos con los datos de la inspección. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	



Tabla 27: Caso de prueba de aceptación HU11_P3

Caso de prueba de aceptación	
Código: HU11_P3	Historia de Usuario: 11
Nombre: Eliminar inspección	
Descripción: Prueba para la funcionalidad que permite eliminar una inspección del cronograma.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de experto.✓ Debe existir al menos una planificación de inspección.	
Pasos de ejecución: Al accionar el módulo Inspección del menú principal el sistema muestra en un campo de selección las inspecciones existentes. Luego de seleccionar la inspección el usuario selecciona la opción “Eliminar Inspección” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la inspección del cronograma, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son eliminados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Las pruebas a las funcionalidades se realizaron en tres iteraciones obteniendo para la primera 6 no conformidades significativas, 3 no significativas y 10 recomendaciones.

Para la segunda iteración, 3 no conformidades significativas, 5 no significativas y 7 recomendaciones y para la última iteración 3 recomendaciones las cuales fueron solucionadas.

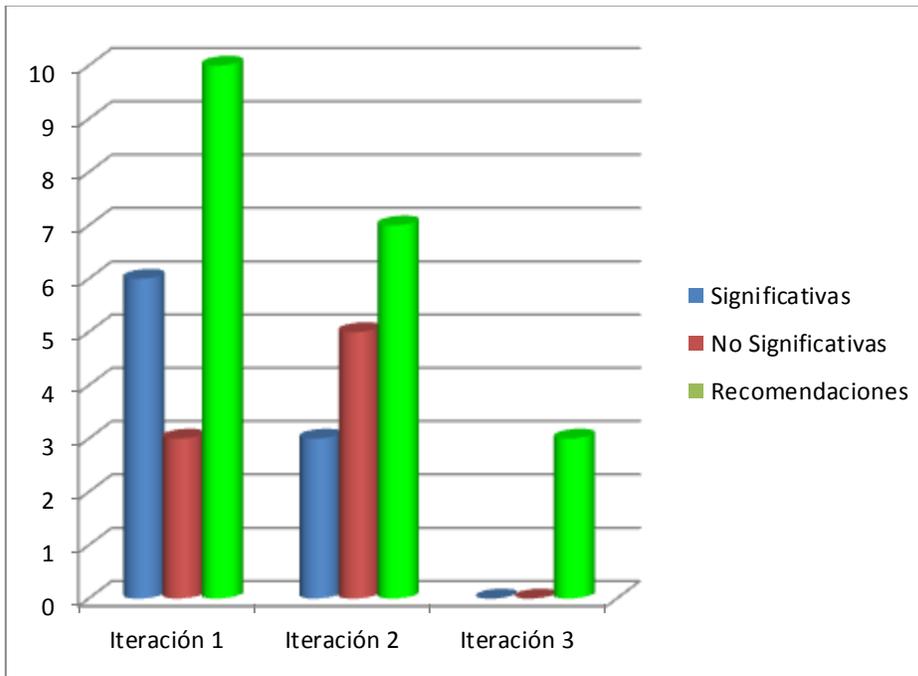


Figura 16: No conformidades significativas, no significativas y las recomendaciones.

Las no conformidades, no significativas, se centraron en errores ortográficos como: omisiones de tildes, paréntesis y cambio de mayúscula por minúscula y las significativas, en errores de validación y cambios en el diseño.

Conclusiones

El diseño propuesto en el capítulo anterior permitió desarrollar las funcionalidades descritas a partir de las herramientas, lenguajes, metodologías y tecnologías seleccionadas en el capítulo 1, solucionando de esta manera las insuficiencias del CICA descritas en la introducción de esta investigación.

En el transcurso de este capítulo se detallaron los patrones de diseño, las tareas de ingeniería, las pruebas realizadas al sistema a lo largo de todo el ciclo de vida, como dicta la metodología y se logró una cobertura de pruebas elevada y que garantiza la calidad del sistema construido. La propuesta de solución de la aplicación a implementar se da por concluida simultáneamente con el fin del capítulo actual.



Conclusiones generales

Después de desarrollar el presente trabajo y analizar los resultados obtenidos, las conclusiones esenciales a las que se arriban son las siguientes:

- ❖ Los métodos científicos y las técnicas de recopilación de datos empleados confirmaron su validez, al permitir el desarrollo de la teoría que sustenta la investigación y la elaboración de las definiciones que posibilitan comprender el problema de la investigación.
- ❖ El estudio realizado permitió definir las herramientas que posibilitaron una base sólida para el desarrollo del sistema, seleccionar como metodología para la guía del proceso a XP, la que posibilitó generar los artefactos fundamentales para el desarrollo de la aplicación.
- ❖ El diseño desarrollado para la aplicación, permitió la implementación de funcionalidades que dieron solución al objetivo general de la investigación.
- ❖ Las pruebas realizadas al sistema de gestión de información demostraron efectividad del empleo de las herramientas y metodologías seleccionadas, lo que favorece además, el trabajo de los expertos del CICA en el manejo y control de la información.



Recomendaciones

Como parte del proceso investigativo llevado a cabo se recomienda:

- ❖ Al cliente, que para nuevas versiones se confeccione el Manual de Usuario de la solución, que guíe a los usuarios que interactúan con la aplicación.
- ❖ Al Centro de Inspección y Control Ambiental generalizar el sistema de gestión de información que se desarrolló, al resto de sus organizaciones en aras de lograr la automatización de sus procesos.
- ❖ A los administradores de la aplicación la actualización sistemática del sistema desarrollado a partir de las nuevas tecnologías y herramientas informáticas.



Trabajos citados

1. **Engels, Federico.** "Carta a Jules Guedes a Paris.". s.l. : Moscú.
2. **Gibert Lamadrid, María del Pilar.** *El profesor y la educación ambiental en la época actual.*
3. **Justicia, Ministerio de.** bvsde.paho. *Gaceta Oficial de Cuba.* [En línea] 22 de 3 de 2005. [Citado el: 11 de 3 de 2013.] <http://www.bvsde.paho.org/bvsacd/cd38/Cuba/R4-05.pdf>. ISSN 1682-7511.
4. **Quiroga, Lic. Lourdes Aja.** bvs.sld. *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones .* [En línea] [Citado el: 11 de 3 de 2013.] http://bvs.sld.cu/revistas/aci/vol10_5_02/aci04502.htm .
5. **Sonora, Instituto Técnico de.** biblioteca itson. *introduccion sistemas.* [En línea] [Citado el: 12 de 3 de 2013.] http://biblioteca.itson.mx/oa/dip_ago/introduccion_sistemas/p3.htm.
6. *GACETA OFICIAL DE LA REPUBLICA DE CUBA. Popular, Asamblea Nacional del Poder.* EDICION EXTRAORDINARIA : LA HABANA, 11 DE JULIO DE 1997, Vol. 7.
7. Medio Ambiente. *CICA.* [En línea] [Citado el: 24 de 3 de 2013.] <http://www.medioambiente.cu/oregulatoria/cica/index.htm>.
8. definicion abc. *contravención.* [En línea] [Citado el: 24 de 3 de 2013.] <http://www.definicionabc.com/derecho/contravencion.php#ixzz2OVLxPs8V>.
9. **Ruz, Fidel Castro.** *Decreto Ley 200.* s.l. : La Habana, 1999.
10. unido.org. *7-Questions.* [En línea] [Citado el: 12 de 4 de 2013.] http://www.unido.org/fileadmin/user_media/Services/Environmental_Management/CP_ToolKit_sp_anish/PR-Volume_07/7-Questions.pdf.
11. **Restrepo, Néstor Julio Jaime.** Control Ambiental. [En línea] 2007. [Citado el: 20 de 3 de 2013.] <http://tcontrolambientals.blogspot.com/> .
12. kioskea. [En línea] [Citado el: 1 de 12 de 2012.] <http://es.kioskea.net/contents/langages/langages.php3>.
13. **Saavedra, jorge.** jorgesaavedra.wordpress. *lenguajes-de-programacion.* [En línea] 5 de 5 de 2007. [Citado el: 4 de 12 de 2012.] <http://jorgesaavedra.wordpress.com/2007/05/05/lenguajes-de-programacion/>.
14. adelat. *lenguajes del lado servidor o cliente.* [En línea] [Citado el: 1 de 12 de 2012.] http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html.



15. maestrosdelweb. *los diferentes lenguajes de programación para la web*. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
16. w3c. *HojasEstilo*. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>.
17. pergaminovirtual. *XHTML*. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.pergaminovirtual.com.ar/definicion/XHTML.html>.
18. hipertexto. *xhtml*. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.hipertexto.info/documentos/xhtmll.htm>.
19. **Centro de Apoyo Tecnológico a Emprendedores, Fundación Parque Científico y Tecnológico de Albacete**. Bilib. [En línea] 13 de 7 de 2012. [Citado el: 18 de 11 de 2012.] http://www.bilib.es/fileadmin/estudio_sistemas_gestion_contenidos_web cms.pdf.
20. **Álvarez, Oswaldo**. ticbeat. *desarrollar portal medida usar cms prefabricado*. [En línea] [Citado el: 17 de 3 de 2013.] <http://www.ticbeat.com/tecnologias/desarrollar-portal-medida-usar-cms-prefabricado/>.
21. **Foote, Ralph Johnson y Brian**. [En línea] 1988. [Citado el: 15 de noviembre de 20012.] <http://encyclopedia.thefreedictionary.com/Software+framework>.
22. **Gutiérrez, Javier J**. lsi. *Framework*. [En línea] [Citado el: 16 de 11 de 2012.] http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
23. **Degiovannini, Marcio**. JavaHispano. *frameworks web*. [En línea] [Citado el: 16 de 11 de 2012.] http://www.javahispano.org/storage/contenidos/frameworks_web.pdf.
24. kabytes. *usar-o-no-un-framework*. [En línea] [Citado el: 16 de 11 de 2012.] www.kabytes.com/programacion/usar-o-no-un-framework/.
25. Consultores Innoves. *struts-2-vs-symfony*. [En línea] [Citado el: 16 de 11 de 2012.] <http://consultores.innoves.es/articulos/struts-2-vs-symfony>.
26. **Newton, Aaron**. jquery vs mootools. [En línea] 5 de 2009. [Citado el: 25 de 2 de 2013.] http://jqueryvsmootools.com/index_es-ar.html.
27. **Rosas, Juan Eladio Sánchez**. antartec. *ExtJS lo bueno lo malo y lo feo*. [En línea] 10 de 2008. [Citado el: 25 de 2 de 2013.] <http://blogs.antartec.com/desarrolloweb/2008/10/extjs-lo-bueno-lo-malo-y-lo-feo/>.



28. **Héctor A. Bareiro, Ricardo G. López.** *Frameworks, conceptos. Comparativa entre Ruby on Rails, Kumbia y Symfony.* [En línea] [Citado el: 29 de 5 de 2013.] <http://www.sicuma.uma.es/sicuma/independientes/argentina08/Bareiro-Lopez/index.htm>.
29. **Álvarez, Miguel Ángel.** desarrolloweb. *manual codeignite.* [En línea] [Citado el: 26 de 2 de 2013.] www.desarrolloweb.com/manuales/manual-codeigniter.html.
30. **Guardado, Iván.** web.ontuts. *utilizando doctrine como orm en php.* [En línea] 6 de 7 de 2010. [Citado el: 26 de 2 de 2013.] <http://web.ontuts.com/tutoriales/utilizando-doctrine-como-orm-en-php>.
31. um. [En línea] [Citado el: 5 de 12 de 2012.] <http://www.um.es/docencia/barzana/IAGP/lagp2.html>.
32. **INTECO, Laboratorio Nacional de Calidad del Software de.** inteco. *guia de ingenieria del software.* [En línea] Marzo de 2009. [Citado el: 21 de 11 de 2012.] <http://www.inteco.es>.
33. Eumed.net. *RUP Etapas Diseño.* [En línea] [Citado el: 25 de 11 de 2012.] <http://www.eumed.net/libros/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.
34. optimus-software. *desventajas-de- scrum.* [En línea] [Citado el: 25 de 11 de 2012.] <http://www.optimus-software.com/noticias/2011/11/28/desventajas-de- scrum>.
35. scribd. *METODOLOGIA-XP.* [En línea] [Citado el: 25 de 11 de 2012.] <http://es.scribd.com/doc/72420606/METODOLOGIA-XP>.
36. asier ares. *leccion 9 xp.* [En línea] 23 de 04 de 2012. <http://asier-ares.blogspot.com/2012/04/leccion-9-xp-33.html>.
37. *Métodos Ágiles de desarrollo de Software.* **Reinoso, Billy.**
38. **Marquez, Pablo.** recursostic educacion. [En línea] [Citado el: 26 de 11 de 2012.] <http://recursostic.educacion.es/observatorio/web/ca/software/software-general/547-pablo-marquez>.
39. guia ubuntu. *PostgreSQL.* [En línea] [Citado el: 26 de 11 de 2012.] <http://www.guia-ubuntu.org/index.php?title=PostgreSQL>.
40. database-bd. *Base De Datos.* [En línea] [Citado el: 27 de 11 de 2012.] <http://database-bd.wikispaces.com/Base+De+Datos>.
41. postgresql. *sobre postgresql.* [En línea] [Citado el: 27 de 11 de 2012.] http://www.postgresql.org.es/sobre_postgresql.



42. **Blanco, Carlos.** carlosblanco.pro. *entornos desarrollo integrado-introduccion*. [En línea] 4 de 2012. [Citado el: 04 de 12 de 2012.] <http://carlosblanco.pro/2012/04/entornos-desarrollo-integrado-introduccion/>.
43. foro.ignetwork. *IDE Entorno integrado de desarrollo*. [En línea] [Citado el: 8 de 12 de 2012.] <http://foro.ignetwork.net/showthread.php?15188-IDE-Entorno-integrado-de-desarrollo-%28Concepto-importante%29>.
44. zend. [En línea] [Citado el: 8 de 12 de 2012.] <http://www.zend.com/en/products/studio/>.
45. **Cerda, Felipe.** api.ning. *netbeans65es_c*. [En línea] [Citado el: 8 de 12 de 2012.] http://api.ning.com/files/PcRFEwSyAz6w2k4r...W/netbeans65es_cl.pdf.
46. netbeans. [En línea] [Citado el: 8 de 12 de 2012.] <http://netbeans.org/>.
47. misrespuestas. *que es un servidor web*. [En línea] [Citado el: 27 de 11 de 2012.] <http://www.misrespuestas.com/que-es-un-servidor-web.html>.
48. sauce.pntic.mec. [En línea] [Citado el: 29 de 11 de 2012.] <http://sauce.pntic.mec.es/crer0052/iis/definici.htm>.
49. scribd. *Servidores Web*. [En línea] [Citado el: 28 de 11 de 2012.] <http://es.scribd.com/doc/27519905/Servidores-Web>.
50. httpd.apache. [En línea] [Citado el: 30 de 11 de 2012.] <http://httpd.apache.org/>.
51. ciberaula. *linux apache intro*. [En línea] [Citado el: 30 de 11 de 2012.] http://linux.ciberaula.com/articulo/linux_apache_intro.
52. **Félix, Alvaro del Castillo San.** redes-linux. *El servidor de web Apache: Introducción práctica: Apache 1.x y 2.0 alpha*. [En línea] Copyright © 2000 by Alvaro del Castillo San Félix bajo FDL. http://www.redes-linux.com/manuales/Servidor_web/apache.pdf.
53. *metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. **Patricio Letelier, María Carmen Penadés**. Valencia : Universidad Politécnica, 2006.
54. **Fowler, M.** designDead. *Is Design Dead?* [En línea] 2001. [Citado el: 17 de 5 de 2013.] <http://www.martinfowler.com/articles/designDead.html>.
55. **Naturales, Departamento de Computación de la Facultad de Ciencias Exactas y.** www-2.dc.uba.ar/materias/seginf/material/Clase02-Unidad2_vf.pdf. [En línea] [Citado el: 24 de 3 de 2013.] http://www-2.dc.uba.ar/materias/seginf/material/Clase02-Unidad2_vf.pdf.



56. **R. Jeffries, A. Anderson y C. Hendrickson.** *Extreme Programming Installed*. Addison-Wesley : s.n., 2001.
57. *Mejorando la gestión de historias de usuario en eXtreme Programming.* **Emilio A. Sánchez, Patricio Letelier, José H. Canós.** s.l. : España.
58. **Leydis Reyes, Maria del Carmen Ruíz, Mónica Vivanco.** *prototipos informáticos*. [En línea] 2009. [Citado el: 5 de 2 de 2013.] <http://sistemas2009unl.wordpress.com/prototipos-informaticos/>.
59. sites.google. *Metodología XP*. [En línea] [Citado el: 8 de 2 de 2013.] <https://sites.google.com/site/xpmetodologia/marco-teorico/funcionamiento>.
60. tecnobits.net. *estructura-de-codeigniter*. [En línea] [Citado el: 24 de 4 de 2013.] <http://www.tecnobits.net/estructura-de-codeigniter/>.
61. **Ariel Erijman, Alejandr Goyén.** *Problemas y soluciones en la implementación de extreme programming*. s.l. : Uruguay, 2001.
62. **Mestras, Juan Pavón.** *Patrones de diseño orientado a objetos*. Facultad de Informática UCM : Madrid, 2004.
63. **J. J. Gutiérrez, M. J. Escalona, M. Mejías y J. Torres.** lsi.us. *Pruebas del sistema en programación extrema*. [En línea] [Citado el: 24 de 5 de 2013.] http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.
64. **Carranza, M. en E.A MA del Consuelo.** *razon y palabra*. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.razonypalabra.org.mx/anteriores/n58/mcarranza.pdf>.
65. **Cuerda, Xavier García.** *Introduccion-a-los-sistemas-de-gestion-de-contenidos-cms-de-codigo-abierto*. [En línea] 29 de 11 de 2004. [Citado el: 18 de 11 de 2012.] <http://mosaic.uoc.edu..>
66. agileopenlima. [En línea] 2011. [Citado el: 25 de 11 de 2012.] <http://agileopenlima2011-ssq.blogspot.com/>.
67. scribd. *Características y Funcionalidad de PostgreSQL*. [En línea] [Citado el: 27 de 11 de 2012.] [2http://es.scribd.com/doc/63764883/52/Caracteristicas-y-Funcionalidad-de-PostgreSQL](http://es.scribd.com/doc/63764883/52/Caracteristicas-y-Funcionalidad-de-PostgreSQL).
68. slideshare. *ventajas y desventajas de los servidores apache y iis*. [En línea] [Citado el: 29 de 11 de 2012.] http://www.slideshare.net/Anthony_mejias/ventajas-y-desventajas-de-los-servidores-apache-y-iis#btnNext.
69. programaciondesarrollo. *ventajas del servidor apache*. [En línea] [Citado el: 30 de 11 de 2012.] <http://programaciondesarrollo.es/ventajas-del-servidor-apache/>.



70. thaira.net. *symfony beneficios e informacion*. [En línea] [Citado el: 4 de 12 de 2012.] <http://www.thaira.net/blog/desarrolloweb/symfony-beneficios-e-informacion/>.
71. **Lopez, Bareiro**. sicuma. [En línea] [Citado el: 4 de 12 de 2012.] <http://www.sicuma.uma.es/sicuma/independientes/argentina08/Bareiro-Lopez/index.htm>.
72. **METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES. Roberth G Figueroa, Camilo J. Solís, Armando A. Cabrera**. Loja : s.n.
73. **Ruiz, Juan Sotomayor**. upcommons.upc. *Web del Colegio de Arquitectos de Andorra*. [En línea] 29 de 6 de 2007. [Citado el: 4 de 12 de 2012.] <http://upcommons.upc.edu/pfc/handle/2099.1/4605>.
74. **Kew, Nick**. *Desarrollo de módulos y aplicaciones con Apache*. España : ANAYA MULTIMEDIA, 2008.
75. **Jiménez, K Camacho**. um . *Retos para evaluar el impacto de Internet:el caso de las organizaciones de la sociedad civil centroamericana*. [En línea] 2000. [Citado el: 6 de 12 de 2012.] <http://www.um.es/fccd/anales/ad07/ad0703.pdf>.
76. **AMBIENTAL, OFICINA DE SUPERVISIÓN Y CONTROL**. uma.pinar. *licencia*. [En línea] [Citado el: 6 de 12 de 2012.] <http://www.uma.pinar.cu/licencia.htm>.
77. koala-soft. *bases de datos difusas*. [En línea] [Citado el: 7 de 12 de 2012.] <http://www.koala-soft.com/bases-de-datos-difusas>.
78. documentoweb. *Notepad procesador de textos*. [En línea] [Citado el: 8 de 12 de 2012.] <http://www.documentoweb.com/Software/Notepad-procesador-de-textos>.
79. secyt.frba.utn.edu. *inteligencia artificial*. [En línea] [Citado el: 11 de 12 de 2012.] http://www.secyt.frba.utn.edu.ar/gia/inteligencia_artificial.htm.
80. E-com DevBlog. *notepad the new ide*. [En línea] enero de 2011. [Citado el: 7 de enero de 2013.] <http://dev.tonic1394.com/2011/01/notepad-the-new-ide/>.
81. **Martínez, Eduardo**. revista futuros. *gestión ambiental*. [En línea] [Citado el: 12 de 3 de 2013.] http://www.revistafuturos.info/futuros_3/gestion_amb.htm . ISSN 1913-6196.



Anexo I

Tabla 28: Historia de Usuario de Gestionar Usuario

Historia de Usuario		
No. 1	Nombre: Gestionar Usuario	
Usuario: Administrador		
Prioridad de Negocio: Baja	Puntos Estimados: 0.5	
Nivel de Complejidad: Baja	Iteración Asignada: 4	
Descripción: Permite crear, modificar y eliminar usuarios en la Base de Datos. En caso de adicionar los permisos administrativos solo los asigna el rol Administrador.		

Tabla 29: Historia de Usuario de Gestionar Rol

Historia de Usuario		
No. 2	Nombre: Gestionar Rol	
Usuario: Administrador		
Prioridad de Negocio: Baja	Puntos Estimados: 1	
Nivel de Complejidad: Baja	Iteración Asignada: 4	
Descripción: Permite crear, modificar y eliminar los roles en la base de datos. En caso de adicionar los permisos administrativos solo los asigna el rol Administrador.		



Observaciones: Para que un usuario posea un determinado rol debe primero registrarse en el sistema.

Tabla 30: Historia de Usuario de Autenticar Usuario

Historia de Usuario	
No. 3	Nombre: Autenticar Usuario
Usuario: Administrador	
Prioridad de Negocio: Baja	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 4
Descripción: El usuario podrá autenticarse para acceder a algunas opciones del sistema según los permisos otorgados. Datos para autenticarse: <ul style="list-style-type: none">✓ Nombre del Usuario (Obligatorio).✓ Contraseña (Obligatorio).	
Observaciones: Para que un usuario pueda autenticarse en el sistema debe primero registrarse.	

Tabla 31: Historia de Usuario de Gestionar Funcionalidades

Historia de Usuario	
No. 4	Nombre: Gestionar Funcionalidades



Usuario: Administrador del sistema

Prioridad de Negocio: Baja

Puntos Estimados: 1

Nivel de Complejidad: Alta

Iteración Asignada: 4

Descripción: Permite crear, modificar y eliminar la estructura del menú y cada funcionalidad (contienen un conjunto de acciones) de la Base de Datos. En caso de adicionar los permisos administrativos solo los asigna el rol Administrador del sistema.

Tabla 32: Historia de Usuario de Gestionar Empresa

Historia de Usuario	
No. 5	Nombre: Gestionar Empresa
Usuario: Director	
Prioridad de Negocio: Alta	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 1
Descripción: Permite crear, modificar y eliminar las empresas en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Director.	

Tabla 33: Historia de Usuario de Gestionar Persona

Historia de Usuario



No. 6	Nombre: Gestionar Persona
Usuario: Administrador	
Prioridad de Negocio: Alta	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 1
Descripción: Permite crear, modificar y eliminar las personas en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de administrador.	

Tabla 34: Historia de Usuario de Gestionar DPT

Historia de Usuario	
No. 7	Nombre: Gestionar DPT
Usuario: Gestor de configuración	
Prioridad de Negocio: Alta	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 2
Descripción: Permite crear, modificar y eliminar una provincia y/o municipio en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Gestor de configuración.	



Tabla 35: Historia de Usuario de Gestionar Cargo

Historia de Usuario	
No. 8	Nombre: Gestionar Cargo
Usuario: Gestor de configuración	
Prioridad de Negocio: Alta	Puntos Estimados: 0.3
Nivel de Complejidad: Baja	Iteración Asignada: 2
Descripción: Permite crear, modificar y eliminar los cargos en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Gestor de configuración.	

Tabla 36: Historia de Usuario de Gestionar Legislación

Historia de Usuario	
No. 9	Nombre: Gestionar Legislación
Usuario: Gestor de configuración	
Prioridad de Negocio: Alta	Puntos Estimados: 1
Nivel de Complejidad: Baja	Iteración Asignada: 2
Descripción: Permite crear, modificar y eliminar legislaciones de acuerdo a las políticas del gobierno en cuanto a la protección ambiental en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Gestor	



de configuración.

Tabla 37: Historia de Usuario de Gestionar Cronograma

Historia de Usuario	
No. 10	Nombre: Gestionar Cronograma
Usuario: Planificador	
Prioridad de Negocio: Alta	Puntos Estimados: 1
Nivel de Complejidad: Alta	Iteración Asignada: 1
Descripción: Permite crear, modificar y eliminar la planificación de las inspecciones en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de planificador.	
Observaciones:	

Tabla 38: Historia de Usuario de Generar Reporte

Historia de Usuario	
No. 12	Nombre: Generar Reporte
Usuario: Gestor de configuración, Director, Experto	
Prioridad de Negocio: Alta	Puntos Estimados: 2
Nivel de Complejidad: Alta	Iteración Asignada: 3



Descripción: Permite generar las planillas de las IAE y de las EIA y exportar estos datos a formato pdf.

Observaciones: Para generar y exportar las plantillas se deben haber realizado con anterioridad las historias de usuarios enumeradas a continuación: 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.

Tabla 39: Historia de Usuario de Gestionar Contravenciones

Historia de Usuario	
No. 13	Nombre: Gestionar Contravenciones
Usuario: Experto	
Prioridad de Negocio: Alta	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 3
Descripción: Permite crear, modificar y eliminar las contravenciones en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Experto.	
Observaciones: Debe haberse realizado las historias de usuarios 11 y/o 14 con anterioridad.	

Tabla 40: Historia de Usuario de Gestionar Dictámenes

Historia de Usuario	
No. 15	Nombre: Gestionar Dictámenes
Usuario: Experto	



Prioridad de Negocio: Alta	Puntos Estimados: 0.5
Nivel de Complejidad: Baja	Iteración Asignada: 3
Descripción: Permite crear, modificar y eliminar los dictámenes en la Base de Datos. En caso de agregar los permisos solo lo asigna el rol que posea permisos de Experto.	

Tabla 41: Historia de Usuario Apariencia

Historia de Usuario	
No. 16	Nombre: Apariencia
Prioridad del negocio: Alta	
Descripción: El sistema contará con una interfaz sencilla e intuitiva, manteniendo el formato en los navegadores.	
Observaciones: El sistema contará con enlaces bien identificados, permitiendo acceder a cualquiera de las secciones con un número mínimo de clics y estará optimizado para una resolución de 1024x768.	

Tabla 42: Historia de Usuario Condiciones Tecnológicas

Historia de Usuario	
No. 17	Nombre: Condiciones tecnológicas
Prioridad del negocio: Media	
Descripción: Garantizar que el acceso a la aplicación se realice desde máquinas	



con pocas características.

Observaciones: El sistema debe visualizarse en estaciones de trabajo para los clientes con sistemas operativos Linux y Windows con al menos 100 MB de capacidad de disco duro. Las estación de trabajo para el servidor debe ser Pentium IV a 3.0 GHz, 1 GB de RAM, 10 GB de disco duro o superior y tener instalado el servidor de base de datos PostgreSQL 8.3.5 o superior y el servidor web Apache.

Tabla 43: Historia de Usuario de Seguridad

Historia de Usuario	
No. 18	Nombre: Seguridad
Prioridad del negocio: Alta	
Descripción: La autenticación debe ser	
Observaciones: Los usuarios deberán estar autenticados para interactuar con la aplicación, contando con diferentes niveles de permisos para el acceso a la información, los administradores para poder realizar algún cambio en el sistema, tendrán obligatoriamente que estar autenticados. Cada usuario autenticado podrá realizar las operaciones correspondientes con su rol definido en el sistema.	

Anexo II

Prototipos de Interfaz de Usuario

BANNER

Menú de Acciones

🔒 Cambiar Contraseña 🚪 Salir

- ▼ Sistema
- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- ▷ Configuración General
- ▷ Inspección
- ▷ Licencia

🏠 Bienvenidos
Sistemas y Funcionalidades
Gestionar Usuario

✓ Adicionar
✎ Modificar
✕ Eliminar
↔ Asignar Perfil
✕ Desconectar

	Usuario	Email	Activo	Nombre Completo
1	True	...

Adicionar Usuario: ✕

Usuaric:

Email:

Rol:

Clave de acceso:

Repetir clave de acceso:

✕ Cancelar ✓ Aceptar



BANNER

Menú de Acciones   Cambiar Contraseña  Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - [Gestionar Usuario](#)
- ▷ Configuración General
- ▷ Inspección
- ▷ Licencia

 Bienvenidos |  Sistemas y Funcionalidades |  [Gestionar Usuario](#)

✓ Adicionar  Modificar  Eliminar  Asignar Perfil  Desconectar

...	Usuario	Email	Activo	Nombre Completo
1	dari	dari@prueb.com	True	Amanda Clark
2	beyonce	beyonc@prueb.com	False	Elena Guibert

Modificar Usuario: dari

Email	Rol:
<input type="text" value="dari@prueb.com"/>	<input type="text" value="Administrador"/>

 Cancelar  Aceptar

BANNER

Menú de Acciones   Cambiar Contraseña  Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - [Gestionar Usuario](#)
- ▷ Configuración General
- ▷ Inspección
- ▷ Licencia

 Bienvenidos |  Sistemas y Funcionalidades |  [Gestionar Usuario](#)

✓ Adicionar  Modificar  Eliminar  Asignar Perfil  Desconectar

...	Usuario	Email	Activo	Nombre Completo
1	dari	dari@prueb.com	True	Amanda Clark
2	beyonce	beyonc@prueb.com	False	Elena Guibert

Confirmación:

 ¿Desea eliminar el usuario seleccionado?



BANNER

Menú de Acciones ◀

🏠 Bienvenidos ⊞ Sistemas y Funcionalidades ⊞ **Gestionar Rol** ⊞

Adicionar Modificar Eliminar Asignar Perfil Desconectar

Adicionar Rol: ⊞

Rol:	<input type="checkbox"/> Sistema
<input type="text"/>	<input type="checkbox"/> Centro de Control
Descripción:	<input type="checkbox"/> Folea de Usuario
<input type="text"/>	<input type="checkbox"/> Sistemas y Funcionalidades
	<input type="checkbox"/> Gestionar Usuario
	<input type="checkbox"/> Configuración General
	<input type="checkbox"/> División Política Territorial- DPT
	<input type="checkbox"/> Empresa
	<input type="checkbox"/> Legislación
	<input checked="" type="checkbox"/> Inspección



BANNER

Menú de Acciones ◀ 🔒 Cambiar Contraseña ✕ Salir

🏠 Bienvenidos 📄 Sistemas y Funcionalidades 📄 Gestionar Rol 📄

Adicionar ✎ Modificar ✕ Eliminar ↔ Asignar Perfil ✕ Desconectar

...	Rol	Descripción
1	Administrador	Administrador del sistema
2	Inspector	Encargado de Inspeccionar

Modificar Rol: Administrador 📄

Rol:

Descripción:

- Sistema
- Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- Configuración General
 - División Política Territorial- DPT
 - Empresa
 - Legislación
- Inspección

✕ Cerrar ✓ Aceptar

BANNER

Menú de Acciones
 Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - [Roles de Usuario](#)
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▶ Configuración General
 - ▶ Inspección
 - ▶ Licencia Ambiental

Bienvenidos
Sistemas y Funcionalidades
[Gestionar Rol](#)

✓ Adicionar
Modificar
Eliminar
Asignar Perfil
Desconectar

	Rol	Descripción
1	Administrador	Administrador del sistema
2	Inspector	Encargado de Inspeccionar

Confirmación:

¿Desea eliminar el rol seleccionado?

Ok
Cancelar

BANNER

Menú de Acciones

Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - Configuración General
 - Inspección

Bienvenidos
Sistemas y Funcionalidades

Sistemas registrados

Adicionar
 Modificar
 Eliminar

Acciones registradas

Registrar acciones
 Eliminar

- Sistema
 - Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - Configuración General
 - Inspección
 - Cronograma de Ins
 - Licencia Ambiental

Adicionar Funcionalidad

Nombre de la funcionalidad:

Referencia:

Cancelar
 Aceptar

BANNER

Menú de Acciones

Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - Configuración General
 - Inspección

Bienvenidos
Sistemas y Funcionalidades

Sistemas registrados

Adicionar
 Modificar
 Eliminar

Acciones registradas

Registrar acciones
 Eliminar

- Sistema
 - Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - Configuración General
 - Inspección
 - Cronograma de Ins
 - Licencia Ambiental

Adicionar Funcionalidad

Nombre de la funcionalidad:

Referencia:

Cancelar
 Aceptar

BANNER

Menú de Acciones Cambiar Contraseña Salir

<ul style="list-style-type: none"> ▼ Sistema <ul style="list-style-type: none"> ▼ Centro de Control <ul style="list-style-type: none"> Roles de Usuario Sistemas y Funcionalidades Gestionar Usuario ► Configuración General ► Inspección 	Bienvenidos Sistemas y Funcionalidades				
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Sistemas registrados</td> <td style="width: 50%;">Acciones registradas</td> </tr> </table>	Sistemas registrados	Acciones registradas		
	Sistemas registrados	Acciones registradas			
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"> <input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar </td> <td style="width: 50%;"> <input type="checkbox"/> Registrar acciones <input checked="" type="checkbox"/> Eliminar </td> </tr> </table>	<input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar	<input type="checkbox"/> Registrar acciones <input checked="" type="checkbox"/> Eliminar		
	<input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar	<input type="checkbox"/> Registrar acciones <input checked="" type="checkbox"/> Eliminar			
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; text-align: center;"> <input type="checkbox"/> Sistema <ul style="list-style-type: none"> <input type="checkbox"/> Centro de Control <ul style="list-style-type: none"> Roles de Usuario Sistemas y Funcionalidades Gestionar Usuario <input type="checkbox"/> Configuración General <input type="checkbox"/> Inspección <input type="checkbox"/> Cronograma <input type="checkbox"/> Licencia Ambiental </td> <td style="width: 50%; text-align: center;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">...</td> <td style="width: 30%;">ID Acción</td> <td style="width: 50%;">Acción</td> </tr> </table> </td> </tr> </table>	<input type="checkbox"/> Sistema <ul style="list-style-type: none"> <input type="checkbox"/> Centro de Control <ul style="list-style-type: none"> Roles de Usuario Sistemas y Funcionalidades Gestionar Usuario <input type="checkbox"/> Configuración General <input type="checkbox"/> Inspección <input type="checkbox"/> Cronograma <input type="checkbox"/> Licencia Ambiental 	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">...</td> <td style="width: 30%;">ID Acción</td> <td style="width: 50%;">Acción</td> </tr> </table>	...	ID Acción	Acción
<input type="checkbox"/> Sistema <ul style="list-style-type: none"> <input type="checkbox"/> Centro de Control <ul style="list-style-type: none"> Roles de Usuario Sistemas y Funcionalidades Gestionar Usuario <input type="checkbox"/> Configuración General <input type="checkbox"/> Inspección <input type="checkbox"/> Cronograma <input type="checkbox"/> Licencia Ambiental 	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">...</td> <td style="width: 30%;">ID Acción</td> <td style="width: 50%;">Acción</td> </tr> </table>	...	ID Acción	Acción	
...	ID Acción	Acción			

Confirmación:

¿Desea eliminar la funcionalidad seleccionada?

Ok Cancelar

BANNER

Menú de Acciones Cambiar Contraseña Salir

<ul style="list-style-type: none"> ▼ Sistema <ul style="list-style-type: none"> ▼ Centro de Control <ul style="list-style-type: none"> Roles de Usuario Sistemas y Funcionalidades Gestionar Usuario ▼ Configuración General <ul style="list-style-type: none"> División Política Territorial Empresa Legislación Persona ► Inspección 	Bienvenidos División Política Territorial				
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"> <input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar </td> <td style="width: 50%;"></td> </tr> </table>	<input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar			
	<input checked="" type="checkbox"/> Adicionar <input type="checkbox"/> Modificar <input checked="" type="checkbox"/> Eliminar				
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Código</td> <td style="width: 50%;">Denominación</td> </tr> <tr> <td> <ul style="list-style-type: none"> ▼ DPT <ul style="list-style-type: none"> ► 22 ► 43 52 </td> <td style="vertical-align: top;"> División Política Territorial Santiago de Cuba </td> </tr> </table>	Código	Denominación	<ul style="list-style-type: none"> ▼ DPT <ul style="list-style-type: none"> ► 22 ► 43 52 	División Política Territorial Santiago de Cuba
	Código	Denominación			
<ul style="list-style-type: none"> ▼ DPT <ul style="list-style-type: none"> ► 22 ► 43 52 	División Política Territorial Santiago de Cuba				

Adicionar DPT

Código:	Denominación:
<input type="text"/>	<input type="text"/>

Cancelar Aceptar

BANNER

Menú de Acciones

 Bienvenidos División Política Territorial

 Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▼ Configuración General
 - División Política Territorial
 - Empresa
 - Legislación
 - Persona
 - ▷ Inspección

Bienvenidos División Política Territorial

 Adicionar Modificar Eliminar

Código	Denominación
▼ DPT	División Política Territorial
▷ 22	Santiago de Cuba
▷ 43	Cienfuegos
52	

Modificar DPT

Código:	Denominación:
43	Cienfuegos

Cancelar
 Aceptar

BANNER

Menú de Acciones

 Bienvenidos División Política Territorial

 Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▼ Configuración General
 - División Política Territorial
 - Empresa
 - Legislación
 - Persona
 - ▷ Inspección

Bienvenidos División Política Territorial

 Adicionar Modificar Eliminar

Código	Denominación
▼ DPT	División Política Territorial
▷ 22	Santiago de Cuba
▷ 43	Cienfuegos
52	

Confirmación:

¿Desea eliminar el dpt seleccionado?

Ok
Cancelar



BANNER

Menú de Acciones   Cambiar Contraseña  Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- ▼ Configuración General
 - División Política Territorial
 - Empresa**
 - Legislación
 - Persona
- ▷ Inspección
- ▷ Licencia Ambiental

 Bienvenidos **Empresa** 

Adicionar  Modificar  Eliminar

	No.	Empresa	Provincia	Municipio
<input type="checkbox"/>	1	Aguas Claras	Santa Clara	Caibarien
<input type="checkbox"/>	2	Ville	Santiago de Cuba	Palma Soriano
<input type="checkbox"/>	3	Guabairo	Cienfuegos	Cruces

Adicionar Empresa 

Empresa: Provincia: Municipio:

Descripción:



BANNER

Menú de Acciones   Cambiar Contraseña  Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▼ Configuración General
 - División Política Territorial
 - Empresa**
 - Legislación
 - Persona
 - Inspección
 - Licencia Ambiental

 Bienvenidos **Empresa** 

Adicionar  Modificar  Eliminar

No.	Empresa	Municipio	Provincia
 1	Aguas Claras	Caibarien	Santa Clara
 2	Ville	Palma Soriano	Santiago de Cuba
 3	Guabairo	Cruces	Cienfuegos

Modificar Empresa: Aguas Claras 

Empresa:	Provincia:	Municipio:
<input type="text" value="Aguas Claras"/>	<input type="text" value="Santa Clara"/> ▼	<input type="text" value="Caibarien"/> ▼
Descripción:		
<input type="text" value="Empresa de agua mineral"/>		

 Cancelar  Aceptar

2	Secretaria	Encargado de hacer papeles



BANNER

Menú de Acciones Cambiar Contraseña Salir

Sistema

- Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- Configuración General
 - División Política Territorial
 - Empresa**
 - Legislación
 - Persona
- Inspección
- Licencia Ambiental

Bienvenidos **Empresa**

Adicionar Modificar Eliminar

	No.	Empresa	Municipio	Provincia
	1	Aguas Claras	Caibarien	Santa Clara
	2	Ville	Palma Soriano	Santiago de Cuba
	3	Guabairo	Cruces	Cienfuegos

Confirmación:

¿Desea eliminar la empresa seleccionada?

Cargos

Adicionar Modificar Eliminar

	No.	Cargo	Descripción
	1	Director	Encargado de dirigir la empresa
	2	Secretaria	Encargado de hacer papeles



BANNER

Menú de Acciones Cambiar Contraseña Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- ▼ Configuración General
 - División Política Territorial
 - Empresa**
 - Legislación
 - Persona
- ▷ Inspección
- ▷ Licencia Ambiental

Bienvenidos **Empresa**

Adicionar Modificar Eliminar

	No.	Empresa	Municipio	Provincia
<input type="checkbox"/>	1	Aguas Claras	Caibarien	Santa Clara
<input type="checkbox"/>	2	Ville	Palma Soriano	Santiago de Cuba
<input type="checkbox"/>	3	Guabairo	Cruces	Cienfuegos

Cargos

Adicionar Modificar Eliminar

No.	Cargo
1	Director
2	Secretaria

Adicionar cargo

Código:

Denominación:

Cancelar Aceptar

			Encargado de hacer papeles
--	--	--	----------------------------



BANNER

Menú de Acciones

Cambiar Contraseña Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- ▼ Configuración General
 - División Política Territorial
 - Empresa**
 - Legislación
 - Persona
- ▷ Inspección
- ▷ Licencia Ambiental

Bienvenidos **Empresa**

✓ Adicionar Modificar Eliminar

No.	Empresa	Municipio	Provincia
1	Aguas Claras	Caibarien	Santa Clara
2	Ville	Palm	
3	Guabairo	Cru	

Modificar Cargo

Cargo:

Descripción:

Cancelar Aceptar

Cargos

✓ Adicionar Modificar Eliminar

No.	Cargo	Descripción
1	Director	Encargado de dirigir la empresa
2	Secretaria	Encargado de hacer papeles



BANNER

Menú de Acciones

Cambiar Contraseña Salir

▼ Sistema

- ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
- ▼ Configuración General
 - División Política Territorial
 - [Empresa](#)
 - Legislación
 - Persona
- ▷ Inspección
- ▷ Licencia Ambiental

Bienvenidos [Empresa](#)

Adicionar Modificar Eliminar

	No.	Empresa	Municipio	Provincia
	1	Aguas Claras	Caibarien	Santa Clara
	2	Ville	Palma Soriano	Santiago de Cuba
	3	Guabairo	Cruces	Cienfuegos

Confirmación:

¿Desea eliminar el cargo seleccionado?

Cargos

Adicionar Modificar

No.	Cargo	Descripción
1	Director	Encargado de dirigir la empresa
2	Secretaria	Encargado de hacer papeles



BANNER

Menú de Acciones ⏪ 🔒 Cambiar Contraseña 🚪 Salir

🏠 Bienvenidos Empresa Legislación

[✓ Adicionar](#) [✎ Modificar](#) [✕ Eliminar](#)

No.	Legislación
1	Ley 80
Descripción: Ley del M	
<input type="checkbox"/>	2
<input type="checkbox"/>	3

Artículo

[✓ Adicionar](#) [✎](#)

Adicionar Legislación

Legislación:

Descripción:

No.	Artículo	Descripción



BANNER

Menú de Acciones Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▼ Configuración General
 - División Política Territorial
 - Empresa
 - Legislación
 - Persona
 - ▷ Inspección
 - ▷ Licencia Ambiental

Bienvenidos Empresa [Legislación](#)

✓ Adicionar [Modificar](#) Eliminar

	No.	Legislación
	1	Ley 80
Descripción: Ley del Medio Ambiente		
	2	Decreto Ley 200
	3	Ley 99

Modificar Legislación: Ley 99

Legislación:

Descripción:

Cancelar Aceptar



BANNER

Menú de Acciones

Cambiar Contraseña Salir

- ▼ Sistema
 - ▼ Centro de Control
 - Roles de Usuario
 - Sistemas y Funcionalidades
 - Gestionar Usuario
 - ▼ Configuración General
 - División Política Territorial
 - Empresa
 - Legislación
 - Persona
 - ▷ Inspección
 - ▷ Licencia Ambiental

Bienvenidos | Empresa | Legislación

Adicionar Modificar Eliminar

	No.	Legislación
<input type="checkbox"/>	1	Ley 80
Descripción: Ley del Medio Ambiente		
<input type="checkbox"/>	2	Decreto Ley 200
<input type="checkbox"/>	3	Ley 99

Confirmación:

¿Desea eliminar la legislación seleccionada?

Artículo	No.	Descripción	Detalle
	1	Artículo 1	Se enuncia....
	2	Artículo 2	Según



Anexo III

Tabla 44: Tarjeta CRC Portal.acceso.Login.Vista

Portal.acceso.Login.Vista	
Responsabilidades	Colaboradores
✓ init	✓ Portal.acceso.LoginWin ✓ Ext.Button

Tabla 45: Tarjeta CRC Portal.acceso.Login.Controlador

Portal.acceso.Login.Controlador	
Responsabilidades	Colaboradores
✓ init	✓ Xtiaris.Lib.Base64 ✓ Xtiaris.Lib.Util

Tabla 46: Tarjeta CRC Portal.portal.Escritorio.Vista

Portal.portal.Escritorio.Vista	
Responsabilidades	Colaboradores
✓ Init	✓ Ext.tree.TreePanel ✓ Ext.Panel ✓ Ext.Button ✓ Ext.TabPanel

Tabla 47: Tarjeta CRC Portal.portal.Escritorio.Modelo

Portal.portal.Escritorio.Modelo	
---------------------------------	--



Responsabilidades	Colaboradores
✓ Init	✓ Ext.tree.TreeLoader

Tabla 48: Tarjeta CRC Portal.portal.Escritorio.Modelo

Portal.portal.Escritorio.Controlador	
Responsabilidades	Colaboradores
✓ Init	✓ Xtiaris.Lib.Util ✓ Ext.Button ✓ Ext.Ajax.request ✓ Ext.decode

Tabla 49: Tarjeta CRC Seguridad.sistemas.Sistemas.Modelo

Seguridad.sistemas.Sistemas.Modelo	
Responsabilidades	Colaboradores
✓ Init	✓ Ext.data.Store ✓ Ext.data.JsonReader ✓ Ext.tree.TreeLoader

Tabla 50: Tarjeta CRC Seguridad.sistemas.Sistemas.Vista

Seguridad.sistemas.Sistemas.Vista	
Responsabilidades	Colaboradores
✓ Init	✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel



	<ul style="list-style-type: none"> ✓ Ext.grid.ColumnModel ✓ Ext.grid.RowNumberer ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Ext.tree.DefaultSelectionModel ✓ Ext.tree.TreePanel ✓ Ext.Viewport
✓ mostrarVentana	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Window

Tabla 51: Tarjeta CRC Seguridad.sistemas.Sistemas.Controlador

Seguridad.sistemas.Sistemas.Controlador	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.getCmp
<ul style="list-style-type: none"> ✓ cargarSistemas ✓ habilitarBotonesUI ✓ eventosGestion 	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.tree.DefaultSelectionModel

Tabla 52: Tarjeta CRC Seguridad.usuario.Usuario.Modelo

Seguridad.usuario.Usuario.Modelo	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Ext.data.Store ✓ Ext.data.JsonReader



Tabla 53: Tarjeta CRC Seguridad.usuario.Usuario.Vista

Seguridad.usuario.Usuario.Vista	
Responsabilidades	Colaboradores
✓ Init	✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel ✓ Ext.grid.ColumnModel ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Ext.Viewport
✓ ventanaUsuario	✓ Ext.Button ✓ Ext.form.FormPanel ✓ Ext.Window
✓ ventanaPersonas	✓ Ext.grid.RowSelectionModel ✓ Ext.grid.ColumnModel ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Ext.Button ✓ Ext.Window

Tabla 54: Tarjeta CRC Seguridad.usuario.Usuario.Controlador

Seguridad.usuario.Usuario.Controlador	
Responsabilidades	Colaboradores
✓ Init	✓ Xtiaris.Lib.Base64 ✓ Xtiaris.Lib.Util



<ul style="list-style-type: none"> ✓ habilitarBotonesUI ✓ eventosGestion 	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.tree.DefaultSelectionModel
--	--

Tabla 55: Tarjeta CRC Seguridad.roles.Rol.Modelo

Seguridad.roles.Rol.Modelo	
Responsabilidades	Responsabilidades
❖ Init	<ul style="list-style-type: none"> ✓ Ext.data.Store ✓ Ext.data.JsonReader ✓ Ext.tree.TreeLoader

Tabla 56: Tarjeta CRC Seguridad.roles.Rol.Vista

Seguridad.roles.Rol.Vista	
Responsabilidades	Responsabilidades
✓ init	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel ✓ Ext.grid.ColumnModel ✓ Ext.grid.RowNumberer ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Ext.Viewport ✓ Xtiaris.Lib.Util
✓ ventanaGestion	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.tree.DefaultSelectionModel ✓ Seguridad.roles.Rol.TreeSistemas



	<ul style="list-style-type: none"> ✓ Ext.form.FormPanel ✓ Ext.Window
--	--

Tabla 57: Tarjeta CRC Seguridad.roles.Rol.Controlador

Seguridad.roles.Rol.Controlador	
Responsabilidades	Responsabilidades
✓ Init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.Button
✓ habilitarBotonesUI	<ul style="list-style-type: none"> ✓ Ext.Button
✓ eventosGestion	<ul style="list-style-type: none"> ✓ Ext.each ✓ Ext.Button ✓ Ext.encode

Tabla 58: Tarjeta CRC Configuracion.nomencladores.Dpt.Modelo

Configuracion.nomencladores.Dpt.Modelo	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Ext.ux.tree.TreeGridLoader

Tabla 59: Tarjeta CRC Configuracion.nomencladores.Dpt.Vista

Configuracion.nomencladores.Dpt.Vista	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Ext.Button



	<ul style="list-style-type: none"> ✓ Ext.Panel ✓ Ext.tree.DefaultSelectionModel ✓ Configuracion.dpt.DptTree ✓ Ext.Viewport ✓ Xtiaris.Lib.Util
✓ ventanaGestion	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Window

Tabla 60: Configuracion.nomencladores.Dpt.Controlador

Configuracion.nomencladores.Dpt.Controlador	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext Button
<ul style="list-style-type: none"> ✓ habilitarBotonesUI ✓ listenersAdicionar 	<ul style="list-style-type: none"> ✓ Ext.Button

Tabla 61: Configuracion.nomencladores.Empresa.Modelo

Configuracion.nomencladores.Empresa.Modelo	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Ext.data.Store ✓ Ext.data.JsonReader

Tabla 62: Configuracion.nomencladores.Empresa.Vista

Configuracion.nomencladores.Empresa.Vista	
---	--



Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel ✓ Configuracion.Empresa.Expander ✓ Ext.Template ✓ Ext.grid.ColumnModel ✓ Ext.grid.RowNumberer ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Ext.grid.ColumnModel ✓ Ext.grid.RowSelectionModel ✓ Ext.Viewport ✓ Xtiaris.Lib.Util
✓ ventanaCargos	<ul style="list-style-type: none"> ✓ Ext.Button
✓ ventanaGestion	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Window ✓ Ext.form.FormPanel

Tabla 63: Configuracion.nomencladores.Empresa.Controlador

Configuracion.nomencladores.Empresa.Controlador	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.Button
✓ eventosAdicionarCargo	<ul style="list-style-type: none"> ✓ Ext.Button
✓ eventosAdicionar	<ul style="list-style-type: none"> ✓ Ext.getCmp



Tabla 64: Tarjeta CRC Seguridadmodel

Seguridadmodel	
Responsabilidades	Colaboradores
✓ __construct	
✓ cargarMenu ✓ cargarArbolSistemas	✓ DatMenuitem
✓ salvarUsuario	✓ Doctrine ✓ SegUsuario
✓ perfilUsuario	✓ Doctrine
✓ devolverUsuarios ✓ cerrarSesionUsuario	✓ SegUsuario
✓ eliminarUsuario ✓ eliminarSistema ✓ eliminarRol	✓ Doctrine_Query
✓ salvarMenuitem ✓ salvaFuncionalidad	✓ Doctrine ✓ DatMenuitem
✓ salvaAccion	✓ Doctrine ✓ SegAccionfuncionalidad
✓ cargarAcciones	✓ DatFuncionalidad
✓ salvarRol ✓ asignarPerfil	✓ Doctrine ✓ Doctrine_Query
✓ devolverRoles	✓ SegRol
✓ validarUsuario	✓



Tabla 65: Tarjeta CRC Configuracionmodel

Configuracionmodel	
Responsabilidades	Responsabilidades
✓ __construct	✓
✓ delvorverDpt ✓ devolverArticulo	✓ DatDpt ✓ DatArticulo
✓ salvarDpt ✓ salvarEmpresa ✓ salvarCargo ✓ salvarLegislacion ✓ salvarArticulo ✓ salvarPersona	✓ Doctrine
✓ eliminarDpt ✓ eliminarEmpresa ✓ eliminarArticulo ✓ eliminarCargo ✓ eliminarLegislacion ✓ eliminarPersona	✓ Doctrine_Query
✓ cargarEmpresas	✓ DatEmpresa
✓ cargarLegislacion	✓ DatLegislacion
✓ cargarPersona	✓ DatPersona
✓ devolverCargos	✓ DatCargo

Tabla 66: Tarjeta CRC DatCargo

DatCargo	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ Doctrine_Query	✓ Doctrine:Query

Tabla 67: Tarjeta CRC DatDocumento

DatDocumento	
Responsabilidades	Colaboradores
✓ setUp	✓

Tabla 68: Tarjeta CRC DatDpt

DatDpt	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ cargarDpt	
✓ devolverArbolDpt	✓ Doctrine_Query

Tabla 69: Tarjeta CRC DatEmpresa

DatEmpresa	
Responsabilidades	Colaboradores



✓ setUp	✓
✓ devolverEmpresas	✓ Doctrine_Query

Tabla 70: Tarjeta CRC SegAccionfuncionalidad

SegAccionfuncionalidad	
Responsabilidades	Colaboradores
✓ setUp	✓

Tabla 71: Tarjeta CRC SegRol

SegRol	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ obtenerRoles	✓ Doctrine_Query

Tabla 72: Tarjeta CRC SegRolfuncionalidad

SegRolfuncionalidad	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ addDqlSegRolFunc	✓ Doctrine_Query



Tabla 73: Tarjeta CRC SegSession

SegSession	
Responsabilidades	Colaboradores
✓ setUp	✓

Tabla 74: Tarjeta CRC SegUsuario

SegUsuario	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ cargarUsuarios ✓ cerrarSession	✓ Doctrine_Query
✓ autenticarUsuario	✓ Doctrine_Query ✓ SegSession

Tabla 75: Tarjeta CRC DatFuncionalidad

DatFuncionalidad	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ cargarAccionesAsociadas	✓ Doctrine_Query

Tarjeta 76: Tarjeta CRC DatMenuitem

DatMenuitem	
-------------	--



Responsabilidades	Colaboradores
✓ setUp	✓
✓ devolverArbolFuncionalidades	✓ Doctrine_Query

Tabla 77: Tarjeta CRC DatArticulo

DatArticulo	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ devolverArticulo	✓ Doctrine_Query

Tarjeta 78: Tarjeta CRC DatLegislacion

DatLegislacion	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ devolverLegislacion	✓ Doctrine_Query

Tarjeta 79: Tarjeta CRC DatPersona

DatPersona	
Responsabilidades	Colaboradores
✓ setUp	✓



Tabla 80: Tarjeta CRC DatPersonainspeccion

DatPersonainspeccion	
Responsabilidades	Colaboradores
✓ setUp	✓

Tabla 81: Tarjeta CRC Welcome

Welcome	
Responsabilidades	Colaboradores
✓ __construct ✓ index ✓ cerrarsesion ✓ devolvermenu ✓ cargarperfilusuario	✓
✓ Autenticarusuario	✓ stdClass

Tabla 82: Tarjeta CRC Seguridad

Seguridad	
Responsabilidades	Colaboradores
✓ __construct ✓ Index ✓ gestusuario ✓ adicionarusuario ✓ modificarusuario ✓ eliminarusuario	✓



<ul style="list-style-type: none"> ✓ desconectarusuario ✓ gestsistemas ✓ devolversistemas ✓ devolversistemaschecked ✓ adicionarsistema ✓ modificarsistema ✓ eliminarsistema ✓ adicionarfuncionalidad ✓ modificarfuncionalidad ✓ adicionaraccion ✓ adicionarusuario ✓ modificarusuario ✓ eliminarusuario ✓ devolveracciones ✓ gestroles ✓ cargarroles ✓ adicionarrol ✓ modificarrol ✓ eliminarrol ✓ asignarpersonas 	
<ul style="list-style-type: none"> ✓ cargarusuarios ✓ devolverroles ✓ devolverroles2 ✓ devolverpersonas 	<ul style="list-style-type: none"> ✓ stdClass

Tabla 83: Tarjeta CRC Configuración

Configuración	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ __construct 	<ul style="list-style-type: none"> ✓



<ul style="list-style-type: none"> ✓ gestionardpt ✓ cargardpt ✓ salvardatosdpt ✓ eliminarddpt ✓ devolverdpt ✓ gestionarempresas ✓ salvarempresa ✓ modificarempresa ✓ cargarempresas ✓ eliminarempresa ✓ devolvercargos ✓ adicionarcargo ✓ modificarecargo ✓ eliminarcargo ✓ gestionarlegislacion ✓ salvarlegislacion ✓ modificarlegislacion ✓ eliminarlegislacion ✓ cargarlegislacion ✓ devolverarticulo ✓ adicionararticulo ✓ modificararticulo ✓ eliminararticulo 	
--	--

Tabla 84: Tarjeta CRC Menu

Menu	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ Index ✓ __construct 	<ul style="list-style-type: none"> ✓



<ul style="list-style-type: none"> ✓ Cerrarsesion ✓ Cargarperfilusuario 	
<ul style="list-style-type: none"> ✓ devolvermenu ✓ autenticarusuario 	<ul style="list-style-type: none"> ✓ stdClass

Tabla 85: Tarjeta CRC Recursos humanos

Recursos humanos	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ __construct ✓ gestpersonas ✓ salvarpersona ✓ modificarpersona ✓ cargarpersona ✓ eliminarpersona ✓ cargarEmpresaPersona 	<ul style="list-style-type: none"> ✓
<ul style="list-style-type: none"> ✓ cargarCargoEmpresa 	<ul style="list-style-type: none"> ✓ stdClass

Tabla 86: Tarjeta CRC Configuracion.nomencladores.Legislacion.Modelo

Configuracion.nomencladores.Legislacion.Modelo	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ init 	<ul style="list-style-type: none"> ✓ Ext.data.Store ✓ Ext.data.JsonReader

Tabla 87: Configuracion.nomencladores.Legislacion.Vista

Configuracion.nomencladores.Legislacion.Vista	
---	--



Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ Init 	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel ✓ Configuracion.Legislacion.Expander ✓ Ext.grid.ColumnModel ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Xtiaris.Lib.Util
<ul style="list-style-type: none"> ✓ ventanaArticulo ✓ ventanaGestion 	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Window ✓ Ext.form.FormPanel

Tabla 88: Tarjeta CRC Configuracion.nomencladores.Legislacion.Controlador

Configuracion.nomencladores.Legislacion.Controlador	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> ✓ init 	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.Ajax.request ✓ Ext.Button
<ul style="list-style-type: none"> ✓ eventosAdicionarArticulo ✓ eventosAdicionar 	<ul style="list-style-type: none"> ✓ Ext.Button

Tabla 89: Tarjeta CRC Configuracion.nomencladores.Persona.Modelo

Configuracion.nomencladores.Persona.Modelo	
Responsabilidades	Colaboradores



✓ init	<ul style="list-style-type: none"> ✓ Ext.data.Store ✓ Ext.data.JsonReader
--------	---

Tabla 90: Tarjeta CRC Configuracion.nomencladores.Persona.Vista

Configuracion.nomencladores.Persona.Vista	
Responsabilidades	Colaboradores
✓ init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.Button ✓ Ext.Panel ✓ Ext.grid.RowSelectionModel ✓ Configuracion.Legislacion.Expander ✓ Ext.grid.ColumnModel ✓ Ext.grid.GridPanel ✓ Ext.PagingToolbar ✓ Xtiaris.Lib.Util
✓ ventanaGestion	<ul style="list-style-type: none"> ✓ Ext.Button ✓ Ext.Window ✓ Ext.form.FormPanel

Tabla 91: Tarjeta CRC Configuracion.nomencladores.Persona.Controlador

Configuracion.nomencladores.Persona.Controlador	
Responsabilidades	Colaboradores
✓ Init	<ul style="list-style-type: none"> ✓ Xtiaris.Lib.Util ✓ Ext.Ajax.request ✓ Ext.Button



✓ eventosAdicionar	✓ Ext.Button
--------------------	--------------

Tabla 92: Tarjeta CRC Inspeccion.plan.Cronograma.Modelo

Inspeccion.plan.Cronograma.Modelo	
Responsabilidades	Colaboradores
✓ Init	✓ Ext.data.GroupingStore ✓ Ext.data.JsonReader ✓ Ext.data.Store

Tabla 93: Tarjeta CRC DatInspeccion

DatInspeccion	
Responsabilidades	Colaboradores
✓ setUp	✓

Tabla 94: Tarjeta CRC Inspeccion.plan.Cronograma.Controlador

Inspeccion.plan.Cronograma.Controlador	
Responsabilidades	Colaboradores
✓ Init	✓ Xtiaris.Lib.Util ✓ Ext.Button ✓ Ext.Ajax.request
✓ listenerGestion	✓ Ext.Button



Tabla 95: Tarjeta CRC DatPlaninspeccion

DatPlaninspeccion	
Responsabilidades	Colaboradores
✓ setUp	✓
✓ devolverPlanes	✓ Doctrine_Query



Anexo IV

Pruebas de aceptación de la Iteración 1

Tabla 96: Caso de prueba de aceptación HU1_P1

Caso de Prueba de Aceptación	
Código: HU1_P1	Historia de Usuario: 1
Nombre: Adicionar Usuario.	
Descripción: Prueba de la funcionalidad adicionar usuario.	
Condiciones de ejecución: ✓ El usuario debe estar autenticado con el rol de Administrador.	
Pasos de ejecución: En el módulo Centro de Control del menú principal, el usuario selecciona la opción “Adicionar Usuario”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 97: Caso de prueba de aceptación HU1P2

Caso de Prueba de Aceptación	
Código: HU1_P2	Historia de Usuario: 1
Nombre: Modificar Usuario.	



Descripción: Prueba de la funcionalidad modificar usuario.

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de Administrador.

Pasos de ejecución: En el módulo Centro de Control del menú principal, el sistema muestra en un campo de selección los usuarios existentes. Luego de seleccionar el usuario, el mismo selecciona la opción “Modificar Usuario” y al accionar el botón modificar se mostrarán los campos con los datos del usuario a modificar. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 98: Caso de prueba de aceptación HU1_P3

Caso de Prueba de Aceptación	
Código: HU1_P3	Historia de Usuario: 1
Nombre: Eliminar usuario.	
Descripción: Prueba de la funcionalidad eliminar usuario.	
Condiciones de ejecución:	
<ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Administrador.	



Pasos de ejecución: Al accionar el módulo Centro de Control del menú principal el sistema muestra en un campo de selección con los usuarios existentes. Luego de seleccionar el usuario, el mismo selecciona la opción “Eliminar Usuario” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar el usuario, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son eliminados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 99: Caso de prueba de aceptación HU3_P1

Caso de Prueba de Aceptación	
Código: HU3_P1	Historia de Usuario: 3
Nombre: Autenticar usuario.	
Descripción: Prueba de la funcionalidad autenticar usuario.	
Condiciones de ejecución: ✓ El usuario debe estar registrado en el sistema.	
Pasos de ejecución: Debe comprobar que sea un usuario válido del sistema.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 100: Caso de prueba de aceptación HU4_P1

Caso de Prueba de Aceptación



Código: HU4_P1	Historia de Usuario: 4
Nombre: Adicionar funcionalidades.	
Descripción: Prueba de la funcionalidad adicionar funcionalidades.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Administrador del Sistema.	
Pasos de ejecución: En el módulo Centro de Control del menú principal, el usuario selecciona la opción “Adicionar Funcionalidades”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Pruebas de aceptación de la Iteración 2

Tabla 101: Caso de prueba de aceptación HU6_P1

Caso de Prueba de Aceptación	
Código: HU6_P1	Historia de Usuario: 6
Nombre: Adicionar persona	
Descripción: Prueba de la funcionalidad adicionar persona.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador.✓ Debe existir una empresa.	



Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar Persona”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 102: Caso de prueba de aceptación HU6_P2

Caso de Prueba de Aceptación	
Código: HU6_P2	Historia de Usuario: 6
Nombre: Modificar persona	
Descripción: Prueba de la funcionalidad modificar persona.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador.✓ Debe existir al menos una empresa.✓ Debe existir al menos una persona.	
Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra en un campo de selección las personas existentes. Luego de seleccionar la persona el usuario selecciona la opción “Modificar Persona” y al accionar el botón modificar se mostrarán los campos con los datos de la persona. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	



Evaluación de la prueba: Prueba satisfactoria.

Tabla 103: Caso de prueba de aceptación HU6_P3

Caso de Prueba de Aceptación	
Código: HU6_P3	Historia de Usuario: 6
Nombre: Eliminar persona	
Descripción: Prueba de la funcionalidad eliminar persona.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de administrador.✓ Debe existir al menos una empresa.✓ Debe existir al menos una persona.	
Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra en un campo de selección las personas existentes. Luego de seleccionar la empresa el usuario selecciona la opción “Eliminar Persona” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la persona, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son eliminados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Pruebas de aceptación de la Iteración 3

Tabla 104: Caso de prueba de aceptación HU7_P1

Caso de Prueba de Aceptación



Código: HU7_P1	Historia de Usuario: 7
Nombre: Adicionar DPT	
Descripción: Prueba de la funcionalidad adicionar DPT.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar DPT”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 105: Caso de prueba de aceptación HU7_P2

Caso de Prueba de Aceptación	
Código: HU7_P2	Historia de Usuario: 7
Nombre: Modificar DPT	
Descripción: Prueba de la funcionalidad modificar DPT.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración.✓ Debe existir al menos una provincia y/o municipio.	



Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra las DPT existentes. Luego el sistema muestra las provincias y/o municipios existentes, al seleccionar la provincia y/o municipio el usuario selecciona la opción “Modificar DPT” y al accionar el botón modificar se mostrarán los campos con los datos de la DPT. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 106: Caso de prueba de aceptación HU7_P3

Caso de Prueba de Aceptación	
Código: HU7_P3	Historia de Usuario: 7
Nombre: Eliminar DPT	
Descripción: Prueba de la funcionalidad eliminar DPT.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración.✓ Debe existir al menos una provincia y/o municipio.	
Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra las DPT existentes. Luego de seleccionar la empresa el usuario selecciona la opción “Eliminar DPT” y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la DPT, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son eliminados satisfactoriamente.	



Evaluación de la prueba: Prueba satisfactoria.

Tabla 107: Caso de prueba de aceptación HU8_P1

Caso de Prueba de Aceptación	
Código: HU8_P1	Historia de Usuario: 8
Nombre: Adicionar cargo	
Descripción: Prueba de la funcionalidad crear cargo.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar Cargo”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 108: Caso de prueba de aceptación HU8_P2

Caso de Prueba de Aceptación	
Código: HU8_P2	Historia de Usuario: 8
Nombre: Modificar cargo	



Descripción: Prueba de la funcionalidad modificar cargo.

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de Gestor de configuración.
- ✓ Debe existir al menos un usuario con cargo.

Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Modificar Cargo”, seguidamente el sistema muestra las empresas con los cargos existentes. Luego de seleccionarlo y accionar el botón modificar se mostrarán los campos con los datos de los mismos. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 109: Caso de prueba de aceptación HU8_P3

Caso de Prueba de Aceptación	
Código: HU8_P3	Historia de Usuario: 8
Nombre: Eliminar cargo	
Descripción: Prueba de la funcionalidad eliminar cargo.	
Condiciones de ejecución:	
<ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración.✓ Debe existir al menos un usuario con cargo.	



Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Eliminar Cargo”, seguidamente el sistema muestra los usuarios con cargos existentes. Luego de seleccionarlo y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar el cargo, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 110: Caso de prueba de aceptación HU9_P1

Caso de prueba de aceptación	
Código: HU9_P1	Historia de usuario: 9
Nombre: Adicionar legislación	
Descripción: Prueba para la funcionalidad que permite adicionar una legislación.	
Condiciones de ejecución: ✓ El usuario debe estar autenticado con el rol de gestor de configuración.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar Legislación”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 111: Caso de prueba de aceptación HU9_P2

Caso de prueba de aceptación	
Código: HU9_P2	Historia de usuario: 9
Nombre: Modificar legislación	
Descripción: Prueba para la funcionalidad que permite modificar una legislación en el sistema.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> ✓ El usuario debe estar autenticado con el rol de gestor de configuración. ✓ Debe existir al menos una legislación de acuerdo a las políticas del gobierno en cuanto a la protección ambiental en la BD. 	
Pasos de ejecución: Al accionar el módulo Configuración General del menú principal el sistema muestra en un campo de selección las legislaciones existentes. Luego de seleccionar la legislación el usuario selecciona la opción “Modificar Legislación” y al accionar el botón modificar se mostrarán los campos con los datos de la legislación. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 112: Caso de prueba de aceptación HU9_P3

Caso de prueba de aceptación	
Código: HU9_P3	Historia de usuario: 9



Nombre: Eliminar legislación

Descripción: Prueba para la funcionalidad que permite eliminar una legislación del sistema.

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de gestor de configuración.
- ✓ Debe existir al menos una legislación de acuerdo a las políticas del gobierno en cuanto a la protección ambiental en la BD.

Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Eliminar Legislación”, seguidamente el sistema muestra las legislaciones existentes. Luego de seleccionarlo y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la legislación, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 113: Caso de prueba de aceptación HU14_P1

Caso de prueba de aceptación	
Código: HU14_P1	Historia de usuario: 14
Nombre: Adicionar LA	
Descripción: Prueba para la funcionalidad que permite adicionar una LA.	
Condiciones de ejecución:	
✓ El usuario debe estar autenticado con el rol de director.	



- ✓ Debe existir al menos un dictamen.

Pasos de ejecución: En el módulo Licencia Ambiental del menú principal, el usuario selecciona la opción “Adicionar LA”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 114: Caso de prueba de aceptación HU14_P2

Caso de prueba de aceptación	
Código: HU14_P2	Historia de usuario: 14
Nombre: Modificar LA	
Descripción: Prueba para la funcionalidad que permite modificar una LA en el sistema.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de director.✓ Debe existir al menos un dictamen.	
Pasos de ejecución: Al accionar el módulo LA del menú principal el sistema muestra en un campo de selección las licencias ambientales existentes. Luego de seleccionar la licencia ambiental el usuario selecciona la opción “Modificar LA” y al accionar el botón modificar se mostrarán los campos con los datos de la legislación. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	



Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 115: Caso de prueba de aceptación HU14_P3

Caso de prueba de aceptación	
Código: HU14_P3	Historia de usuario: 14
Nombre: Eliminar LA	
Descripción: Prueba para la funcionalidad que permite eliminar una LA.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de director.✓ Debe existir al menos un dictamen.	
Pasos de ejecución: En el módulo Licencia Ambiental del menú principal, el usuario selecciona la opción “Eliminar LA”, seguidamente el sistema muestra las LA existentes. Luego de seleccionarlo y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la legislación, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son eliminados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	



Pruebas de aceptación de la Iteración 4

Tabla 116: Caso de prueba de aceptación HU12_P1

Caso de Prueba de Aceptación	
Código: HU12_P1	Historia de Usuario: 12
Nombre: Generar Reporte.	
Descripción: Prueba de la funcionalidad generar reporte.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Gestor de configuración, Director, Experto.✓ Deben de existir al menos una empresa, una persona, un DPT, un cargo, una legislación, un cronograma, una inspección, una contravención y un dictamen.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Generar reporte”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 117: Caso de prueba de aceptación HU13_1

Caso de Prueba de Aceptación	
Código: HU13_P1	Historia de Usuario: 13



Nombre: Adicionar Contravenciones.

Descripción: Prueba de la funcionalidad adicionar contravenciones.

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de Experto.
- ✓ Debe existir al menos una Inspección Ambiental o una Licencia Ambiental.

Pasos de ejecución: En el módulo Inspección del menú principal, el usuario selecciona la opción “Adicionar Contravenciones”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 118: Caso de prueba de aceptación HU13_P2

Caso de Prueba de Aceptación	
Código: HU13_P2	Historia de Usuario: 13
Nombre: Modificar Contravenciones.	
Descripción: Prueba de la funcionalidad modificar contravenciones.	
Condiciones de ejecución:	
<ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Experto.✓ Debe existir al menos una Inspección Ambiental o una Licencia Ambiental.	
Pasos de ejecución: Al accionar el módulo Inspección del menú principal el sistema	



muestra en un campo de selección las contravenciones existentes. Luego de seleccionar la contravención el usuario selecciona la opción “Modificar Contravención” y al accionar el botón modificar se mostrarán los campos con los datos de la contravención. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 119: Caso de prueba de aceptación HU13_P3

Caso de Prueba de Aceptación	
Código: HU13_P3	Historia de Usuario: 13
Nombre: Eliminar Contravenciones.	
Descripción: Prueba de la funcionalidad eliminar contravenciones.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Experto.✓ Debe existir al menos una Inspección Ambiental o una Licencia Ambiental.	
Pasos de ejecución: En el módulo Inspección del menú principal, el usuario selecciona la opción “Eliminar Contravenciones”, seguidamente el sistema muestra las Contravenciones existentes. Luego de seleccionarlo y accionar sobre el botón eliminar, debe mostrar un mensaje para confirmar si desea eliminar la legislación, de ser afirmativo se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	



Evaluación de la prueba: Prueba satisfactoria.

Tabla 120: Caso de prueba de aceptación HU15_P1

Caso de Prueba de Aceptación	
Código: HU15_P1	Historia de Usuario: 15
Nombre: Adicionar Dictámenes.	
Descripción: Prueba de la funcionalidad adicionar dictámenes.	
Condiciones de ejecución: <ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Experto.	
Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Adicionar Dictámenes”, seguidamente el sistema muestra los campos para introducir los datos requeridos. Después de haber introducido los datos, el usuario acciona sobre el botón aceptar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.	
Resultado esperado: Los datos son registrados satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 121: Caso de prueba de aceptación HU15_P2

Caso de Prueba de Aceptación	
Código: HU15_P2	Historia de Usuario: 15
Nombre: Modificar Dictámenes.	



Descripción: Prueba de la funcionalidad modificar dictámenes.

Condiciones de ejecución:

- ✓ El usuario debe estar autenticado con el rol de Experto.

Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Modificar Dictámenes”, seguidamente el sistema muestra en un campo de selección los dictámenes existentes. Luego de seleccionarlo y accionar el botón enviar se mostrarán los campos con los datos de los mismos. Después de haberlos modificado, el usuario acciona sobre el botón enviar y se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 122: Caso de prueba de aceptación HU15_P3

Caso de Prueba de Aceptación	
Código: HU15_P3	Historia de Usuario: 15
Nombre: Eliminar dictamen.	
Descripción: Prueba de la funcionalidad eliminar dictamen.	
Condiciones de ejecución:	
<ul style="list-style-type: none">✓ El usuario debe estar autenticado con el rol de Experto.	



Pasos de ejecución: En el módulo Configuración General del menú principal, el usuario selecciona la opción “Eliminar Dictamen”, seguidamente el sistema muestra en un campo de selección los dictámenes existentes. Luego de seleccionarlo y accionar sobre el botón eliminar, se le mostrará un mensaje confirmando que la acción fue ejecutada satisfactoriamente.

Resultado esperado: Los datos son registrados satisfactoriamente.

Evaluación de la prueba: Prueba satisfactoria.



Glosario de términos

Código abierto: Es una tendencia internacional del desarrollo de software que proclama la distribución del código junto a las aplicaciones, se rige por licencias tales como GNU/GPL.

Scripts: Guión o fichero de texto que contiene una serie de instrucciones que se pueden ejecutar.

Software libre: Software que puede ser distribuido, modificado, redistribuido, copiado y usado libremente. Se basa en cuatro libertades: libertad para usarlo con cualquier propósito, libertad para modificarlo a las necesidades, libertad para distribuir copias y para mejorarlo. Que un software sea libre no quiere decir que sea gratuito, error que viene de la traducción Free Software.

Objetos: En el paradigma de POO, un objeto se define como la unidad que en tiempo de ejecución realiza las tareas de un programa. También a un nivel más básico se define como la instancia de una clase. Estos objetos interactúan unos con otros, en contraposición a la visión tradicional en la cual un programa es una colección de subrutinas (funciones o procedimientos), o simplemente una lista de instrucciones para el computador.

Licencia GNU/GPL: En español Licencia Pública General, es una licencia creada por la Free Software Foundation y orientada principalmente a los términos de distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software Libre.