

Universidad de las Ciencias Informáticas

Facultad 3



Portal Web del Ministerio de Justicia de Cuba.

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

Autores: Arletis Francis Alitimón.

Nicole Almenares Serrano.

Tutor: MsC. Yarina Amoroso Fernández.

Co-tutor: Ing. Yoslenys Roque Hernández.

La Habana, 2014

“Año 56 de la Revolución”



“Se puede adquirir conocimientos y conciencia a lo largo de toda la vida, pero jamás en ninguna otra época de su existencia una persona volverá a tener la pureza y el desinterés con que, siendo joven, se enfrenta a la vida.”

Fidel Castro.

[Mensaje a los estudiantes universitarios de Cuba, el 3 de septiembre de 2010]

DECLARACIÓN DE AUTORÍA

Declaramos que somos las únicas autoras de este trabajo y autorizamos a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Nicole Almenares Serrano.

Firma del autor.

Arletis Francis Alitimón.

Firma del autor.

MsC. Yarina Amoroso Fernández

Firma del tutor.

Ing. Yoslenys Roque Hernández

Firma del co-tutor.

Agradecimientos

Nicole:

En nuestro corazón siempre llevamos a esas personas especiales que han significado todo en nuestra vida, por eso mi agradecimiento especial es para mi abuelo Pipa, quien fue un ejemplo a seguir y mi fortaleza cuando más lo necesitaba. Agradezco a mi mamá por compartir mis sueños y darme la confianza de luchar por alcanzarlos, por tantos desvelos y sacrificios, por tanto amor y principalmente por ser mi guía, que este sea el primero de muchos triunfos. A mi papá, por su confianza y apoyo, por ser mi mejor amigo y el mejor padre. A mi abuela Mami por criarme rodeada de tanto amor. A toda mi familia, especialmente a mi hermano, por cuidarme (aunque a veces en exceso) y por darme esas dos alegrías que son mis sobrinos; a mi tía Eulalia, a mi primo Franty y a mi padrastro por brindarme su ayuda en todo momento. A Rigo por estar a mi lado y ser mi apoyo en estos días tan difíciles, por su amor y su paciencia. A todas mis amistades, con las que he compartido momentos buenos y malos. A mis tutores por su tiempo y su paciencia, en especial a Yoslenys por su dedicación y preocupación constante. A mi compañera de tesis porque sin ella nada hubiera sido igual.

Arletis:

Agradezco a todos lo que de una manera u otra estuvieron allí cuando más los necesitaba, a los que sin esperar nada a cambio me dieron la mano para seguir adelante, me apoyaron en mis labores y compartieron conmigo en los momentos malos y buenos de estos cinco cursos. A mi mamá que es mi razón de ser en esta vida, gracias a ti mimi por la confianza y por estar ahí siempre para mí, por haber sacrificado lo que tienes para que no me falte nunca nada. A mi padre por apoyarme, ser mi guía y ejemplo a seguir. A mi hermana Arlenis que la quiero mucho y a mi hermano por su amor. A mis abuelas que aunque no estén hoy presentes siguen cada paso que doy en la vida. A mis tíos y tías por acogerme en sus hogares y por su apoyo durante la carrera. A mis hermanas y amigas Zoima, Zoemi, Dalili, Victoria, Selma, Ivette y Yanet que no se separaron de mí nunca y pude contar con ellas en cualquier momento. Doy gracias de corazón a mis tutores por ayudar a que este sueño se hiciera realidad y poder contar siempre con su disposición. A Nicole por ser la mejor compañera de tesis que pude pedir, que sin ella no sería posible este resultado a tan alta escala. A mis compañeros de aula por soportarme cada día y acogerme en su familia de amigos. Gracias a Aliú por enseñarme tantas cosas y por su apoyo absoluto. A todos les agradezco mucho y siempre los llevaré en mi corazón.

Dedicatoria

Nicole:

A mi madre, por ser este el primero de nuestros sueños en hacerse realidad.

A mi padre, por su apoyo incondicional.

A mi abuela Mami, por su amor infinito.

A mi abuelo Pipa, porque siempre esté orgulloso de mí... te llevo en mi corazón.

Arletis:

A mi mamá y a mi papá por todo el amor y el apoyo que siempre me han brindado, por ayudar a hacer realidad mis sueños y por confiar en mí.

A mis hermanos por su amor incondicional.

A mi familia por apoyarme y creer en mí.

Resumen

El Ministerio de Justicia es un organismo gubernamental que tiene entre sus ideas rectoras la ejecución de la política en el ámbito jurídico, para ejercer las facultades que le señala la ley. Actualmente dicho Ministerio cuenta con un portal web sencillo que no brinda los servicios suficientes a los ciudadanos y es por ello que en el presente trabajo de diploma se traza como objetivo principal desarrollar un portal web para la gestión de información y de trámites, que permita a los usuarios el acceso a la información detallada de las leyes y servicios que brinda el Ministerio de Justicia de la República de Cuba. Partiendo del estudio del estado del arte de sistemas homólogos de otros países quedaron seleccionadas las funcionalidades adecuadas que forman parte de la solución. Para guiar el desarrollo se empleó como metodología de investigación Programación Extrema, realizando los artefactos correspondientes a cada una de sus fases mediante el uso de las tecnologías y herramientas seleccionadas. Además, para garantizar la calidad del sistema se realizaron las validaciones correspondientes y se aplicaron las pruebas funcionales, obteniéndose como resultado el portal web que facilitará la gestión de los servicios y apoyará la divulgación y promoción del uso del gobierno electrónico, con la incorporación de la informática jurídica en el país.

Palabras claves: gobierno electrónico, gestión de los servicios, informática jurídica, Ministerio de Justicia, portal web.



Índice de contenidos

Agradecimientos	4
Dedicatoria	5
Resumen	6
Índice de figuras	10
Índice de tablas	11
Introducción	12
Capítulo 1: Fundamentación teórica	17
1.1 Introducción	17
1.2 Conceptos fundamentales asociados al dominio del problema	17
1.3 Tendencias actuales	18
1.4 Metodología de desarrollo de software	19
1.4.1 Programación Extrema	21
1.5 Lenguaje de modelado	24
1.5.1 UML	24
1.6 Herramientas CASE para UML	25
1.7 Sistema gestor de contenidos	26
1.7.1 Drupal	27
1.7.2 Joomla	27
1.7.3 Wordpress	28
1.8 Sistema gestor de base de datos	30
1.8.1 MySQL	30
1.9 Lenguajes de programación	31
1.9.1 PHP	31
1.9.2 HTML	32
1.9.3 CSS	32
1.9.4 Lenguajes de programación a utilizar	33
1.10 Servidor web Apache	33
1.11 Entorno de desarrollo integrado	34
1.11.1 Selección del entorno de desarrollo a utilizar	34
1.12 Conclusiones del capítulo	35



Capítulo 2: Propuesta de solución	36
2.1 Introducción	36
2.2 Descripción del sistema	36
2.2.1 Modelo de dominio	37
2.3 Requisitos de software	39
2.3.1 Requisitos funcionales	40
2.3.1.1 Métrica utilizada para la validación de requisitos	41
2.3.2 Requisitos no funcionales	42
2.3.3 Diagrama de interacción de los módulos	44
2.4 Fase de planificación	44
2.4.1 Historias de usuario	45
2.4.2 Plan de entregas	46
2.4.3 Plan de iteraciones	46
2.5 Fase de diseño	47
2.5.1 Arquitectura de la información	47
2.5.2 Arquitectura del sistema	49
2.5.2 Patrón arquitectónico	51
2.5.3 Patrones de diseño	52
2.5.4 Modelo de datos del módulo Trámites	54
2.5.5 Tarjetas CRC	55
2.6 Implementación	56
2.6.1 Estándares de codificación	56
2.6.2 Tareas de ingeniería por historia de usuario	56
2.6.3 Diagrama de despliegue	57
2.7 Conclusiones del capítulo	57
Capítulo 3: Validación de la solución	59
3.1 Introducción	59
3.2 Validación del diseño	59
3.2.1 Tamaño operacional de las clases (TOC)	59
3.2.2 Relaciones entre clases (RC)	60
3.3 Definición de la estrategia de prueba	61
3.4 Prueba de caja negra	62



Portal web del Ministerio de Justicia de la República de Cuba.

3.4.1 Técnica de partición de equivalencia -----	62
3.4.2 Diseño de caso de prueba-----	63
3.4.3 Resultados de las pruebas de caja negra -----	64
3.5 Prueba de caja blanca-----	65
3.5.1 Técnica de prueba: Camino básico -----	66
3.6 Pruebas de carga y estrés-----	67
3.6.1 Resultados de las pruebas de carga y estrés -----	67
3.7 Prueba de usabilidad -----	69
3.7.1 Resultados de la prueba de usabilidad-----	69
3.8 Prueba de seguridad-----	70
3.8.1 Resultados de la prueba de seguridad -----	70
3.9 Validación de la variable de la investigación -----	71
3.10 Conclusiones del capítulo -----	73
Conclusiones generales-----	74
Recomendaciones-----	75
Bibliografía -----	76
Glosario -----	80



Índice de figuras

Figura 1: Ciclo de vida de XP.	23
Figura 2: Modelo de dominio del sistema.	38
Figura 3: Diagrama de interacción de módulos.	44
Figura 4: Arquitectura en tres niveles del CMS Joomla.	50
Figura 5: Línea de código del patrón de la Fábrica.	53
Figura 6: Línea de código del patrón Instancia única.	53
Figura 7: Modelo de datos del módulo Trámites.	55
Figura 8: Diagrama de despliegue.	57
Figura 9: Comportamiento de los atributos de calidad de la métrica TOC.	60
Figura 10: Comportamiento de los atributos de calidad de la métrica RC.	61
Figura 11: NC detectadas y resueltas por iteración.	65
Figura 12: NC resueltas con éxito.	65
Figura 13: Grafo de flujo.	66
Figura 14: Resultados del Jmeter.	68
Figura 15: Resultados de la prueba de usabilidad.	69



Índice de tablas

Tabla 1: Comparación entre metodologías ágiles.-----	21
Tabla 2: Estadísticas extraídas de Google Trends. -----	29
Tabla 3: Requisitos funcionales del MINJUS.-----	40
Tabla 4: HU#4 Gestionar noticia. -----	45
Tabla 5: HU#6 Gestionar datos del directorio.-----	45
Tabla 6: HU#15 Gestionar trámites.-----	46
Tabla 7: Plan de entregas. -----	46
Tabla 8: Plan de iteraciones.-----	47
Tabla 9: Tarjeta CRC de la clase Trámite.-----	56
Tabla 10: Tarea de ingeniería#4 Validar formularios correspondientes a los trámites. -----	57
Tabla 11: Comportamiento de los atributos de calidad por clases de la métrica TOC. -----	60
Tabla 12: Comportamiento de los atributos de calidad por clases de la métrica RC. -----	61
Tabla 13: Caso de prueba#16: Gestionar trámites. -----	64
Tabla 14: Caminos linealmente independientes.-----	67
Tabla 15: Salidas del informe agregado.-----	68
Tabla 16: Variables dependientes de la investigación. -----	71
Tabla 17: Medición de los indicadores.-----	72



Introducción

Desde finales del siglo pasado las Tecnologías de la Informática y las Comunicaciones (TIC) se han venido consolidando como uno de los principales pilares de la sociedad, estando presente prácticamente en todo el quehacer del hombre. Las TIC han suscitado transformaciones en la economía, la cultura y la sociedad en general, influyendo en la rama del derecho. De esta manera surge la informática jurídica, disciplina que aplica la tecnología a la ciencia del derecho para reducir los problemas jurídicos y contribuir a la sistematización y automatización de la información jurídica (1). Esto trae consigo un ahorro de esfuerzo, tiempo y un aumento de la calidad en el desarrollo de los procesos jurídicos.

Para llevar a cabo la automatización del conocimiento, de las fuentes de producción y los procesos de administración jurídica del país, la Universidad de Ciencias Informáticas (UCI) cuenta con el Centro de Gobierno Electrónico (CEGEL). Este centro tiene como línea principal de trabajo la informática jurídica, dándole continuidad al uso del gobierno electrónico en el país y a la implantación de herramientas que permitan mejorar la eficiencia y eficacia de los procesos internos que se manejan en estas instituciones, logrando además su vinculación con la sociedad.

Este centro contiene estrecha relación en su trabajo con el MINJUS¹ de Cuba, organismo de la Administración Central del Estado que tiene como misión asistir al Gobierno en la preparación de la política jurídica en el ámbito de su competencia y una vez aprobada, participar en su ejecución y control. Ofrece al ciudadano servicios como un Centro de información y permite realizar trámites con respecto a cada uno de los registros existentes: Registro de la Propiedad, Registro Mercantil, Registro del Estado Civil, Registro Central de Sancionados y Registro de Actos y Última Voluntad; además de ofrecer servicios de consultoría y recepción de quejas.

Actualmente estos servicios de consultoría y recepción de quejas a la población no se resuelven con la prontitud que se necesita debido al cúmulo de las mismas. A esto se le suma la insuficiente información que existe relacionada con las Oficinas Registrales del MINJUS de Cuba (localización, direcciones, teléfonos, correo electrónico, horarios de atención y ámbitos de jurisdicción) y con los trámites que se realizan en las mismas, lo cual conlleva a engorrosas

¹ Ministerio de Justicia.



Portal web del Ministerio de Justicia de la República de Cuba.

gestiones legales. Por otra parte, existe la necesidad de realizar una serie de encuestas con el objetivo de medir la calidad de los servicios que se brindan.

Actualmente el MINJUS de Cuba posee una interfaz de información estática que no satisface las necesidades de los usuarios en el ámbito nacional e internacional y no propicia la comunicación con las personas y entidades interesadas en conocer del ámbito jurídico en el país. Además, no es posible dar seguimiento a los estados por los que transitan los trámites registrales y notariales realizados por parte de los ciudadanos; lo que trae consigo la falta de calidad y rapidez a la hora de realizarlos, así como el costo asociado. Tampoco se brinda un servicio eficiente de consulta legal en línea y la información que se presenta no se actualiza periódicamente.

La problemática antes descrita ha generado la necesidad de desarrollar una investigación que dé respuesta al siguiente **problema a resolver**:

¿Cómo mejorar los servicios de gestión de la información y de trámites que ofrece el MINJUS de la República de Cuba para cubrir las principales necesidades de los usuarios?

El **objeto de estudio** de la investigación es el desarrollo de aplicaciones web para sistemas de gestión de información de entidades de gobierno electrónico.

Para dar solución al problema planteado se define como **objetivo general** desarrollar un portal web para la gestión de información y de trámites que ofrece el MINJUS de la República de Cuba, que permita cubrir las principales necesidades de los usuarios.

Definiendo como **campo de acción** los sistemas de gestión de información electrónica para el MINJUS de la República de Cuba.

Como **idea a defender** se plantea que si se desarrolla el portal web para el MINJUS de la República de Cuba, se puede mejorar la gestión de la información y de los trámites en el ámbito jurídico, cubriendo las principales necesidades de los usuarios.

Del objetivo general se derivan los siguientes **objetivos específicos**:

- ✓ Elaborar el marco teórico-referencial de la investigación.
- ✓ Realizar el levantamiento de las funcionalidades para el desarrollo del portal web.
- ✓ Realizar el análisis y diseño del portal web.
- ✓ Realizar la implementación del portal web.



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ Validar el correcto funcionamiento de la herramienta y que la solución propuesta dé cumplimiento al objetivo general de la investigación.

Para dar cumplimiento a los objetivos específicos se trazan las siguientes **tareas de investigación:**

- ✓ Estudio del estado del arte referente al tema propuesto.
- ✓ Selección de la metodología de desarrollo de software y las herramientas para el desarrollo del portal web.
- ✓ Selección del lenguaje de modelado.
- ✓ Identificación y validación de los requisitos funcionales y no funcionales a tener en cuenta para el desarrollo del portal web.
- ✓ Especificación y evaluación de funcionalidades.
- ✓ Elaboración del marco teórico-referencial de la investigación.
- ✓ Identificación y fundamentación de los patrones arquitectónicos.
- ✓ Definición y validación de la arquitectura a utilizar para el desarrollo del portal.
- ✓ Realización de los diagramas de modelo de dominio, de despliegue y de interacción de módulos.
- ✓ Implementación de los diferentes módulos del sistema.
- ✓ Diseño de los casos de prueba, realización de las pruebas y validación de la solución.
- ✓ Redacción del informe final.

Los métodos de investigación son la forma de abordar la realidad, de estudiar la naturaleza, la sociedad y el pensamiento, con el propósito de descubrir su esencia y sus relaciones. Estos se clasifican en teóricos y empíricos, los cuales están dialécticamente relacionados (2). A continuación se exponen los métodos utilizados en el desarrollo de la presente investigación.

Métodos teóricos:

Histórico-Lógico: se utilizó para el análisis de los documentos necesarios referentes a los trámites registrales y notariales, sus antecedentes y las soluciones utilizadas anteriormente. Lo que permitió realizar un estudio del estado del arte de la investigación que sirviera de base al desarrollo de la misma.

Análítico-Sintético: mediante este método se determinaron los elementos bases para la implementación del sistema, como las tecnologías y estándares a utilizar, además de las metodologías y herramientas.



Portal web del Ministerio de Justicia de la República de Cuba.

Modelación: se utilizó para realizar el modelado del sistema, su análisis y diseño.

Métodos empíricos:

Entrevista: fueron realizadas a funcionarios del MINJUS de Cuba con el objetivo de reunir información relevante para el desarrollo del sistema y conocer todo lo referente a los trámites registrales y notariales del Ministerio en el país. Las entrevistas se realizaron con la intención de ir fomentando la comunicación con el cliente. El tipo de entrevista realizada es la estructurada, donde se elaboró un cuestionario con preguntas que sirvieran de apoyo al desarrollo de la investigación.

Medición: se utilizó para medir los atributos de calidad que posee el software, además para obtener información numérica sobre el grado del esfuerzo realizado en el momento de efectuar las pruebas.

Observación: permitió obtener conocimiento acerca del comportamiento del objeto de investigación tal y como este se da en la realidad, es una forma de obtener la información directa e inmediata sobre el fenómeno u objeto que está siendo investigado (3).

El presente trabajo estará conformado por tres capítulos, los cuales abordan los siguientes temas:

CAPÍTULO 1: Fundamentación teórica.

En este capítulo se muestra el resultado del estudio de los portales web que brindan información sobre los Ministerios de Justicia en diferentes países. Se abordan los principales conceptos asociados al dominio del problema, se expone la metodología que guiará todo el proceso de desarrollo del software y las herramientas y tecnologías empleadas.

CAPÍTULO 2: Propuesta de solución.

En este capítulo se describen las principales características del sistema a desarrollar. Se realiza un levantamiento de los requisitos funcionales especificados a través de las historias de usuarios y de los no funcionales. Se detalla la interacción entre los principales módulos, los usuarios y el contenido del sistema. De forma general, se generan los artefactos y diagramas correspondientes a las fases de la metodología seleccionada.

CAPÍTULO 3: Validación y prueba.



Portal web del Ministerio de Justicia de la República de Cuba.

En este capítulo se muestran los resultados de la aplicación de las métricas para validar el diseño del portal. Se realiza la validación del sistema, estableciendo una estrategia de prueba que se ejecuta a través de técnicas y métodos, presentándose los resultados obtenidos.



Capítulo 1: Fundamentación teórica

1.1 Introducción

En este capítulo se muestra el resultado del estudio del estado del arte de los portales web que brindan información sobre los Ministerios de Justicia en diferentes regiones. Se referencian además los principales conceptos asociados al dominio del problema y se exponen la metodología que guiará todo el proceso de desarrollo del software, el lenguaje de modelado empleado por la misma, los lenguajes de programación, el Entorno de Desarrollo Integrado (IDE), los Sistemas Gestores de Contenido y de Bases de Datos utilizados; así como la herramienta de Ingeniería del Software Asistida por Computadora (CASE, por su nombre en inglés) empleada en el desarrollo del portal.

1.2 Conceptos fundamentales asociados al dominio del problema

MINJUS: es el organismo encargado de asistir al gobierno en la preparación y ejecución de la política en el ámbito jurídico; asesorar en el proceso legislativo; ejercer las facultades que le señala la ley respecto a los tribunales provinciales y municipales; ejercer la dirección y el control técnico, normativo y metodológico de las actividades a su cargo y promover el desarrollo de la actividad jurídica (4).

Informática jurídica: conjunto de aplicaciones de la informática en el ámbito jurídico; es una técnica interdisciplinaria que tiene por objeto el estudio e investigación de los conocimientos aplicables a la recuperación de información jurídica, así como la elaboración y aprovechamiento de los instrumentos de análisis y tratamiento de información jurídica, necesaria para lograr dicha recuperación (5).

Gobierno electrónico: es la aplicación de las tecnologías de la informática y las comunicaciones (TIC) al conocimiento de los procesos internos de gobierno, la entrega de servicios al Estado y al funcionamiento del sector público. Tiene como objetivo rector incrementar la eficiencia, la transparencia y la participación ciudadana. Se basa en la implantación de herramientas como portales web, buscando mejora en los procesos estatales y el vínculo con la sociedad (6).

Aplicaciones web: aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador. En otras palabras,



Portal web del Ministerio de Justicia de la República de Cuba.

es una aplicación (software) que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador (7).

Portal web: es un sitio web que ofrece, de forma integrada, una amplia variedad de servicios y recursos al usuario. Por lo general, los portales ofrecen noticias actualizadas al instante, buscadores, foros para compartir opiniones, chat, juegos online, tiendas virtuales para realizar compras electrónicas y servicio de correo electrónico (8).

Ingeniería de software: es una disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo. Es el estudio de los principios y metodologías para el desarrollo y mantenimiento del software a fin de lograr que sea fiable (9).

Ingeniería de requisitos: cumple un papel primordial en el proceso de producción de software, ya que da la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; minimizando los problemas relacionados por la mala gestión de los requisitos en el desarrollo de sistemas (10).

Arquitectura de la información: es una interdisciplina que se nutre de diferentes áreas del conocimiento. El rol de arquitecto de información dentro de los equipos de desarrollo de software permite realizar productos que se adapten a las necesidades y expectativas de los clientes (11).

1.3 Tendencias actuales

El MINJUS es el organismo encargado de asistir al Gobierno en el ámbito jurídico, además de ejercer la dirección y el control normativo y metodológico de la actividad jurídica. Con el desarrollo de la tecnología y el internet, en la mayoría de los países existen portales web para mejorar el trabajo de las instituciones y entidades que se encargan de promover información sobre la justicia. Estos portales tienen como principal objetivo difundir las principales tareas e información relacionadas con la justicia en el país y brindan varios espacios y servicios. Además de que poseen varias funcionalidades y elementos en común que solo se diferencian de acuerdo a su forma de trabajo y a las leyes y normativas específicas de cada nación.

Realizando un estudio de los portales web de algunos países iberoamericanos como: Argentina, Bolivia, Ecuador, España, Venezuela y Perú, se encuentra que en su mayoría exponen información explícita sobre el MINJUS de su país, la visión, la misión, el organigrama,



Portal web del Ministerio de Justicia de la República de Cuba.

autoridades, funciones y estructura, cada una de ellas explicadas con gran envergadura. Brindan información y permiten ejecutar trámites, gestiones personales, programas de atención, documentos y certificados, de acuerdo al lugar en que se encuentre; los cuales sirven de ayuda a la hora de realizar una gestión de forma rápida y segura. Permiten además, mantener al tanto al usuario respecto a la actualidad noticiosa en los diferentes medios de comunicación, galería de imágenes y multimedias relacionadas con el Ministerio. Ofrecen al ciudadano el acceso a reglamentos, resoluciones, leyes y decretos supremos. Realizan consultas en línea sobre la nacionalidad y temas relacionados con cada uno de los registros. Además, mantienen integridad con otros portales web de estrecha relación con el Ministerio y el Gobierno.

En Cuba se han hecho varios intentos por establecer un portal web que satisfaga las necesidades de usuarios nacionales e internacionales. En la actualidad no se ha concretado un resultado satisfactorio, el cual propicie la comunicación con las personas y entidades interesadas en conocer del ámbito jurídico en el país y que permita realizar trámites en línea; además de favorecer la calidad y retroalimentación del propio organismo respecto a los procesos que le conciernen.

Con el estudio realizado se decide utilizar funcionalidades que permitan al usuario acceder a información relevante sobre el organismo, conocer sobre el acontecer noticioso relacionado con la justicia, efectuar consultas en línea sobre temas de interés y realizar trámites de solicitudes de certificaciones; todas orientadas hacia las leyes y normativas del MINJUS de la República de Cuba.

1.4 Metodología de desarrollo de software

La metodología es una parte del proceso de investigación o método científico, que sigue a la propedéutica², permite sistematizar los métodos y las técnicas necesarias para llevarla a cabo. Son un conjunto de procedimientos racionales utilizados para alcanzar una gama de objetivos que rigen en una investigación científica. Ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto (software) surgen las metodologías de desarrollo de software. Estas se clasifican en dos tipos: tradicionales y ágiles.

²Enseñanza preparatoria para el estudio de una disciplina.



Portal web del Ministerio de Justicia de la República de Cuba.

Las metodologías tradicionales se centran en la definición detallada de los procesos y tareas a realizar, las herramientas a utilizar y requiere una extensa documentación, ya que pretende prever todo de antemano. Este tipo de metodología es más eficaz y necesaria cuando se tiene un proyecto de larga duración donde se deben planificar las fases del mismo para ahorrar tiempo y recursos. Otro rasgo que caracteriza a este tipo de metodología es que necesita de gran cantidad de participantes, pues requiere un equipo de trabajo capaz de administrar el proceso en sus diferentes etapas (12).

Las metodologías ágiles se encargan de valorar al individuo y las iteraciones del equipo más que a las herramientas o los procesos utilizados, es más importante crear un producto que funcione que escribir mucha documentación, el cliente está en todo momento colaborando en el proyecto y es más importante la capacidad de respuesta ante un cambio realizado que el seguimiento estricto de un plan (12).

A partir de lo analizado y teniendo en cuenta que el producto a elaborar está formado por un grupo de trabajo de 2 integrantes, con la probabilidad de que las funcionalidades sean cambiantes debido a las características del cliente y observando además el uso de las metodologías ágiles con resultados satisfactorios de otros centros y proyectos que se dedican a desarrollar portales web en la universidad, se decide emplear una metodología ágil.

A continuación, en la tabla 1, se expone una comparación entre las metodologías ágiles más utilizadas (13).

Metodología ágil:	Objetivo:	Características:	Ciclo de vida:
Programación Extrema (XP)	Centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software.	<ol style="list-style-type: none">1. Tratamiento de las historias de usuario de forma dinámica y flexible.2. El período de desarrollo es corto, en dependencia del proyecto que se elabora.3. El cliente forma parte del equipo de desarrollo.4. Las dimensiones del proyecto son pequeñas.	Consta de cuatro fases: <ol style="list-style-type: none">1. Planificación.2. Diseño.3. Implementación o codificación.4. Pruebas



Portal web del Ministerio de Justicia de la República de Cuba.

<p>Scrum</p>	<p>Maximizar la realimentación sobre el desarrollo pudiendo corregir problemas y mitigar riesgos de forma temprana.</p>	<ol style="list-style-type: none"> 1. Define un proceso empírico, iterativo e incremental de desarrollo que intenta obtener ventajas respecto a los procesos definidos. 2. Suministra un marco de administración basado en patrones organizacionales. 3. Especialmente indicada para proyectos con un rápido cambio de funcionalidades. 4. Se realizan reuniones a lo largo del proyecto. 	<p>Consta de cuatro fases:</p> <ol style="list-style-type: none"> 1. Planeamiento 2. Montaje. 3. Desarrollo. 4. Liberación.
<p>Crystal Clear</p>	<p>Centrada en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos.</p>	<ol style="list-style-type: none"> 1. Maneja iteraciones cortas con retroalimentación frecuente por parte de los usuarios/clientes, minimizando de esta forma la necesidad de productos intermedios. 2. Es más fácil de aprender e implementar. 	<p>Enfatiza el proceso como un conjunto de ciclos anidados. En la mayoría de los proyectos se perciben siete ciclos: el proyecto, el de entrega de una unidad, la iteración, la semana laboral, el período de integración y el episodio de desarrollo de una sección de código.</p>

Tabla 1: Comparación entre metodologías ágiles.

Al analizar las características fundamentales de cada una de las metodologías ágiles mencionadas anteriormente se decide hacer uso de la metodología XP, a continuación se presentan los elementos que sirvieron de utilidad a la hora de hacer la elección.

1.4.1 Programación Extrema-Extreme Programming (XP) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de



Portal web del Ministerio de Justicia de la República de Cuba.

software, promoviendo el trabajo en equipo. Se basa en una retroalimentación continua entre el cliente y el equipo de desarrollo, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico (14).

La programación extrema es una metodología de desarrollo ligera basada en una serie de valores y una docena de prácticas que propician un aumento en la productividad a la hora de generar software. Los valores representan aquellos aspectos que se consideran fundamentales para garantizar el éxito de un proyecto de desarrollo de software (15). Los cuatro valores de XP son:

1. Comunicación: es muy importante que haya una comunicación constante con el cliente y dentro de todo el equipo de trabajo, de esto dependerá que el desarrollo se lleve a cabo de una manera sencilla, entendible y que se entregue al cliente lo que necesita.
2. Simplicidad: el diseño debe ser sencillo y amigable al usuario, el código debe ser simple y entendible, programando sólo lo necesario y lo que se utilizará.
3. Retroalimentación: es la comunicación constante entre el desarrollador y el usuario.
4. Coraje: se refiere a la valentía que se debe tener al modificar o eliminar el código que se realizó con tanto esfuerzo; el desarrollador debe saber cuándo el código que desarrolló no es útil en el sistema y, por lo mismo, debe ser eliminado. También se refiere a tener la persistencia para resolver los errores en la programación.

A continuación se exponen las prácticas llevadas a cabo en el desarrollo de la presente investigación:

1. Planificación incremental: permite obtener rápidamente un plan inicial y contar con mecanismos que permitan conocer con precisión la situación actual del proyecto. Como es lógico, la planificación es iterativa: un representante del negocio decide al comienzo de cada iteración qué características concretas se van a implementar.
2. Pruebas: se realizarán pruebas de unidad así como de aceptación, con el objetivo de prevenir los errores, para garantizar que el código haga lo que debe hacer. El cliente es el responsable de definir las pruebas de aceptación, no necesariamente de implementarlos. Es la persona mejor calificada para decidir cuál es la funcionalidad más valiosa.
3. Programación en parejas: las decisiones se toman entre el grupo de trabajo de dos personas, proporcionando un mecanismo de seguridad enormemente valioso. Con dos



A partir de ellas y de la arquitectura que se utilizará se planifican las entregas, así como los objetivos de cada una y las iteraciones con que contará.

La principal ventaja que brinda esta metodología es precisamente la poca documentación que se genera durante el desarrollo del proyecto, lo cual permite un aprovechamiento al máximo del tiempo disponible para la elaboración del mismo; sin embargo, si el proyecto fuera muy extenso, al tener poca documentación, se volvería algo difícil su mantenimiento o continuidad. Esta metodología además:

- ✓ Asume que la planificación nunca será perfecta y que los requisitos cambian a lo largo de todo el ciclo de vida de la aplicación según varíen las necesidades del negocio.
- ✓ El período de desarrollo es corto, en dependencia del proyecto que se elabora.
- ✓ El cliente forma parte del equipo de desarrollo.
- ✓ Las dimensiones del proyecto son pequeñas.
- ✓ Promueve un entorno físico con un ambiente que permite la comunicación y colaboración entre todos los miembros del equipo de trabajo.

1.5 Lenguaje de modelado

Un lenguaje de modelado es un lenguaje artificial diseñado para expresar modelos. Ya que los modelos habitualmente se muestran en forma de diagramas por comodidad, los lenguajes de modelado suelen incorporar notaciones gráficas. Igual que los lenguajes naturales, los lenguajes de modelado poseen un léxico, es decir, un conjunto de palabras que existen en el lenguaje; y una sintaxis, conjunto de reglas que expresan cómo se pueden combinar dichas palabras para componer "frases" que tengan sentido (17).

1.5.1 UML

El Lenguaje Unificado de Modelado³ es un método formal de modelado y un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software determinada (18).

UML tiene como funciones fundamentales:

- ✓ Visualizar: permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.

³ UML: Unified Modeling Language



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ Especificar: permite especificar cuáles son las características de un sistema antes de su construcción.
- ✓ Construir: a partir de los modelos especificados se pueden construir los sistemas diseñados.
- ✓ Documentar: los propios elementos gráficos sirven como documentación del sistema desarrollado para su futura revisión.

Para la solución del problema se utiliza UML en su versión 2.0 que tiene como principales ventajas (19):

- ✓ Mayor rigor en la especificación.
- ✓ Permite realizar una verificación y validación del modelo realizado.
- ✓ Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos). Esto permite que el modelo y el código estén actualizados, con lo que siempre se puede mantener la visión en el diseño de más alto nivel de la estructura de un proyecto.
- ✓ Contiene muchas novedades y cambios que se centran en resolver carencias prácticas.

1.6 Herramientas CASE para UML

Las herramientas CASE⁴ son diversas aplicaciones informáticas destinadas a aumentar la productividad, reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software, en tareas como el diseño de proyectos, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática y documentación o detección de errores (20).

Visual Paradigm es una herramienta **CASE** para **UML** visual que soporta el ciclo de vida completo del desarrollo de software. Permite realizar un diseño enfocado al negocio que genera un software de mayor calidad y posee capacidades de ingeniería directa e inversa.

Esta herramienta tiene un diseño enfocado al negocio que permite el modelado de base de datos y la confección de los prototipos de interfaz. Brinda el uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación. Tiene la disponibilidad de integrarse en los principales Entornos de Desarrollo Integrado (21).

⁴Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadoras.



Portal web del Ministerio de Justicia de la República de Cuba.

Ventajas de Visual Paradigm:

- ✓ Es una herramienta multiplataforma de modelado visual UML.
- ✓ Tributa una excelente interoperabilidad con otras herramientas CASE.
- ✓ Permite la captura de funcionalidades, análisis, diseño e implementación, también proporciona características tales como generación del código, ingeniería inversa y generación de informes.
- ✓ Permite dibujar todos los tipos de diagramas.
- ✓ Apoya los estándares más recientes de las notaciones de UML.

Teniendo en cuenta la importancia del uso del software libre en Cuba, la existencia de una licencia UCI para el uso de Visual Paradigm y las características de esta herramienta, se selecciona la misma en su versión 8.0 para realizar el diseño UML del portal web para el MINJUS de Cuba. Además de que ofrece a los desarrolladores una plataforma intuitiva para construir aplicaciones con calidad en un menor tiempo.

1.7 Sistema gestor de contenidos

Un Sistema Gestor de Contenido (Content Management System o CMS) es un programa que permite crear una estructura de soporte para la creación y administración de contenidos de páginas web. Plataforma que contiene un conjunto de herramientas de software que controla una o varias bases de datos donde se aloja el contenido. Permite gestionar de forma uniforme y accesible un sitio web dinámico, proporcionando un entorno que posibilita la presentación, mantenimiento, ampliación y actualizaciones periódicas, con la colaboración de múltiples usuarios ya sea a través de Internet o Intranet, que acceden al gestor a través del navegador web (22).

Entre sus principales características se encuentra que brinda cuatro funcionalidades principales: creación, gestión, publicación y presentación de contenido. Además tiene incorporados estándares como CSS para la creación de estilos, XML como lenguaje de marcado para estructurar documentos, WAI⁵ que permite asegurar la accesibilidad del sistema y RSS⁶ para syndicar⁷ contenidos de tipo noticia.

⁵ Web Accessibility Initiative, Iniciativa de Accesibilidad Web.

⁶ Really Simple Syndication, Distribuidora de Noticias Realmente Simple.

⁷ Facilitar el contenido para que otras web puedan incorporarlos.



Portal web del Ministerio de Justicia de la República de Cuba.

Algunos de los CMS más utilizados a escala mundial son:

1.7.1 Drupal

Es un sistema de administración de contenido para sitios web que fue originalmente escrito por Dries Buytaert. Poderoso CMS muy conocido por la calidad de su código y por la seguridad que brinda, es estable y de actualización continua. Permite publicar artículos, imágenes u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos (23). Drupal ha llegado a ser mucho más que sólo un portal de noticias gracias a su arquitectura flexible. Tiene una capa básica que soporta los módulos que permiten comportamientos adicionales. Los módulos disponibles para Drupal proveen un amplio surtido de características, incluyendo sistemas de comercio electrónico, galerías de fotos y administración de listas de correo electrónico. Este se usa en la intranet de compañías, enseñanza en línea, comunidades de arte y administración de proyectos de gran envergadura (24).

Contiene una serie de características, de ellas las más importantes son:

- ✓ Ayuda on-line: un robusto sistema online de páginas de ayuda para los distintos módulos, que sirve de apoyo para usuarios y para administradores.
- ✓ Búsqueda: todo su contenido es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- ✓ Código abierto: su código fuente está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de 'blogs' o de gestión de contenido propietarios, es posible extenderlo o adaptarlo según las necesidades.
- ✓ Módulos: la comunidad de Drupal ha desarrollado módulos que proporcionan funcionalidades como página de categorías y mensajes privados.
- ✓ Personalización: un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- ✓ URLs amigables: Drupal usa el mod_rewrite de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.

1.7.2 Joomla

Joomla es un CMS que permite crear sitios web de alta interactividad, profesionalidad y eficiencia. Su administración está enteramente basada en la gestión en línea de contenidos,



Portal web del Ministerio de Justicia de la República de Cuba.

porque todas las acciones que realizan los administradores, ya sea para modificar, agregar o eliminar contenidos, se realizan exclusivamente mediante un navegador web.

Entre sus principales virtudes está la de permitir editar el contenido de un sitio web de manera sencilla. Es una aplicación de código abierto programada mayoritariamente en PHP bajo una licencia GPL. Este administrador de contenidos puede trabajar en Internet o Intranets y requiere de una Base de datos MySQL, así como, preferiblemente, de un Servidor HTTP Apache (25).

Una de las mayores potencialidades que tiene este CMS es la gran cantidad de extensiones existentes programadas por su comunidad de usuarios, que aumentan sus posibilidades con nuevas características y se integran fácilmente.

Está diseñado para sitios web donde las personas no necesitan gran conocimiento de la tecnología, es más fácil de configurar y poner en marcha con una curva de aprendizaje relativamente baja; es significativamente potente y flexible además de poseer un rendimiento y escalabilidad aceptable para aplicaciones no muy grandes. Las funcionalidades de edición y construcción del contenido son excelentes y la consola de administración es mucho más fácil para organizar y encontrar contenido. Joomla posee además una interfaz amigable en cuanto a diseño y aspecto estético.

Está desarrollado en una arquitectura Modelo-Vista-Controlador lo que permite:

- ✓ Interactuar directamente con la parte de vista de Joomla.
- ✓ Sobrescribir desde la plantilla de Joomla la parte de vista de un componente, módulo o complemento, lo cual permite un gran nivel de personalización en el desarrollo de las plantillas.

Con Joomla es posible crear sitios web corporativos, intranets, páginas personales, pequeños sitios de negocios y aplicaciones de gestión pública, permitiendo realizar publicaciones en línea como de revistas y periódicos.

1.7.3 Wordpress

CMS enfocado a la creación de blogs y desarrollado en PHP y MySQL, bajo licencia GPL y código modificable, se ha convertido en el más popular de la Blogosfera. Las causas de su enorme crecimiento son, entre otras, su licencia, su facilidad de uso y sus características como gestor de contenidos. Tiene soporte para HTML en los títulos de las imágenes. Es de fácil instalación,



Portal web del Ministerio de Justicia de la República de Cuba.

actualización y personalización y brinda la capacidad de crear páginas estáticas. Hace posible controlar y moderar todos los blogs desde un solo entorno de administración.

Algunas de las ventajas que brinda este gestor de contenidos son:

1. Fácil implementación, aprendizaje y adaptación en relación con otros CMS.
2. Infinidad de plantillas disponibles en la web, de tipo gratis y de pago.
3. Administrador muy amigable para el usuario.
4. Gran comunidad de soporte.
5. Infinidad de complementos (plugins), por lo que puedes expandir tu sitio web sin saber programar o contratar servicios de terceros (26).

A continuación, en la tabla 2 se exponen algunas estadísticas extraídas de Google Trends, de acuerdo al uso de los distintos sistemas de gestión de contenidos y el volumen de búsquedas en esos países:

País	Drupal	Joomla	WordPress
Chile	0	1	0.82
Francia	0.2	1	0.68
Alemania	0.16	1	0.61
Italia	0.17	1	0.6
Polonia	0.15	1	0.43
Federación Rusa	0.3	1	0.5
España	0.26	1	1
Suiza	0.15	1	0.45
Reino Unido	0.4	1	1.65
EE.UU	0.66	1	2.3
China	0.8	1	3.2

Tabla 2: Estadísticas extraídas de Google Trends.

Tomando en consideración el análisis de los CMS antes expuestos y basados en las estadísticas de usabilidad, se escoge Joomla en su versión 2.5.17 como sistema gestor de contenidos para el desarrollo del portal web. El mismo permitirá realizar un sitio web dinámico para el MINJUS de Cuba que suplantarán al existente. Este CMS contiene funcionalidades específicas del gestor de contenido de Joomla como las prestaciones de extensiones, módulos y componentes que se pueden instalar e incluir en el desarrollo del portal. Ofrece además gran flexibilidad para que los



desarrolladores puedan integrar nuevos módulos según las necesidades pactadas con el cliente y contiene un alto rendimiento en portales web de alto tráfico de usuarios. Otro criterio que se tuvo en cuenta es que Joomla se utiliza para sistemas que no van a crecer en número de prestaciones ni secciones. Con este CMS se realizará una web sencilla y acorde a las necesidades del cliente.

1.8 Sistema gestor de base de datos

Un Sistema Gestor de Base de Datos (SGBD, en inglés DBMS: DataBase Management System) es 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 mismos, ya sea de forma interactiva o a través de un lenguaje de programación.

1.8.1 MySQL

Es un SGBD relacionales rápido, sólido y flexible. Idóneo para la creación de bases de datos con acceso desde páginas web dinámicas, así como para la creación de cualquier otra solución que implique el almacenamiento de datos, posibilitando realizar múltiples y rápidas consultas. Está desarrollado en C y C++, facilitando su integración en otras aplicaciones desarrolladas también en esos lenguajes.

Es un sistema cliente/servidor, por lo que permite trabajar como servidor multiusuario y de subprocesamiento múltiple, o sea, cada vez que se crea una conexión con el servidor, el programa servidor establece un proceso para manejar la solicitud del cliente, controlando así el acceso simultáneo de un gran número de usuarios a los datos y asegurando el acceso a usuarios autorizados solamente (27).

Características principales de MySQL:

- ✓ Mayor velocidad y robustez.
- ✓ Soporta gran cantidad de tipos de datos para las columnas.
- ✓ Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- ✓ Cada base de datos cuenta con 3 archivos: uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- ✓ Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo⁸.

⁸ Multitarea, o sea, puede hacer varios hilos o tareas, dígame procesos, simultáneamente.



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ Flexible sistema de contraseñas y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- ✓ El servidor soporta mensajes de error en distintas lenguas.

Para el desarrollo del sistema se ha decidido utilizar MySQL en su versión 5.5.16, ya que es el SGBD propio de Joomla, además de que es totalmente gratuito bajo la licencia GPL y proporciona a los desarrolladores y administradores todo lo que necesitan para desarrollar BD⁹ para aplicaciones web con gran éxito. Se desarrolla y actualiza de forma más rápida que muchos de sus rivales, soporta lenguajes distintos de programación como PHP y se integra con la mayor parte de los entornos de programación.

1.9 Lenguajes de programación

Un lenguaje de programación es un lenguaje artificial que se utiliza para expresar programas de ordenador. Está formado por un conjunto de símbolos, palabras claves utilizables y por reglas gramaticales para construir sentencias sintácticas y semánticas, que definen su estructura y el significado de sus elementos y expresiones (28).

1.9.1 PHP

Es un lenguaje de programación utilizado para la creación de sitios web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, se caracteriza por ser un lenguaje gratuito y multiplataforma. Además de su posibilidad de acceso a distintos tipos de bases de datos, también es importante destacar su capacidad de crear páginas dinámicas, así como la posibilidad de separar el diseño del contenido de una web. PHP es la solución para la construcción de webs con independencia de la base de datos y del servidor web, válido para cualquier plataforma (29).

Presenta cuatro grandes características:

1. Velocidad: no solo es rápido al ser ejecutado sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema. PHP se integra muy bien junto a otras aplicaciones.
2. Estabilidad: utiliza su propio sistema de administración de recursos y posee de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
3. Seguridad: maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo **.ini**.

⁹ Bases de Datos.



Portal web del Ministerio de Justicia de la República de Cuba.

4. Simplicidad: usuarios con experiencia en C y C++ podrán utilizarlo rápidamente. Además de que dispone de una amplia gama de bibliotecas, y permite la posibilidad de agregarle extensiones. Esto le permite su aplicación en múltiples áreas, tales como encriptado y gráficos.

PHP es un poderoso lenguaje e intérprete capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Está diseñado específicamente para ser un lenguaje más seguro para escribir programas CGI¹⁰ que Perl o C, con la selección correcta de opciones de configuración en tiempos de compilación y ejecución y siguiendo algunas prácticas correctas de programación. Fue diseñado originalmente para la creación de páginas web dinámicas.

1.9.2 HTML

Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de HyperText Markup Language, en español Lenguaje de Marcas Hipertextuales). HTML es el lenguaje que permite diseñar los hipertextos. Hoy día, la mayoría de los procesadores de textos disponen de opciones para guardar los documentos en este formato por lo que no presenta dificultad (30).

Ventajas:

- ✓ Texto presentado de forma estructurada y agradable.
- ✓ No necesita de grandes conocimientos cuando se cuenta con un editor de páginas web.
- ✓ Archivos pequeños.
- ✓ Despliegue rápido.
- ✓ Lenguaje de fácil aprendizaje.
- ✓ Lo admiten todos los exploradores.

1.9.3 CSS

CSS es el acrónimo de Cascading Style Sheets (es decir, hojas de estilo en cascada). Es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (31).

Algunas ventajas de utilizar CSS son:

- ✓ Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.

¹⁰ Interfaz de entrada común (en inglés *Common Gateway Interface*, abreviado *CGI*).



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ Separación del contenido de la presentación, lo que facilita al creador, diseñador, usuario o dispositivo electrónico que muestre la página, la modificación de la visualización del documento sin alterar el contenido del mismo, sólo modificando algunos parámetros del CSS.
- ✓ Optimización del ancho de banda de la conexión, pues pueden definirse los mismos estilos para muchos elementos con un sólo selector; o porque un mismo archivo CSS puede servir para una multitud de documentos.
- ✓ Mejora en la accesibilidad del documento, pues con el uso del CSS se evitan antiguas prácticas necesarias para el control del diseño (como las tablas) y que iban en perjuicio de ciertos usos de los documentos, por parte de navegadores orientados a personas con algunas limitaciones sensoriales (32).

1.9.4 Lenguajes de programación a utilizar

En la presente investigación se decide utilizar como lenguaje de programación PHP en su versión 5.3.8 ya que es el requerido por el gestor de contenido Joomla. Es completamente expandible y modificable, se puede ejecutar bajo Apache y permite la interacción con el gestor de base de datos MySQL. Además se utiliza CSS 3 para modificar los estilos del portal web del MINJUS de Cuba y HTML 5 para la visualización de las interfaces.

1.10 Servidor web Apache

Apache es un servidor web HTTP de código abierto, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Presenta entre otras características altamente configurables bases de datos de autenticación y negociado de contenido.

Ventajas:

- ✓ Modular.
- ✓ Código abierto.
- ✓ Multi-plataforma.
- ✓ Extensible.
- ✓ Popular (fácil conseguir ayuda/saporte).

Apache es el servidor web más prestigioso por su configurabilidad y robustez, siendo además casi universal al estar en una multitud de sistemas operativos. Al ser una tecnología gratuita de código fuente abierta le da una transparencia inmediata. Trabaja con gran cantidad de lenguajes



como Perl, PHP y Java, teniendo todo el soporte que se necesita para realizar páginas dinámicas. Por su diseño modular es muy sencillo ampliar sus capacidades y permite personalizar la respuesta ante los posibles errores que se den en el servidor. Por lo anteriormente expuesto se decide utilizar en su versión 2.2.21.

1.11 Entorno de desarrollo integrado

Un entorno de desarrollo integrado, llamado también IDE ¹¹, es un programa informático compuesto por un conjunto de herramientas de programación, que puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios.

Algunos de los principales y más populares son:

- BlueJ: desarrollado como un proyecto de investigación universitaria, es libre.
- Eclipse: desarrollado por la Fundación Eclipse, es libre y de código abierto.
- Jbuilder: desarrollado por Borland, es comercial pero también existe la versión gratuita.
- JCreator: desarrollado por Xinox, es comercial pero también existe la versión gratuita.
- JDeveloper: desarrollado por Oracle Corporation, es gratuito.
- NetBeans: gratuito y de código abierto.

Los más utilizados por ser de códigos abiertos son Eclipse y NetBeans, de forma general son bastante similares, ambos facilitan la programación proporcionando el auto-completamiento y la inserción automática de código, una descripción sencilla sobre la estructura del programa, opciones de refactorización y depuración, a pesar de que NetBeans realiza mejor el autocompletado (33).

1.11.1 Selección del entorno de desarrollo a utilizar

Después de realizar un estudio de las herramientas, para el desarrollo de la aplicación se selecciona el NetBeans en su versión 7.3 debido a que es libre y de código abierto. Este IDE posibilita el auto-completamiento de código, contiene una función que permite importar clases y permite la integración de servidores. Cuenta con herramientas para crear aplicaciones web profesionales y dinámicas, que le facilitan a los desarrolladores la implementación de las funcionalidades de la solución mediante un entorno agradable y con la incorporación de bibliotecas que engrandecen el desarrollo.

¹¹ Siglas en inglés de Integrated Development Environment



1.12 Conclusiones del capítulo

El análisis de los principales conceptos asociados al dominio del problema facilitó una mejor comprensión del contenido que se necesita en la presente investigación. El estudio del estado del arte realizado permitió obtener las tendencias de la forma de estructurar y organizar los portales web dedicados a la divulgación y promoción de la información jurídica, proporcionando ideas para el diseño del portal para el MINJUS de Cuba. Por otra parte, la selección de la metodología XP para guiar el proceso de desarrollo de software permitirá aumentar la productividad y los resultados y garantizará la obtención de un producto de mayor calidad que satisfaga plenamente las necesidades del cliente. Así mismo, la herramienta CASE Visual Paradigm 8.0 ayudará a automatizar el proceso de desarrollo del software y el CMS Joomla facilitará la construcción de un portal sencillo para el usuario. Además de que la gestión de los datos se realizará de manera segura con el uso del SGBD MySQL.



Capítulo 2: Propuesta de solución

2.1 Introducción

En este capítulo se expone una descripción detallada de las características del sistema y sus principales funcionalidades. Se realiza un levantamiento de los requisitos funcionales mediante la descripción contenida en las historias de usuario y de los requisitos no funcionales necesarios para el correcto funcionamiento del portal web del MINJUS de Cuba. Se hace énfasis además en las particularidades del diseño e implementación de dicho portal, se describe la arquitectura del sistema y los patrones de diseño a utilizar y se generan los artefactos y diagramas correspondientes de la metodología que se utiliza.

2.2 Descripción del sistema

Tomando como partida el problema identificado en la investigación se decide realizar un portal web para el MINJUS de la República de Cuba que brinde, de forma dinámica, información relacionada con el Ministerio y las Oficinas Registrales. Este portal posibilitará a los usuarios nacionales e internacionales la comunicación con las personas y entidades interesadas en conocer del ámbito jurídico en el país. Los actos registrales se realizarán de forma detallada mediante la generación dinámica de formularios para la captación de datos de solicitud de Trámites Registrales y Notariales, de forma tal que los usuarios puedan dar seguimiento a los estados por los que transitan sus trámites. Además se brindará un servicio eficiente de consulta legal en línea y serán publicadas una serie de encuestas con el objetivo de medir la calidad de los servicios de gestión de la información y de trámites que ofrece el Ministerio, para cubrir las principales necesidades de los usuarios.

El sistema contará con los siguientes niveles de usuarios:

- ✓ Webmaster: Cuenta con todos los privilegios, por lo que es el encargado del control total del sistema, principalmente de la parte técnica. Se encarga de controlar el acceso al portal, mediante la creación de usuarios y la asignación de los roles con sus respectivos permisos. También puede crear, editar, revisar y eliminar noticias en el portal, así como gestionar la base de datos.
- ✓ Administrador: Al igual que el webmaster cuenta con todos los privilegios, pero no se encarga de la parte técnica del sistema. Puede crear, modificar y eliminar contenido del



Portal web del Ministerio de Justicia de la República de Cuba.

portal, así como gestionar cualquier otra información. Es el encargado de responder las consultas y opiniones enviadas por otros usuarios en el sitio.

- ✓ Editor: Tiene permisos de adición, modificación y eliminación de noticias. Cuenta además con el privilegio de consultar cualquier otro artículo publicado por otro usuario.
- ✓ Reportero: Tiene los mismos permisos de gestionar noticias que el usuario reportero, lo que solamente podrá cambiar todo el contenido al que su nivel de privilegio le permita acceder.
- ✓ Usuario registrado: Sus privilegios son mínimos, podrá consultar el contenido presentado en el portal y realizar las encuestas publicadas. Tendrá acceso a los servicios de consulta legal en línea y captación de datos de solicitud de Trámites Registrales y Notariales, mediante la realización de formularios, dándole seguimiento a los trámites que ha realizado; además de poder enviar opiniones, sugerencias, quejas y reclamos a través del portal. Comprende a todos los funcionarios del MINJUS de Cuba.
- ✓ Usuario invitado: No tiene privilegios, navega a través de portal sin autenticarse, solamente puede consultar las informaciones, datos y noticias que se publican que sean de dominio público, sin comprometer la información confidencial que se maneje en el sitio. Comprende a todos los ciudadanos que puedan acceder al portal.

2.2.1 Modelo de dominio

Se utiliza para capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de un sistema. Cuando se realiza la programación orientada a objetos, el funcionamiento interno del software va a imitar en alguna medida a la realidad, por lo que el mapa de conceptos del modelo de dominio constituye una primera versión del sistema.



Portal web del Ministerio de Justicia de la República de Cuba.

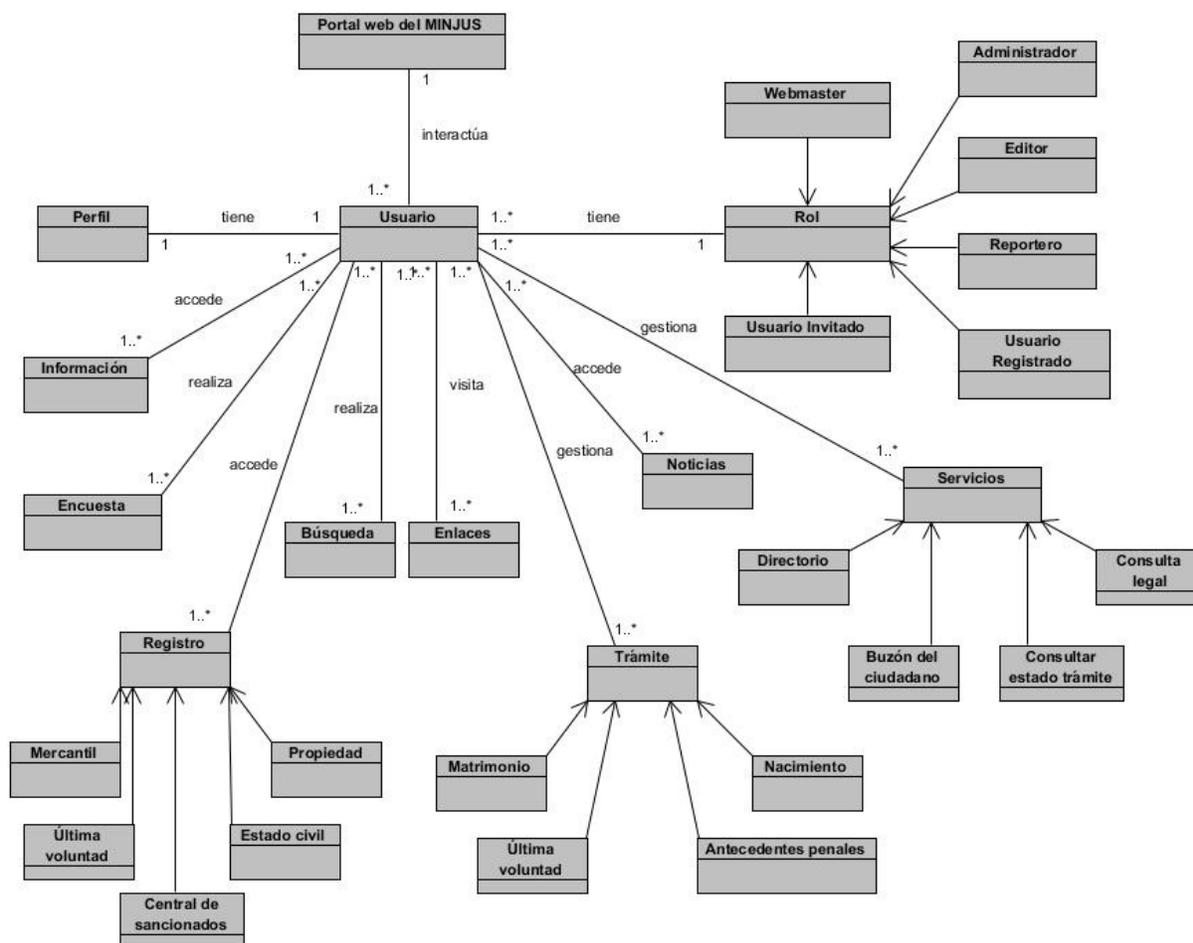


Figura 2: Modelo de dominio del sistema.

En el modelo del dominio se identificaron varios conceptos fundamentales para el desarrollo del sistema y su correcto funcionamiento. A continuación se muestran cada uno de ellos con su explicación y la función que realizan en el sistema, para ello fue necesaria la participación de los expertos del negocio.

Portal web del MINJUS: es el sistema encargado de gestionar toda la información que se mostrará en el portal web y las acciones de los usuarios sobre el mismo. Se encarga principalmente de la gestión de contenidos. Contiene usuarios, información, noticias, servicios, registros, trámites y otros.

Usuarios: son las personas que interactúan con el sistema, estos tienen asociado un rol, el cual les brinda los permisos sobre los servicios que ofrece el portal.

Perfil: muestra información sobre los usuarios registrados.



Portal web del Ministerio de Justicia de la República de Cuba.

Roles: identifican los niveles de acceso en el sistema, que les permitirán a los usuarios navegar en él, limitando las acciones de cada uno según el nivel que posea. Estos pueden ser: webmaster, administrador, reportero, editor, usuario registrado y usuario invitado.

Información: son artículos redactados por los publicadores del sistema con el objetivo de dar a conocer una información referente a un tema determinado.

Registros: estos representan en forma de artículos datos relacionados con los distintos registros con que trabaja el Ministerio: Registro de la Propiedad, Registro del Estado Civil, Registro Mercantil, Registro de Actos de Última Voluntad y Registro Central de Sancionados.

Encuestas: son un conjunto de preguntas que responden a indicadores de usabilidad previamente elaborados e identificados. Estas promueven el control social sobre la calidad de los servicios que se brindan.

Búsqueda: son realizadas por los usuarios, es una forma de optimizar la búsqueda de un contenido en particular.

Enlaces: brindan acceso a otros sitios de interés, estos se encuentran divididos en: Gobierno, Jurídicos, Prensa nacional, Políticos y de Interés.

Noticias: son artículos sobre el acontecer noticioso relacionados con la justicia, son tomados de fuentes previamente identificadas por los especialistas del centro.

Trámites: son formularios generados dinámicamente que realizan la captación de datos en línea, tales como: certificación de matrimonio, de antecedentes penales, de última voluntad, de divorcios y de nacimiento.

Servicios: son funcionalidades que brinda el sistema a los usuarios con el fin de responder a sus necesidades, como directorio del MINJUS, consulta legal, buzón del ciudadano y consultar estado de un trámite.

2.3 Requisitos de software

Para la obtención de los requisitos funcionales y no funcionales del software se emplea la técnica de entrevistas a los clientes.



2.3.1 Requisitos funcionales

Los requisitos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. Como se define en la ingeniería de requisitos, los requisitos funcionales establecen los comportamientos del sistema (10).

En la presente investigación se capturaron 43 requisitos funcionales, los cuales se exponen en la tabla 3. Para conocer a profundidad en qué consiste cada uno de ellos, consultar el documento anexo: Requisitos funcionales del MINJUS de Cuba.

RF 1 Autenticar usuario.	RF 23 Modificar encuesta.
RF 2 Añadir usuario.	RF 24 Eliminar encuesta.
RF 3 Modificar usuario.	RF 25 Buscar encuesta.
RF 4 Eliminar usuario.	RF 26 Visualizar encuesta.
RF 5 Añadir rol.	RF 27 Visualizar resultados.
RF 6 Modificar rol.	RF 28 Enviar Consulta.
RF 7 Eliminar rol.	RF 29 Modificar Consulta.
RF 8 Añadir noticia.	RF 30 Eliminar Consulta.
RF 9 Modificar noticia.	RF 31 Buscar Consulta.
RF 10 Eliminar noticia.	RF 32 Notificar Consulta.
RF 11 Buscar noticia.	RF 33 Enviar buzón
RF 12 Añadir información	RF 34 Modificar buzón
RF 13 Modificar información.	RF 35 Eliminar buzón
RF 14 Eliminar información.	RF 36 Buscar buzón.
RF 15 Buscar información	RF 37 Notificar buzón.
RF 16 Añadir datos del directorio	RF 38 Visualizar buzón.
RF 17 Modificar datos del directorio	RF 39 Añadir trámite.
RF 18 Eliminar datos del directorio	RF 40 Modificar trámite
RF 19 Visualizar datos del directorio	RF 41 Eliminar trámite
RF 20 Exportar información.	RF 42 Buscar trámite.
RF 21 Visualizar estadísticas.	RF 43 Visualizar estado del trámite.
RF 22 Añadir encuesta.	

Tabla 3: Requisitos funcionales del MINJUS.



2.3.1.1 Métrica utilizada para la validación de requisitos

Asegurar la validez de los requisitos previamente a comenzar un desarrollo de software, es una fase importante dentro del desarrollo del sistema, que tiene como objetivo comprobar que estos sean correctos. Para ello debe hacerse una comprobación de la correspondencia entre las descripciones iniciales y si el modelo es capaz de responder al planteamiento inicial. Esta fase debe realizarse o de lo contrario se corre el riesgo de implementar una mala especificación, con el costo que eso conlleva.

Técnica de estabilidad de requisitos:

La validación de requisitos se realiza para asegurar que todos los requisitos del software se han establecido de manera precisa. Se debe asegurar que se hayan detectado y corregido los errores e inconsistencias de estos y que cumplan con los estándares establecidos. Validar los requisitos es primordial ya que son la base y medida de la calidad del sistema. Esta técnica tiene como misión demostrar que los requisitos definen realmente el sistema que el usuario necesita o que el cliente desea.

La técnica de estabilidad de requisitos se calcula a través de la siguiente fórmula:

$$ETR = \left[\frac{RT - RM}{RT} \right] * 100$$

$$ETR = [(43 - 4) / 43] * 100 = 90.$$

RT: Total de requisitos definidos.

RM: Cantidad de requisitos modificados.

ETR: Valor de la estabilidad de los requisitos.

El valor **ETR** más cercano a 100 es el mejor porque mostrará que no se están realizando cambios sobre los requisitos, se demuestra que son estables y por tanto son confiables para realizar las siguientes fases del software. Los requisitos son estables cuando no existen adiciones o supresiones en ellos que impliquen modificaciones en las funcionalidades principales (34).

Al medir la estabilidad de los requisitos el valor fue próximo a 100, por lo que se llegó a la conclusión de que los requisitos definidos son estables, por lo tanto, es confiable trabajar el análisis y diseño sobre ellos.

Técnica para validar la especificación de los requisitos:



La técnica para validar la especificación de los requisitos o ausencia de ambigüedad, se mide a través de la siguiente fórmula:

$$ER = Nui / Nr$$

ER: especificidad de los requisitos.

Nr: total de requisitos definidos.

Nui: total de requisitos para los que los revisores tuvieron interpretaciones idénticas.

$$ER = 40 / 43 = 0.93.$$

Al sustituir los valores en la expresión se obtiene un resultado de 0.93, un valor bastante cercano a 1. Esto arroja un resultado satisfactorio, ya que será menor la ambigüedad, representando un 7% lo que indica que es baja y el 93% son entendibles. Los requisitos ambiguos fueron modificados para garantizar su correcta comprensión.

2.3.2 Requisitos no funcionales

Un requisito no funcional en la ingeniería de software es un requisito que especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos. Por tanto, se refieren a todos los requisitos que ni describen información a guardar, ni funciones a realizar (10).

Para el desarrollo del sistema se encontraron 17 requisitos no funcionales, los cuales se exponen a continuación:

Rendimiento:

RNF 1 Los tiempos de respuesta y velocidad de procesamiento de la información serán rápidos, no mayores de 25 segundos para las actualizaciones y para la aplicación web no debe exceder a los 2 minutos.

Disponibilidad:

RNF 2 La información debe estar disponible para los usuarios con acceso en todo momento.

Seguridad:

RNF 3 El sistema debe cumplir con los principios básicos de seguridad de cualquier sistema informático manteniendo la integridad, confidencialidad y disponibilidad de la información.



Portal web del Ministerio de Justicia de la República de Cuba.

RNF 4 El sistema debe implementar mecanismos orientados a la prevención, detección y recuperación en caso de fallas.

RNF 5 El sistema concederá acceso a partir de un nombre de usuario y una contraseña. Las contraseñas deberán tener más de siete caracteres de longitud y tener una fortaleza media.

RNF 6 El sistema implementará el uso de campos obligatorios y validaciones para garantizar la integridad de la información que se introduce por el usuario.

RNF 7 Se establecerán los servidores tanto de aplicación como de base de datos en lugares de restringido acceso al que solo podrá acceder el personal autorizado.

Hardware:

RNF 8 Los requisitos mínimos de hardware para el servidor:

- ✓ Memoria RAM: 1 GB.
- ✓ Micro procesador: Intel Pentium IV o superior.
- ✓ Capacidad Mínima de Disco Duro: 160 GB.

RNF 9 Para el Cliente:

- ✓ Procesador: Pentium IV o superior.
- ✓ Memoria RAM: 768 MB o superior.

Software:

RNF 10 Para el cliente:

- ✓ Navegador Mozilla Firefox 16.0 o superior.
- ✓ Sistema operativo Windows 2003 o superior, Linux.

RNF 11 Para el servidor:

- ✓ Sistema operativo Linux en cualquiera de sus distribuciones.
- ✓ Un servidor Apache 2.12 o superior con módulo PHP 5.0 o superior disponible
- ✓ Un servidor de base de datos MySQL 5.1.37 o superior.

Soporte:

RNF 12 Se debe incluir un manual con indicaciones referentes al uso del sistema y acerca de sus servicios.

RNF 13 En caso de requerirlo debe darse entrenamiento al personal que de cierta forma se relaciona con el sistema y sus servicios.

Interfaz:

RNF 14 La interfaz será atractiva para todos los usuarios, intuitiva, muy sugerente y amigable.



Portal web del Ministerio de Justicia de la República de Cuba.

RNF 15 Contará con diferentes funcionalidades agrupadas en varios tipos de contenidos, módulos y menús.

RNF 16 Debe poseer un diseño sencillo con colores agradables a la vista, que permita un balance adecuado entre funcionalidad y simplicidad.

Facilidad de uso:

RNF 17 El sistema podrá ser manejado por cualquier usuario con conocimientos básicos de informática.

2.3.3 Diagrama de interacción de los módulos

En el portal web se divide el contenido por módulos para hacer más asequible la manipulación del mismo, dentro de los cuales se destacan: Trámites, Encuestas, Servicios, Información, Registros, Noticia y Enlaces. Para especificar la interacción entre ellos se realiza el diagrama de interacción de módulos.

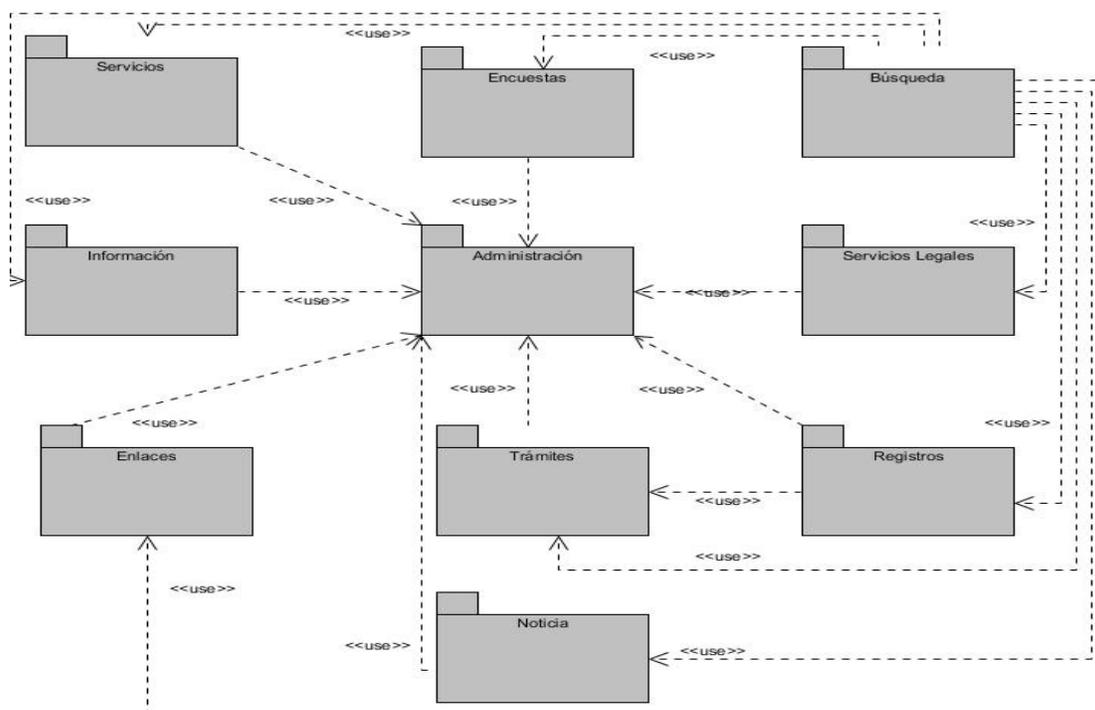


Figura 3: Diagrama de interacción de módulos.

2.4 Fase de planificación



2.4.1 Historias de usuario

Las historias de usuario (HU) son utilizadas en las metodologías de desarrollo ágiles para la especificación de requisitos. Son una forma rápida de administrar los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos. Permiten responder rápidamente a requisitos cambiantes (35).

Para el desarrollo del sistema se realizaron 16 historias de usuario. A continuación se muestran 3 de las historias de usuario del sistema:

Historia de Usuario	
Número: 4.	Nombre de la Historia de usuario: Gestionar noticia.
Cantidad de modificaciones a la Historia de usuario: Ninguna.	
Usuario: reportero, editor y webmaster.	Iteración asignada: 1.
Prioridad en negocio: Media.	Puntos estimados: 1 semana.
Riesgo en desarrollo: Media.	Puntos reales: 1 semana.
Referencias: RF 8, RF 9, RF 10	
Descripción: El sistema debe permitir al usuario añadir, modificar y eliminar noticias. Estas dos últimas de acuerdo al usuario que la creó.	

Tabla 4: HU#4 Gestionar noticia.

Historia de Usuario	
Número: 6.	Nombre de la Historia de usuario: Gestionar datos del directorio.
Cantidad de modificaciones a la Historia de usuario: Ninguna.	
Usuario: webmaster.	Iteración asignada: 2.
Prioridad en negocio: Alta.	Puntos estimados: 2 semanas.
Riesgo en desarrollo: Medio.	Puntos reales: 2 semanas.
Referencias: RF 16, RF 17, RF 18, RF 19	
Descripción: El sistema debe permitir al usuario crear, modificar, eliminar y visualizar elementos de directorio (páginas que se muestran en el servicio "Directorio del MINJUS").	

Tabla 5: HU#6 Gestionar datos del directorio.

Historia de Usuario	
Número: 15.	Nombre de la Historia de usuario: Gestionar trámites.
Cantidad de modificaciones a la Historia de usuario: Ninguna.	
Usuario: administrador.	Iteración asignada: 1.



Prioridad en negocio: Alta.	Puntos estimados: 2 semanas.
Riesgo en desarrollo: Alta.	Puntos reales: 2 semanas.
Referencias: RF 41, RF 42, RF 43	
Descripción: El sistema debe permitir al usuario añadir, modificar y eliminar trámites.	

Tabla 6: HU#15 Gestionar trámites.

2.4.2 Plan de entregas

El plan de entregas es un documento que especifica exactamente qué historias de usuario serán implementadas en cada entrega del sistema y sus prioridades, de modo que también permita conocer con exactitud qué historias de usuario serán implementadas en la próxima liberación (36). A continuación se presenta el plan de entregas.

Iteraciones	Historias de Usuario	Prioridad	Fecha de Inicio	Fecha Fin
Iteración 1	HU 1	Alta	10/03/2014	17/03/2014
Iteración 1	HU 2	Alta	03/03/2014	10/03/2014
Iteración 1	HU 3	Alta	17/03/2014	24/03/2014
Iteración 2	HU 4	Media	24/03/2014	31/03/2014
Iteración 2	HU 5	Media	27/03/2014	03/04/2014
Iteración 3	HU 6	Alta	14/04/2014	28/03/2014
Iteración 3	HU 7	Media	17/04/2014	24/04/2014
Iteración 2	HU 8	Media	29/03/2014	05/04/2014
Iteración 2	HU 9	Alta	03/04/2014	10/04/2014
Iteración 2	HU 10	Media	05/04/2014	12/04/2014
Iteración 3	HU 11	Alta	18/04/2014	02/05/2014
Iteración 3	HU 12	Alta	21/04/2014	28/04/2014
Iteración 3	HU 13	Alta	23/04/2014	30/04/2014
Iteración 3	HU 14	Alta	28/04/2014	05/05/2014
Iteración 3	HU 15	Alta	21/04/2014	05/05/2014
Iteración 2	HU 16	Media	07/04/2014	14/04/2014

Tabla 7: Plan de entregas.

2.4.3 Plan de iteraciones

El plan de iteración consiste en seleccionar las historias de usuario que, según el plan de entregas, corresponderían a cada iteración. Se determinaron tres iteraciones, teniendo en cuenta las dependencias entre las historias de usuario y las prioridades definidas por el cliente. En la



Portal web del Ministerio de Justicia de la República de Cuba.

primera iteración se realizan las HU pertenecientes a la administración de los usuarios con los roles y permisos asignados; en la segunda, en dependencia de los niveles de acceso, se garantiza la gestión de los contenidos del portal y en la tercera, las HU correspondientes a los trámites que dependían de la configuración y gestión de los contenidos publicados y de los roles definidos.

Iteración	Descripción de la Iteración	Orden de la HU a implementar	Duración total (en semanas)
1	Implementación del módulo de Administración.	HU#1 Autenticar usuario.	3
		HU#2 Gestionar usuario.	
		HU#3 Gestionar rol.	
2	Implementación de los módulos: Noticia, Información, Encuestas y Búsqueda.	HU#4 Gestionar noticia.	3
		HU#5 Gestionar información.	
		HU#8 Visualizar estadísticas.	
		HU#9 Gestionar encuestas.	
		HU#10 Visualizar encuestas.	
		HU#16 Realizar búsqueda.	
3	Implementación de los módulos: Servicios, Servicios Legales, Trámites y Registros.	HU#6 Gestionar datos del directorio.	3
		HU#7 Exportar contenido.	
		HU#11 Gestionar consultas.	
		HU#12 Notificar consulta.	
		HU#13 Gestionar buzón.	
		HU#14 Notificar buzón.	
		HU#15 Gestionar trámites.	

Tabla 8: Plan de iteraciones.

2.5 Fase de diseño

2.5.1 Arquitectura de la información

La arquitectura de la información para portales web profesionales tiene como objetivo principal facilitar los procesos de comprensión y asimilación de la información, así como las tareas que ejecutan los usuarios en un espacio de información definido (37). Las ventajas más óptimas de la utilización de la arquitectura de la información son:



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ La optimización de tiempos de ejecución en la fase de diseño web.
- ✓ Evita que los usuarios abandonen la página por falta de entendimiento.
- ✓ Permite organizar y estructurar los contenidos de un sitio web pensando en el usuario y crea un orden necesario para su comprensión.

Para estructurar el portal web del MINJUS de Cuba se realizó la unión de las tres estructuras existentes: la estructura lineal, donde una página lleva a otra mediante un enlace, la estructura jerárquica, que contiene navegación en cascada y la estructura en red, donde se puede acceder a cualquier sección o menú desde cualquier parte del sitio. En la estructura de red las páginas se relacionan entre sí mediante enlaces internos (38).

La información para el portal web queda estructurada y organizada de la siguiente forma:

1. El Ministerio: en esta sección se encuentran artículos relacionados con los objetivos y la estructura del ministerio.
 - ✓ Misión
 - ✓ Estructura
 - ✓ Funciones
 - ✓ Órgano de Relación.
2. Información: la información del ministerio se expone a través de artículos donde aparecerán noticias externas e internas relacionadas con temas jurídicos. Además se describen los actos registrales que se realizarán en el portal.
 - ✓ Noticias
 - ✓ Artículos
 - ✓ Registros
3. Servicios: los servicios que se exponen están clasificados en servicios legales y servicios. Estos están disponibles para los ciudadanos y entidades interesadas.
 - ✓ Directorio
 - ✓ Consulta Legal
 - ✓ Buzón del Ciudadano
 - ✓ Consultar estado de un trámite
 - ✓ Servicios Legales
 - ✓ Gestionar Encuestas
 - ✓ Búsqueda



Portal web del Ministerio de Justicia de la República de Cuba.

- ✓ Mapa del Sitio
 - ✓ Autenticación de Usuario
4. Enlaces: aparecen hipervínculos desde el portal hacia otros sitios web. Estos están divididos en los módulos que aparecen a continuación:
- ✓ Globales
 - ✓ De Interés
 - ✓ Prensa nacional
 - ✓ Políticos
5. Descargas: en esta sección se podrán realizar descargas de documentos oficiales del Ministerio.
- ✓ Libros
 - ✓ Revistas
 - ✓ Normativa
6. Trámites en línea: los trámites se realizarán a través de formularios con la captación dinámica de datos. Estos recogen solicitudes, donde luego los interesados podrán seguir los estados por los que estas transitan.

2.5.2 Arquitectura del sistema

Según Kruchten Philippe¹² la arquitectura de software tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada, para satisfacer la mayor funcionalidad y requisitos de desempeño de un sistema, así como requisitos no funcionales.

La arquitectura en tres niveles es la especialización de la arquitectura cliente-servidor donde la carga se divide en tres capas con un reparto claro de funciones: una capa para la presentación (interfaz de usuario), otra para el cálculo (donde se encuentra modelado el negocio) y otra para el almacenamiento (persistencia). Una capa solamente tiene relación con la siguiente y entre ellas. El desarrollo se puede llevar a cabo en varios niveles y, en caso de cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado (39).

¹² Ingeniero de software canadiense y profesor de Ingeniería de Software de la Universidad de British Columbia en Vancouver, Canadá.

En Joomla la arquitectura en tres niveles está formada por las capas: extensiones, aplicación y marco de trabajo, las cuales interactúan entre sí de acuerdo a una serie de patrones establecidos. En la siguiente figura se representan los diferentes componentes del CMS pertenecientes a cada capa de la arquitectura.



Figura 4: Arquitectura en tres niveles del CMS Joomla.

Capa de Extensión: está formada precisamente por casi todo lo que se conoce como extensiones de Joomla: módulos, componentes y plantillas del sistema. Se trata de programas con una función específica, desarrollados total o parcialmente con el marco de trabajo de Joomla para ejecutarse sobre alguna de las aplicaciones de la capa inferior.

Capa de Aplicación: esta capa se comunica con la capa de extensión para recibir las solicitudes y presentar los resultados y con la capa de marco de trabajo para solicitar al gestor de base de datos almacenar o recuperar datos de él. En esta aparecen tres aplicaciones básicas que extienden de la clase JApplication, estas proporcionan una serie de funcionalidades para la gestión de contenidos y una estructura sobre la que se ejecutan extensiones.

1. El instalador de Joomla o JInstalator: su función es instalar todo el sistema Joomla sobre un servidor web.
2. El JSite: su función es la administración del sitio. Es la aplicación encargada de crear y gestionar la parte que se presentará de forma abierta a todos los usuarios.
3. El administrador o JAdministrator: es la aplicación encargada de gestionar todos los parámetros de configuración de la parte pública. Además, permite la ejecución de extensiones, que generalmente desempeñarán funciones específicas de la administración de sitios web: gestión de ficheros, copia de seguridad y control de URLs.



Portal web del Ministerio de Justicia de la República de Cuba.

Capa de Marco de trabajo: en esta capa residen los datos y es la encargada de acceder a los mismos. Se encarga del encapsulamiento de la lógica de acceso a datos. Está formada por bibliotecas externas, que proporcionan funcionalidades específicas que ya han sido desarrolladas por terceros, el marco de trabajo propiamente dicho y una serie de complementos que permiten extender sus funcionalidades.

Aunque el CMS Joomla tiene sus cimientos en la arquitectura de tres capas, ya que ayuda a estructurar la aplicación en grupos de subtareas, en el que cada uno pertenece a una capa en particular; este también utiliza como patrón arquitectónico el Modelo Vista Controlador en su desarrollo, el cual tiene sus bases en la arquitectura 3 capas, permitiendo mayor seguridad y protección de los datos.

2.5.2 Patrón arquitectónico

El patrón arquitectónico Modelo Vista Controlador (MVC) es un patrón que separa los datos y la lógica de negocio de una aplicación, de la interfaz de usuario y del módulo encargado de gestionar los eventos y las comunicaciones. Para ello el MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador. En Joomla este se utiliza específicamente para desarrollar la estructura de los Componentes (/components) del CMS.

- ✓ El modelo (JModel): es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación o lógica de negocio. Notifica a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada a un usuario. Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador'.
- ✓ El controlador (JController): responde a eventos o acciones del usuario e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información, como el registro en una base de datos. También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta de 'modelo', ejemplo: desplazamiento de un documento o algunos registros de una base de datos; por tanto se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo'.
- ✓ La vista (JView): genera la interfaz con la que la aplicación interacciona con el usuario. Presenta el 'modelo' o la información y la lógica de negocio en un formato adecuado para



Portal web del Ministerio de Justicia de la República de Cuba.

interactuar con la interfaz de usuario, por tanto requiere del 'modelo' la información que debe representar como salida. Esta envía las acciones del usuario al 'controlador'.

La estructura de un componente queda de la siguiente forma: un componente es dividido en dos partes, una parte para mostrar el contenido que es ejecutado en el sitio, el cual se guarda en el directorio /components y la otra parte es para la administración del contenido, del lado de la administración, la cual se guarda en el directorio /administrator/components. Cada una de estas partes es una implementación del patrón MVC, por lo que básicamente, un componente contiene dos subsistemas, uno para la administración y el otro para el sitio (25).

Entre los componentes que se utilizaron para dar solución a la propuesta se encuentra el componente AcePolls para la realizar las encuestas y el Fabrik para la realización de los distintos formularios de los trámites y consultas.

2.5.3 Patrones de diseño

Los patrones de diseño expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de software. Joomla los implementa con el objetivo de definir los componentes y subsistemas que lo integran.

Patrones GoF¹³:

Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de software o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación, que resuelve un problema general de diseño en un contexto particular. Son menores en escala que los patrones arquitectónicos y tienden a ser independientes de los lenguajes y paradigmas de programación. Su aplicación no tiene efectos en la estructura fundamental del sistema, pero sí sobre la de un subsistema, debido a que especifica a un mayor nivel de detalle, sin llegar a la implementación, el comportamiento de los componentes del subsistema (40).

El patrón de la Fábrica (Factory): se utiliza para construir y devolver objetos. Este patrón se utiliza en los casos en donde diferentes clases, que por lo general derivan de una clase abstracta, son instanciadas dependiendo de los parámetros. Joomla proporciona la clase estática JFactory, que

¹³Gang of Four, en español Banda de los Cuatro, formada por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides.

implementa el patrón de la fábrica. Esta clase es importante porque permite acceder de forma sencilla a instancias de los objetos globales (41).

Este ejemplo muestra cómo acceder a algunos de los objetos globales utilizando JFactory en la administración del módulo mod_menu de la aplicación.

```
20
21 // Initialise variables.
22 $lang      = JFactory::getLanguage();
23 $user      = JFactory::getUser();
24 $menu      = new JAdminCSSMenu();
25 $enabled   = JRequest::getInt('hidemainmenu') ? false : true;
26
```

Figura 5: Línea de código del patrón de la Fábrica.

El patrón Instancia única (Singleton Pattern): se utiliza para permitir la creación de un único objeto de una clase específica. Esto se consigue haciendo el constructor privado y usando un método estático para crear instancias de la clase. Muchas de las clases de Joomla utilizan un modelo pseudo-singleton que permite crear instancias de objetos y de acceso. Para lograr esto, a menudo utiliza un método estático llamado getInstance(), en algunos casos JFactory actúa como una interfaz a través de este método. Se puede pensar en ellos como una jerarquía en la forma de crear instancias de objetos y se deben utilizar en orden de prioridad: método JFactory, método getInstance() y el constructor normal (new) (25).

```
9
10 function __construct()
11 {
12     parent::__construct();
13     $this->_sessionId = (int) JRequest::getVar('sessionId') ;
14     $this->sessionWhoTable =& JTable::getInstance ('sessionwho', 'Table') ;
15 }
16
```

Figura 6: Línea de código del patrón Instancia única.

Patrones GRASP¹⁴:

Representan los principios básicos de la asignación de responsabilidades a objetos, expresados en forma de patrones (42). Los patrones GRASP permiten obtener un diseño de clases con mayor

¹⁴ General Responsibility Assignment Software Patterns (Patrones Generales de Software para Asignar Responsabilidades)



Portal web del Ministerio de Justicia de la República de Cuba.

cohesión, contribuyen al bajo acoplamiento, facilitando el mantenimiento y ofreciendo oportunidades para la reutilización.

Experto: se encarga de asignar una responsabilidad al experto en información, o sea, aquella clase que cuenta con la información necesaria para cumplir la responsabilidad. Con la utilización de este patrón la información se mantendrá encapsulada. Este se encuentra principalmente en las agrupaciones de clases *ModelForm* que extienden del *JModel* de cada uno de los componentes que contienen la información necesaria de la tabla que representan.

Creador: este patrón es el responsable de asignarle a la clase B la responsabilidad de crear una instancia de clase A, B es un creador de los objetos A. El objetivo de este patrón es encontrar al creador que se debe relacionar con el objeto que se produce. Este patrón se evidencia en la clase controladora *ControllerForm()* que extiende de *JController*.

Alta Cohesión: asigna una responsabilidad de forma tal que la cohesión siga siendo alta. La cohesión es la medida que determina cuán relacionadas y adecuadas están las responsabilidades de una clase. La alta cohesión consiste en garantizar que una clase colabore con otras para compartir el trabajo si la tarea es grande. Hace de los componentes del sistema un conjunto bien acoplado.

Bajo Acoplamiento: es el encargado de asignar una responsabilidad para conservar bajo acoplamiento. El acoplamiento es una medida de la fuerza con que una clase se relaciona con otras. Su utilización facilita el diseño de clases más independientes, lo cual posibilita que se adapten mejor a los cambios. Es importante mencionar que el grado de acoplamiento no puede considerarse aisladamente de otros principios como Experto y Alta Cohesión. Este patrón está presente en las clases *JModel()*, *JView()* y *JController* de cada componente de la aplicación.

Controlador: posibilita la asignación de responsabilidades para controlar el flujo de eventos del sistema a clases específicas. Sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario enviándolo a las distintas clases según el método llamado. Este patrón se evidencia en las clases que forman parte de *JController*.

2.5.4 Modelo de datos del módulo Trámites

El modelo de datos es un conjunto de conceptos, reglas y convenciones para describir distintos aspectos del negocio del sistema. Su función es la descripción estructurada de la organización

del negocio del cliente para la construcción del software (43). A continuación se muestra el diagrama ORM (Mapeo de Objetos Relacionales), donde se modelan las tablas de la base de datos pertenecientes al módulo Trámite, las relaciones existentes entre ellas y los atributos y llaves primarias de las entidades.

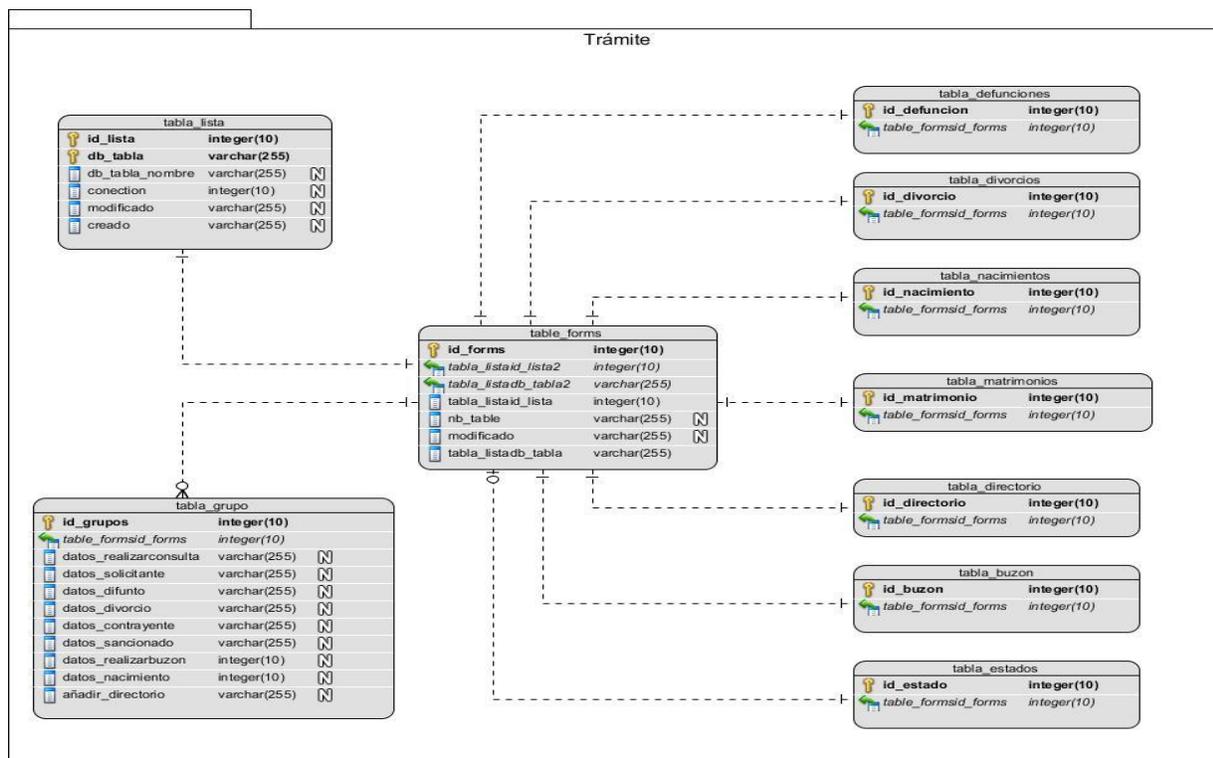


Figura 7: Modelo de datos del módulo Trámites.

2.5.5 Tarjetas CRC¹⁵

Describen las clases utilizadas en la programación de una historia. Estas tarjetas se dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores. Para cada clase identificada se rellenará una tarjeta de este tipo y se especificará su finalidad así como otras clases con las que interactúe. Su objetivo es facilitar la comunicación y documentar los resultados.

A continuación se representa la tarjeta CRC de la clase Trámite:



¹⁵Clase - Responsabilidad – Colaborador.



Responsabilidades	Colaboraciones
Añadir trámite.	Usuario
Modificar trámite	Registros
Eliminar trámite	Formulario
Buscar trámite	
Visualizar estado del trámite.	

Tabla 9: Tarjeta CRC de la clase Trámite.

2.6 Implementación

2.6.1 Estándares de codificación

Los estándares de codificación definen la propiedad del código compartido así como las reglas para escribir y documentar la comunicación entre diferentes piezas de código, desarrolladas por diferentes equipos. Los programadores las han de seguir de tal manera que el sistema se vea como si hubiera estado escrito por una sola persona.

Joomla utiliza las normas de codificación PEAR (Repositorio de Aplicaciones y Extensiones de PHP), para establecer un único formato estandarizado, que haga el código más fácil de leer y permita que otros desarrolladores puedan modificar el código fácilmente. Las normas más comunes que establece PEAR se encuentran en el [Anexo 1 Normas de Codificación](#).

Con el fin de mejorar la coherencia y la legibilidad del código fuente, en Joomla también se ejecuta una herramienta para el análisis del estilo de codificación de cambios que se insertan en el repositorio. Esto es para controlar que las nuevas contribuciones o extensiones que se van a incorporar, cumplan con los estándares de codificación y que el estilo de codificación en el código fuente sea consistente (25).

Esta herramienta, llamada PHP_CodeSniffer, es un script que utiliza PHP5, JavaScript y CSS para detectar violaciones de un estándar de codificación definido. Garantiza que el código permanezca limpio y consistente. También puede ayudar a prevenir algunos errores semánticos comunes cometidos por los desarrolladores (44).

2.6.2 Tareas de ingeniería por historia de usuario

Las tareas de la ingeniería son actividades sencillas que se derivan de las historias de usuario para simplificar la implementación de las mismas, estas se plasman en tarjetas de papel donde

se describe lo que se debe realizar. Son muy dinámicas y flexibles ya que pueden ser cambiadas por otras más generales o más específicas, agregarse nuevas o modificarse según las necesidades existentes. El conjunto de tareas de la programación a elegir debe proporcionar suficiente disciplina para alcanzar una alta calidad para el software. Cada una de estas tareas podrá ser comprobada después a través de los casos de prueba (34).

Tarea de ingeniería	
Número tarea: 4	Número Historia de usuario: 20
Nombre tarea: Validar formularios correspondientes a los trámites.	
Tipo tarea: Corrección.	Puntos estimados: 1 semana.
Programador responsable: Arletis Francis Alitimón.	
Descripción: Se realizará la validación de la captación de datos de los formularios de acuerdo al trámite en ejecución.	

Tabla 10: Tarea de ingeniería#4 Validar formularios correspondientes a los trámites.

2.6.3 Diagrama de despliegue

El diagrama de despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado, que muestra las relaciones físicas de las distintas partes que componen un sistema. Describen la topología del sistema, la estructura de los elementos de hardware y sus relaciones.

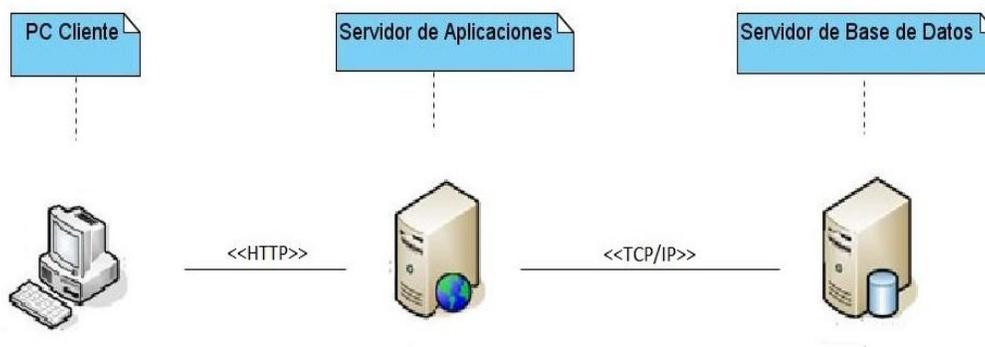


Figura 8: Diagrama de despliegue.

2.7 Conclusiones del capítulo

En el presente capítulo se elaboraron los elementos descriptivos y artefactos fundamentales correspondientes a las fases de planificación, diseño e implementación que propone la metodología de desarrollo utilizada XP. Se inicia con la conceptualización de los principales objetos del sistema para la realización del diagrama de modelo de dominio y sus principales



Portal web del Ministerio de Justicia de la República de Cuba.

funcionalidades, para lograr una mejor comprensión de la propuesta. Se logró realizar un levantamiento de los requisitos funcionales mediante la descripción contenida en las historias de usuario. Quedó definida la arquitectura del sistema y durante el proceso de implementación fueron identificados algunos patrones GoF y GRASP que posibilitan la optimización durante esta fase. Se detallaron los planes de entrega e iteración y fueron generadas las tarjetas CRC para establecer el comportamiento y funcionamiento de las clases principales del sistema. Se describe además el diagrama de despliegue, teniendo en cuenta la definición de los requisitos no funcionales identificados.

Capítulo 3: Validación de la solución

3.1 Introducción

En el presente capítulo se muestran los resultados de la aplicación de las métricas para validar el diseño del portal, utilizando el tamaño operacional de las clases y las relaciones existentes entre ellas. Se realiza además la validación del sistema, estableciendo una estrategia de prueba que se ejecuta a través de técnicas y métodos, presentándose por último los resultados obtenidos.

3.2 Validación del diseño

Las métricas del diseño orientadas a clases permiten evaluar cuantitativamente la calidad de los atributos del software durante el desarrollo, midiendo la complejidad, funcionalidad y eficiencia de este proceso (45). A continuación se comprueba la calidad del diseño del portal web del MINJUS de Cuba, empleando las métricas tamaño operacional de las clases y relaciones entre clases.

3.2.1 Tamaño operacional de las clases (TOC)

La métrica TOC permite medir la responsabilidad (R), la complejidad de implementación (CI) y la reutilización de las clases del diseño (Reu). Es importante para esta métrica que la responsabilidad y la complejidad sean inversamente proporcionales a la reutilización, por lo que a mayor responsabilidad y complejidad de implementación de una clase, menor será su nivel de reutilización. Para la realización de la misma se determinan la cantidad de atributos de la clase (CA) y de operaciones (CO), cuya suma devuelve la cantidad de procedimientos (CP). En el [Anexo 2](#) se muestran los indicadores a tener en cuenta para la clasificación en categorías de los distintos valores que toman las clases, los atributos y el modo en que se afectan (34).

Se determinaron un total de 11 clases pertenecientes al módulo Trámites, de ellas se describen la cantidad de operaciones y de procedimientos existentes, quedando definidos los valores asociados a los atributos de calidad antes mencionados después de haber sido sometidos a la métrica.

No	Clases	CA	CO	CP	R	CI	Reu
1	Usuario	5	3	8	Media	Media	Media
2	Encuesta	5	5	10	Media	Media	Media



Portal web del Ministerio de Justicia de la República de Cuba.

3	Trámite	5	4	9	Media	Media	Media
4	Servicio	5	3	8	Media	Media	Media
5	Enlaces	4	3	7	Baja	Baja	Alta
6	Noticia	4	3	7	Baja	Baja	Alta
7	Registro	4	3	7	Baja	Baja	Alta
8	Información	4	3	7	Baja	Baja	Alta
9	Rol	2	4	6	Baja	Baja	Alta
10	Búsqueda	3	1	4	Baja	Baja	Alta
11	Perfil	5	1	6	Baja	Baja	Alta

Tabla 11: Comportamiento de los atributos de calidad por clases de la métrica TOC.

El promedio de la cantidad de procedimientos de las clases es 7.18, obteniendo como resultado al aplicar la métrica y el umbral definido que existen 7 clases de responsabilidad baja, que representan un 64% del total, por lo que la complejidad de implementación es baja, favoreciendo la reutilización de las mismas.

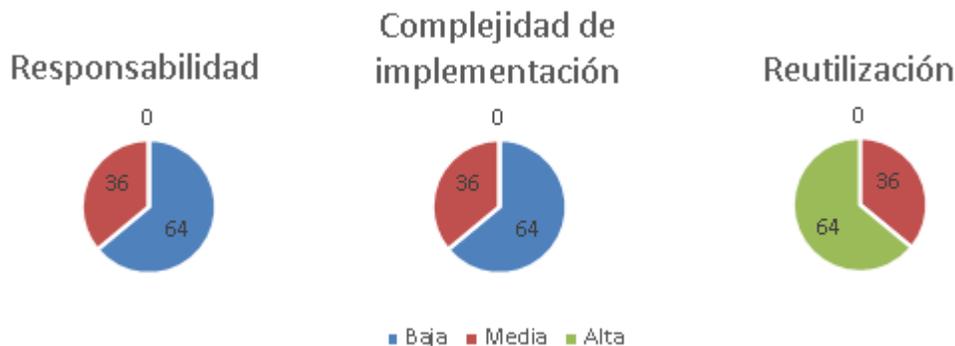


Figura 9: Comportamiento de los atributos de calidad de la métrica TOC.

3.2.2 Relaciones entre clases (RC)

La métrica RC permite evaluar el acoplamiento (A), la complejidad de mantenimiento (CM) y la cantidad de pruebas de unidad (CP) para probar una clase, teniendo en cuenta las relaciones existentes entre ellas. Se utiliza también para medir el atributo referente a la cantidad de relaciones de uso y reutilización de clases. El proceso se inicia calculando el promedio de asociaciones de uso que es 3,81 para determinar las categorías en la que se encuentran los atributos de calidad. En el [Anexo 3](#) se observan las tablas con los criterios para evaluar las categorías de los atributos y el modo en que se afectan.

No	Clases	CR	A	CM	CP	Reu
1	Usuario	10	Alto	Alta	Alta	Baja
2	Encuesta	1	Bajo	Baja	Baja	Alta
3	Trámite	6	Alto	Media	Media	Media
4	Servicio	4	Alto	Baja	Baja	Alta
5	Enlaces	1	Bajo	Baja	Baja	Alta
6	Noticia	1	Bajo	Baja	Baja	Alta
7	Registro	6	Alto	Media	Media	Media
8	Información	1	Bajo	Baja	Baja	Alta
9	Rol	1	Bajo	Baja	Baja	Alta
10	Búsqueda	10	Alto	Alta	Alta	Baja
11	Perfil	1	Bajo	Baja	Baja	Alta

Tabla 12: Comportamiento de los atributos de calidad por clases de la métrica RC.

Los resultados obtenidos después de aplicar la métrica RC demuestran que un 55% de las clases presentan bajo acoplamiento. Se encuentran pocas relaciones de uso y dependencias entre las mismas, por lo que no es muy alto el grado de complejidad de mantenimiento y existe un bajo grado de esfuerzo para realizar corrección y cambios en el portal. Este resultado favorece la reutilización de las clases.



Figura 10: Comportamiento de los atributos de calidad de la métrica RC.

3.3 Definición de la estrategia de prueba

La estrategia de prueba del software debe ser suficientemente flexible para promover la creatividad y adaptabilidad necesarias para adecuarlas a todos los sistemas. Para determinar el nivel de calidad se deben efectuar unas medidas que permitan comprobar el grado de cumplimiento, respecto a las especificaciones iniciales de la aplicación.



Portal web del Ministerio de Justicia de la República de Cuba.

En el proceso de prueba al portal web del MINJUS de Cuba se aplicarán tres iteraciones, en caso de que persistan las no conformidades (NC) dicho proceso no será abandonado hasta que el equipo de desarrollo logre solucionarlas. Se comenzará revisando las NC detectadas con antelación y las pendientes, siendo resueltas todas con éxito. Se realizarán pruebas funcionales para comprobar el cumplimiento de los requisitos iniciales del sistema planteados por el cliente y pruebas unitarias, dentro de las cuales se encuentra la prueba de caja negra y la prueba de caja blanca. La primera se realizará por escenarios correspondientes a los casos de pruebas, con el objetivo de validar el correcto funcionamiento de las funcionalidades y la segunda se efectuará al código para determinar que las funciones internas del portal se ejecutan de forma correcta; esta prueba se realizará a través del camino básico y la complejidad ciclomática existente.

Para las pruebas de carga y estrés se realizarán iteraciones redoblando en cada una el número de usuarios tratando de hacer colapsar el sistema. Las pruebas de seguridad determinan los puntos más vulnerables de la aplicación, esta se realizan con el fin de que el sistema sea lo más seguro posible y conserve los aspectos de seguridad. Para cada una de las pruebas a realizar se presentan los resultados obtenidos de forma gráfica para su mejor comprensión.

3.4 Prueba de caja negra

Este método de prueba se realiza dando distintos valores a las entradas; los datos de las mismas se escogen atendiendo a las especificaciones del problema sin importar los detalles internos, con el fin de verificar que el programa se ejecute correctamente. Esta prueba se centra en los requisitos funcionales del software y permite obtener entradas que prueben todos los flujos de una funcionalidad. Con la misma se intenta encontrar:

- ✓ Funcionalidades incorrectas o ausentes.
- ✓ Errores de interfaz.
- ✓ Errores en estructuras de datos o en accesos a las bases de datos externas.
- ✓ Errores de inicialización y finalización.

3.4.1 Técnica de partición de equivalencia

La partición de equivalencia permite examinar los valores válidos e inválidos de las entradas existentes en el software. Esta técnica se dirige a la definición de casos de pruebas que descubran clases de errores en las funcionalidades. Para comprobar la calidad del producto desarrollado, si este presenta defectos graves o cumple con los requisitos necesarios, se realiza



un caso de prueba por cada historia de usuario, de manera que se pueda abarcar todos los requisitos que el sistema debe cumplir (34).

El procedimiento para realizar la técnica de partición equivalente es el siguiente:

1. Se identifican clases de equivalencia válida (CEV) y no válidas (CENV).
2. Asignar un número único a cada clase de equivalencia.
3. Escribir casos de prueba hasta que sean cubiertas todas las CEV, intentando cubrir en cada caso tantas CEV como sea posible.
4. Para cada CENV, escribir un caso de prueba, cubriendo en cada caso una CENV (46).

3.4.2 Diseño de caso de prueba

Un caso de prueba es un conjunto de entradas, condiciones de ejecución y resultados esperados; desarrollado para conseguir un objetivo particular o condición de prueba y para verificar el cumplimiento de un requisito específico. Para llevarlo a cabo, es necesario definir las precondiciones y postcondiciones, identificar unos valores de entrada y conocer el comportamiento que debería tener el sistema ante dichos valores. Tras realizar ese análisis e introducir dichos datos en el sistema, se observará si su comportamiento es el previsto o no y de esta forma se determinará si ha pasado o no la prueba (47).

El objetivo es diseñar casos de prueba que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y de esfuerzo. Los casos de prueba de caja negra pretenden demostrar que:

- ✓ Las funciones del software son operativas.
- ✓ La entrada se acepta de forma correcta.
- ✓ Se produce una salida correcta.
- ✓ La integridad de la información externa se mantiene.

Caso de Prueba	
Número Caso de prueba: 16	Número Historia de usuario: 15
Nombre Caso de prueba: Gestionar trámites.	
Descripción: El sistema debe permitir al usuario añadir, modificar y eliminar trámites. El usuario asignado podrá modificar un trámite. El usuario asignado podrá eliminar un trámite.	
Condiciones de ejecución:	



El sistema debe brindar un espacio al administrador para poder crear, modificar, eliminar y visualizar un trámite.

- ✓ Para crear un nuevo trámite el usuario llena los datos correspondientes y otras informaciones de los formularios para enviar solicitudes.
- ✓ Para editar, se selecciona la lista correspondiente al formulario que desea modificar, oprime en el botón de editar y arregla el campo que desee. Oprima el botón de guardar una vez terminada todas las modificaciones.
- ✓ Para eliminar se selecciona el elemento que desea eliminar, oprime en el botón de eliminar y se eliminará automáticamente todos los atributos referentes al mismo, además de las tablas correspondientes a este trámite en la base de datos.

Entradas:

Para crear un trámite se deben llenar o entrar diferentes informaciones de los usuarios que recibirán la solicitud y del implicado.

Para editar o modificar un trámite se debe seleccionar el trámite deseado.

Para eliminar trámite se debe seleccionar el trámite deseado.

Posibles Resultados:

Se obtendrá la solicitud de un trámite en caso de que se llenen correctamente todos los campos necesarios. En caso contrario se mostrarán los errores.

Se modificará o editará un trámite en caso de que se seleccione correctamente al que se le desea realizar la acción y los cambios realizados a los campos sean correctos. En caso contrario no se mostrará ningún elemento.

Se eliminará el trámite una vez sea seleccionando el elemento deseado. En caso contrario no será borrado el evento.

Evaluación: Satisfactorio.

Tabla 13: Caso de prueba#16: Gestionar trámites.

3.4.3 Resultados de las pruebas de caja negra

Se evaluaron los valores de entrada para cada escenario de los casos de prueba, donde se validó que el comportamiento del sistema para estos valores en las salidas fueran resultados esperados por el cliente. Estas pruebas fueron realizadas a medida que fue concluyendo la implementación de cada una de las iteraciones, en estas se encontraron mayormente errores ortográficos, de validaciones y errores en las interfaces. En la primera iteración se detectaron un total de 25 no conformidades, de las cuales 13 fueron resueltas satisfactoriamente, arrojando un 52% de no conformidades resueltas con éxito. En la segunda iteración se corrigieron los errores de la

anterior, se encontraron 9 no conformidades, donde 7 fueron resueltas satisfactoriamente, quedando evidente que el 77,8% se resolvió con éxito. Por último, en la tercera iteración se corrigieron los errores pendientes de la segunda iteración y no se detectaron nuevas no conformidades, por lo que quedaron resueltas el 100% de las mismas con el éxito esperado.

A continuación se describe una tabla con las no conformidades encontradas por iteración, las no conformidades solucionadas y el porcentaje de pruebas exitosas.

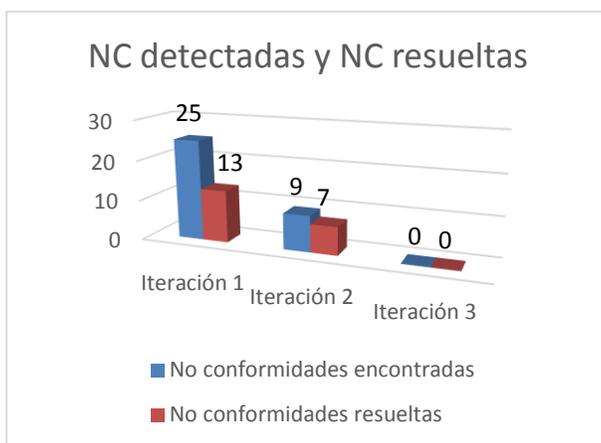


Figura 11: NC detectadas y resueltas por iteración.

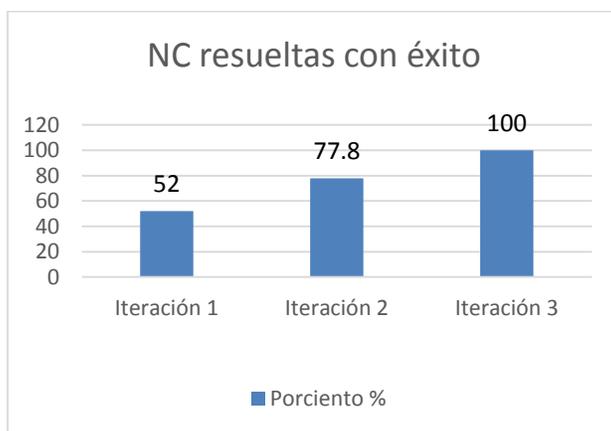


Figura 12: NC resueltas con éxito.

3.5 Prueba de caja blanca

Se realiza para comprobar que las funciones internas del portal se ejecutan correctamente y para asegurar que se ajustan a las especificaciones, verificando que todos los componentes internos se han probado de forma adecuada. Estas pruebas intentan garantizar que:

- ✓ Se ejecutan al menos una vez todos los caminos independientes de cada módulo.
- ✓ Se utilizan todas las estructuras de datos internas.
- ✓ Definir todos los caminos lógicos.

- ✓ Desarrollar casos de prueba para todos los caminos lógicos.
- ✓ Evaluar los resultados.

3.5.1 Técnica de prueba: Camino básico

El método del camino básico permite obtener una medida de la complejidad de un diseño procedimental y utilizar esta como guía para la definición de una serie de caminos básicos de ejecución, diseñando casos de prueba que garanticen que cada camino se ejecuta al menos una vez. La generación del grafo de flujo tiene que descomponer las condiciones compuestas en condiciones sencillas (10).

Se analiza el código que se encuentra en el [Anexo 4: Obtener Form\(\)](#) y se procede a representar el grafo, donde cada nodo representa una o más sentencias procedimentales y un solo nodo puede corresponder a una secuencia de pasos del proceso y a una decisión, las flechas (aristas) representan el flujo de control.

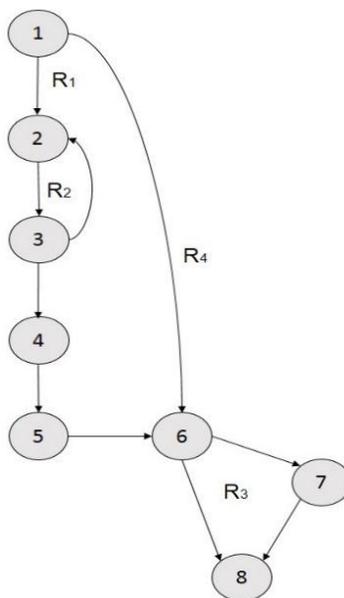


Figura 13: Grafo de flujo.

Complejidad ciclomática:

Es una medida que proporciona una idea de la complejidad lógica de un código, esta coincide con el número de regiones del grafo de flujo. Luego del análisis realizado al grafo de flujo de la figura 10 la complejidad ciclomática sería:

1. $V(G) = \text{Aristas} - \text{Nodos} + 2$
 $V(G) = 10 - 8 + 2 = 4$
2. $V(G) = \text{Nodos de predicado} + 1$



$$V(G) = 3 + 1 = 4$$

3. $V(G)$ = Cantidad de regiones

$$V(G) = 4$$

A partir del valor de la complejidad ciclomática se obtiene el número de caminos independientes, que brindan un valor límite para el número de pruebas que se tienen que diseñar, en este caso sería $V(G) = 4$.

1	1 - 6 - 8
2	1 - 6 - 7 - 8
3	1 - 2 - 3 - 4 - 5 - 6 - 8
4	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8

Tabla 14: Caminos linealmente independientes.

3.6 Pruebas de carga y estrés

Las pruebas de carga y estrés se realizan con el fin de medir la respuesta de la aplicación a distintos volúmenes de carga esperados, como la cantidad de usuarios y peticiones emitidas al sistema. Se ejecutan para encontrar el volumen de datos o de tiempo en que la aplicación comienza a fallar o es incapaz de responder a las peticiones. Con estas pruebas se simula la concurrencia de forma que se acerque al resultado esperado en la explotación real del portal. Para esto se debe preparar un caso de prueba para cada módulo y sus funcionalidades y después cambiarle los escenarios.

Para la elaboración de esta prueba se utiliza la herramienta Apache Jmeter 2.9, una aplicación de escritorio construida en Java para probar software, es muy utilizada para aplicaciones web y permite la generación de reportes con los resultados obtenidos (48).

3.6.1 Resultados de las pruebas de carga y estrés

Al ejecutar las pruebas de carga y estrés se tuvo en cuenta los requisitos de hardware y de rendimiento definidos, se analizaron las prestaciones, el rendimiento de los recursos dinámicos y estáticos, emulando grandes niveles de concurrencia sobre el servidor del portal del MINJUS de Cuba y la base de datos (49).

Se seleccionaron un número de 1000 usuarios simultáneos para probar la secuencia de peticiones HTTP que se pueden realizar. Los casos de pruebas escogidos son Gestión de



Portal web del Ministerio de Justicia de la República de Cuba.

Trámites, Servicios y Registros, con sus escenarios y funcionalidades. Los resultados arrojados fueron:

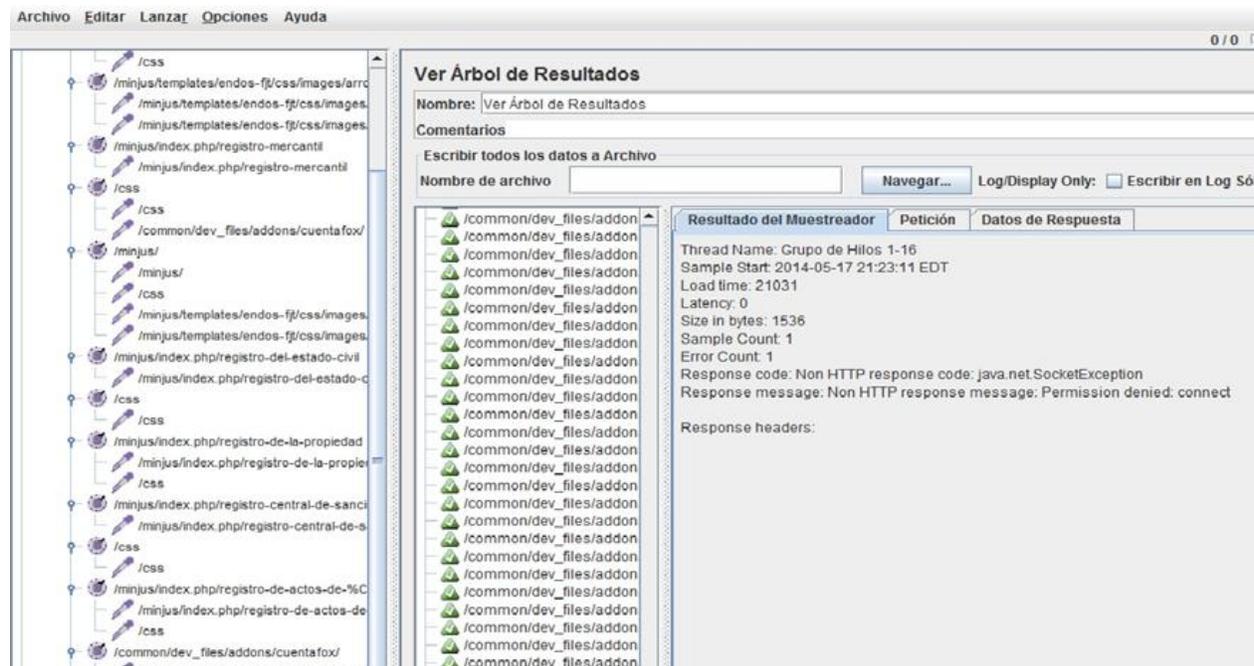


Figura 14: Resultados del Jmeter.

Las salidas finales del informe agregado por módulos son las siguientes:

Label:	# Muestras:	Media:	Mín.:	Máx.:	Error %:	Rendimiento:
Trámites	18 500	29 769	0	391 531	0%	22.6 s
Registros	29 500	5 686	0	200 243	0%	20 s
Servicios	10 000	14 368	0	21 376	0%	21.5 s

Tabla 15: Salidas del informe agregado.

Donde:

Label: nombre de la muestra o del conjunto de muestras.

Muestras: número de muestras para cada URL. Es el número resultante de la multiplicación de la cantidad de usuarios por el número de repeticiones de las pruebas.

Media: tiempo medio transcurrido para un conjunto de resultados, en milisegundos.

Mín.: mínimo tiempo transcurrido para las muestras de la URL dada, en milisegundos.

Máx.: máximo tiempo transcurrido para las muestras de la URL dada, en milisegundos.

Error %: porcentaje de las peticiones con errores.

Rendimiento: rendimiento medido en base a peticiones por segundo/minuto/hora.



Portal web del Ministerio de Justicia de la República de Cuba.

Concluyendo que el rendimiento no excede de los 25 segundos para la cantidad de muestras simuladas en el Jmeter.

3.7 Prueba de usabilidad

La usabilidad se refiere a la facilidad o nivel de uso, al grado en el que el diseño de un software facilita o dificulta su manejo (50). La prueba de usabilidad es una técnica usada para evaluar si un producto cumple con el propósito para el cual fue diseñado, en este caso el portal del MINJUS de Cuba. Estas pruebas miden la usabilidad y facilidad de uso y se realizan a través de diferentes métodos para evaluar una interfaz de usuario sin involucrar a usuarios reales (51).

3.7.1 Resultados de la prueba de usabilidad

Las pruebas de usabilidad fueron realizadas midiendo el comportamiento del software a través de la Lista de Chequeo Usabilidad de Sitios Web de CALISOFT (Centro Nacional de Calidad de Software). Con esta planilla se recogen los puntos eficientes y los ineficientes que tienen los elementos comprobados. Está compuesta por preguntas para cada uno de los indicadores a los cuales se les da una evaluación y son recogidos la cantidad de elementos afectados.

A continuación se muestra un gráfico con el resultado para cada uno de los indicadores de las pruebas de usabilidad, los indicadores se encuentran del lado izquierdo y en la derecha aparecen las barras que indican el resultado obtenido. Se evalúan de bien los que tienen más del 75% de las preguntas de los indicadores evaluados satisfactoriamente, si son menores serán evaluados de mal.

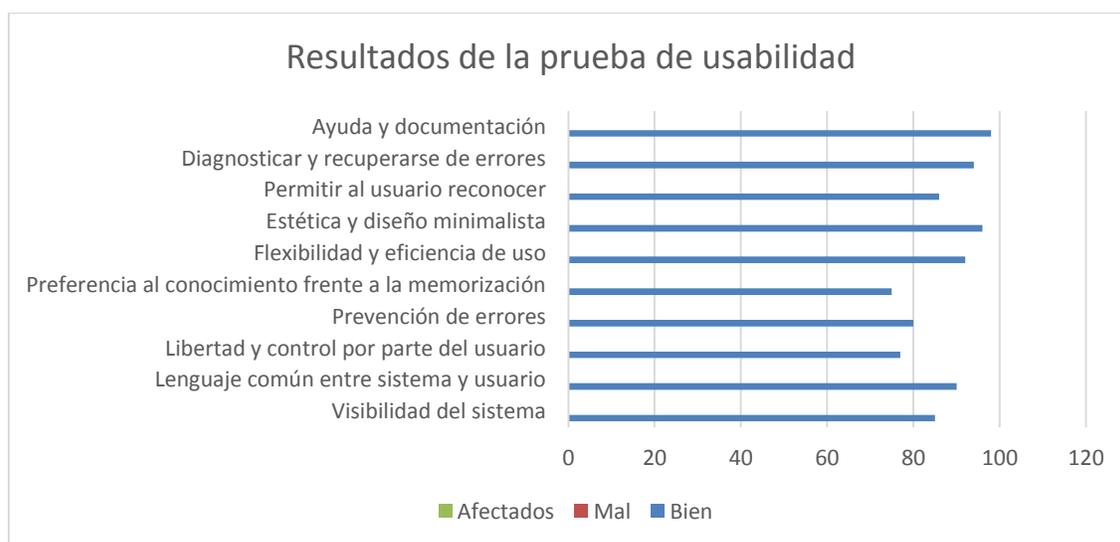


Figura 15: Resultados de la prueba de usabilidad.



3.8 Prueba de seguridad

La creciente complejidad del software hace que sea casi imposible garantizar que esté libre de vulnerabilidades. Las pruebas de seguridad están encaminadas a verificar que un sistema de información proteja sus datos y funcione de acuerdo a los propósitos para los que fue diseñado. Estas buscan medir la confidencialidad, integridad y disponibilidad de los datos, se identifican amenazas y riesgos existentes desde el uso de la interfaz de usuario final. Permiten además, evaluar el nivel de resistencia del sistema ante posibles ataques, identificar vulnerabilidades y puntos de acceso a nivel de infraestructura e implementar buenas prácticas para prevenirlos. Una vez ejecutadas las pruebas de seguridad es posible medir y cuantificar los riesgos a los cuales se ven expuestas las aplicaciones, tanto en la infraestructura interna como externa.

El beneficio de las pruebas de seguridad es:

- ✓ Medir la exposición real a posibles ataques de intrusos.
- ✓ Controlar accesos no autorizados a información sensible.
- ✓ Contar con estrategias para la mitigación de riesgo de explotación de vulnerabilidad.
- ✓ Asegurar que los riesgos de seguridad identificados se mantienen en niveles aceptables y medibles y por tanto lograr protección real de los datos.

3.8.1 Resultados de la prueba de seguridad

Se utilizó la Lista de Chequeo de Pruebas de Seguridad para portales web que ofrece CALISOFT y se verificó el cumplimiento de los requisitos no funcionales de seguridad. Luego de la comparación de los elementos de esta planilla con el sistema se concluye que se garantiza la confidencialidad de la información, ya que se evita que usuarios no autorizados accedan a la misma. El indicador definido es el de pruebas de autenticación, el personal para acceder debe entrar su usuario y contraseña y se muestran mensajes de error al autenticarse incorrectamente.

Se garantiza la integridad de la información ya que se puede asegurar la exactitud de la información, así como los métodos que se utilizan para su procesamiento. El indicador definido fue el de validación de datos, ya que se enmascaran datos confiables cuando se visualizan en la aplicación, como la contraseña. El sistema muestra mensajes de error en campos de entrada para los nombres, apellidos, carné de identidad y correo electrónico. Permite la funcionalidad de cambio de contraseña únicamente a usuarios autenticados, verificando la antigua y la nueva contraseña.



Portal web del Ministerio de Justicia de la República de Cuba.

También se garantiza la disponibilidad de la información ya que está accesible para usuarios y sistemas autorizados en el momento que lo requieran. Los indicadores definidos son los de pruebas de autenticación y de gestión de sesión de cuentas. Se bloquea la cuenta del usuario después de un número de 3 intentos fallidos.

3.9 Validación de la variable de la investigación

Variable independiente: desarrollo de un portal web.

Variables dependientes: gestión de la información y gestión de los trámites.

Después de definir la variable independiente y las dependientes extraídas de la idea a defender y del objetivo general de la investigación, se expone una tabla con las variables dependientes, con la cual se mide el estado actual y el estado en el que se encontrarían una vez puesto en práctica el sistema, además de centrar el objetivo de desarrollo de las mismas.

Variables dependientes	Antes	Después
Gestión de la información	La información presentada en el portal no se gestionaba de forma organizada, por lo cual los ciudadanos debían llegar hasta los centros jurídicos para asesorarse de cómo realizar las actividades, lugares y registros que debían visitar y solicitar. No existía un directorio electrónico con los datos de localización de las oficinas registrales.	El portal mejorará la gestión de los servicios que brinda el MINJUS de Cuba y ofrecerá un mayor número de estos. Para la gestión de la información se realiza el Directorio del portal, el cual tendrá los datos de localización de las oficinas de los registros provinciales.
Gestión de los trámites	Las solicitudes de trámites debía ser solo personal en los centros jurídicos, no se podían realizar por vía correo electrónico. Los ciudadanos o usuarios no podían ver el estado en el que se encontraban sus trámites sin asistir a la institución.	El portal mejorará la gestión de los trámites que brinda el MINJUS de Cuba y ofrecerá un mayor número de estos.

Tabla 16: Variables dependientes de la investigación.



Portal web del Ministerio de Justicia de la República de Cuba.

Para verificar la mejora de la gestión de la información, servicios y trámites del portal web del MINJUS de Cuba, se evaluarán las variables a través de los indicadores que se muestran a continuación:

Indicadores	Antes	Después
Presentación del contenido	La presentación del contenido es limitada, presenta espacios en blanco innecesarios, páginas huérfanas y un lenguaje algunas veces poco entendible. Las imágenes en artículos y otras informaciones son limitadas y restringidas.	Está exento de páginas huérfanas, espacios en blanco innecesarios y presenta un lenguaje entendible.
Administración de servicios	Los servicios del portal eran limitados y carecían de actualización y enriquecimiento de su contenido.	Permite la inclusión de nuevos servicios y la actualización de estos se realiza de forma fácil y sencilla.
Navegabilidad	Posee sistemas de navegación global y local, sin embargo imposibilita al usuario conocer dónde se encuentra y dónde ha estado.	Después de estudiar la arquitectura de la información se realiza un sistema de navegación jerárquico, global y local. Le garantiza al usuario en todo momento conocer dónde se encuentra y dónde ha estado e indicándole en todo momento hacia dónde puede ir.
Confiabilidad de la información	Se garantiza la confidencialidad e integridad de la información ya que se requiere de la autenticación para administrar el sitio por parte de los usuarios.	Se garantiza la confidencialidad e integridad de los datos, se realiza el control de acceso basado en roles y permisos y se proporcionan mecanismos de defensa contra ataques de inyección de código SQL.

Tabla 17: Medición de los indicadores.



3.10 Conclusiones del capítulo

Mediante las métricas TOC y RC quedó validado el diseño, obteniéndose resultados satisfactorios en cuanto al grado de complejidad, implementación, mantenimiento, responsabilidad, reutilización y acoplamiento. Además, la cantidad de pruebas necesarias para realizar a las clases favoreció la creación de un diseño sencillo.

Se evaluó el cumplimiento de las funcionalidades del sistema de acuerdo al análisis de las pruebas de caja negra, que fueron aplicadas mediante la técnica de partición de equivalencia, lográndose con las mismas la calidad requerida. Las pruebas de caja blanca sirvieron para comprobar internamente las funciones del portal, facilitando la detección de no conformidades para su corrección.

Con la validación de las variables dependientes de la investigación mediante los indicadores, se pudo demostrar que el empleo del portal web para el MINJUS de Cuba cubrirá en gran medida las principales necesidades de usuarios nacionales e internacionales y mejorará la gestión de los servicios de información y de trámites.



Conclusiones generales

Con la gestión de la información y de trámites del MINJUS de Cuba en el presente trabajo se arribó a las siguientes conclusiones:

- ✓ Mediante el estudio del estado del arte se obtuvieron las tendencias de la forma de estructurar y organizar los portales web dedicados a la divulgación y promoción de la información jurídica.
- ✓ El análisis de las diferentes metodologías y herramientas permitió seleccionar las más adecuadas para el desarrollo del portal web.
- ✓ La especificación de los requisitos funcionales y no funcionales a partir de los métodos empíricos de entrevistas, tormenta de ideas y sistemas existentes, permitió lograr un producto con la calidad requerida.
- ✓ Con la aplicación de la metodología seleccionada se obtienen como resultados los artefactos correspondientes a cada una de las fases.
- ✓ Mediante la planificación, diseño, implementación, validación y puesta en práctica del sistema se mejora la gestión de los servicios de información y de trámites en el MINJUS de Cuba.



Recomendaciones

- ✓ Añadir nuevas funcionalidades de acuerdo a las necesidades crecientes del cliente, que permitan ampliar los servicios y recursos que brinda el portal web.
- ✓ Integrar al módulo Trámites los procesos restantes del MINJUS de Cuba para que los ciudadanos puedan realizarlos de forma rápida y con calidad.
- ✓ Mantener las actualizaciones del CMS Joomla y de las extensiones instaladas para la correcta administración del portal.
- ✓ Continuar la gestión de la información de forma organizada y actualizada a medida que crezcan los servicios y prestaciones del MINJUS de Cuba.



Bibliografía

1. Peláez, Carlos A. *Incorporación de sistemas informáticos de gestión en el Poder Judicial Boliviano*. Bolivia : s.n., 2002.
2. Hernández León, Rolando Alfredo y Coello González, Sayda. *El proceso de Investigación Científica*. La Habana : Universitaria del Ministerio de Educación Superior, 2011.
3. Álvarez de Zayas, Dr. Cs. Carlos . *Metodología de la Investigación Científica*. Santiago de Cuba : s.n., 1995.
4. Ministerio de Justicia de la República de Cuba. [En línea] 2012-2014. <http://www.minjus.cu/news.php>.
5. EcuRed. [En línea] 10 de 2012. http://www.ecured.cu/index.php/Inform%C3%A1tica_jur%C3%ADdica.
6. OECD. *The e-government imperative: main findings*. PolicyBrief, PublicAffairsDivision, PublicAffairs and CommunicationsDirectorate. 2003.
7. Mínguez Sanz, Daniel y García Morales, Emilio José. *Metodologías para el desarrollo de aplicaciones web*.
8. Definición.DE. *Definición de*. [En línea] 2008-2014. <http://definicion.de/portal/>.
9. Sommerville, Ian. *Ingeniería de Software. 7ma Edición*. 2005.
10. Pressman, Roger S. *Ingeniería del Software. 6th Edición*. s.l. : McGraw-Hil.
11. Ronda León, Rodrigo. *Arquitectura de Información: análisis histórico-conceptual*. 2008. ISSN 1886-8592.
12. Cáceres, Paloma y Marcos, Esperanza. *Procesos ágiles para el desarrollo de aplicaciones web*. España : s.n.
13. H. Canós, José , Letelier, Patricio y Penadé, María Carmen . *Métodologías Ágiles en el Desarrollo de Software*. Valencia : s.n.
14. Calderón, Amaro , Dámaris, Sarah y Valverde Rebaza, Jorge Carlos. *Metodologías Ágiles*. Perú : s.n., 2007.
15. Calero Solís, Manuel. *Una explicación de la Programación Extrema (XP)*. Madrid : s.n., 2003.
16. Pérez Pérez, María José. *Guía comparativa de Metodologías Ágiles*. Segovia : s.n.



17. ConML. ConML. *ConML*. [En línea] 2010-2013. [Citado el: 30 de 11 de 2013.] <http://www.conml.org/FAQ.aspx>.
18. Hernández Orallo, Enrique. *El Lenguaje Unificado de Modelado (UML)*.
19. Unified Modeling Language (UML). *Unified Modeling Language (UML)*. [En línea] 1997-2014. [Citado el: 23 de 10 de 2013.] <http://www.uml.org/>.
20. González López, Pascual, Gonzáles López, Ana Amelia y Gallud Lázaro, Jose Antonio. *Herramientas CASE*.
21. Visual Paradigm for UML. [En línea] 2004. [Citado el: 27 de 11 de 2013.] <http://www.visual-paradigm.com/product/vpuml>.
22. Rojas. *Sistemas gestores de contenidos*. 2010.
23. Drupal, History. History. Drupal. [En línea] [Citado el: 25 de Octubre de 2013.] <http://drupal.org/node/769>.
24. (CMS), Diseñando webs con gestores de contenidos. Diseñando webs con gestores de contenidos (CMS). [En línea] [Citado el: 2 de Noviembre de 2013.] <http://luisalarcon.com/blog/?p=450>.
25. Joomla. ¿Qué es Joomla? *Joomla.org*. [En línea] Noviembre de 2011. <http://www.joomla.org>.
26. WordPress. WordPress en español. *WordPress.org*. [En línea] [Citado el: 22 de Noviembre de 2013.] <http://es.wordpress.org>.
27. Gilfillan, Lan. *La Biblia de MYSQL*. s.l. : Anaya, 2009.
28. Rodríguez Sala. *Lenguajes de Programación*. 2003.
29. Palomo Duarte, Manuel y Montero Pérez, Ildfonso. *Programación en PHP a través de ejemplos*. España : s.n.
30. Menéndez, Rafael y Asensio, Barzanallana. *Lenguajes de programación HTML y CSS. Parte 1: Conceptos básicos de HTML*.
31. —. *Lenguajes de programación HTML y CSS. Parte 2. Formato con CSS*.
32. Pérez, Javier Eguíluz. *Introducción a CSS*. 2009.
33. NetBeans. MicroSystems. *NetBeans*. [En línea] Oracle, 2013. [Citado el: 25 de Octubre de 2013.] http://netbeans.org/index_es.html.
34. Pressman, Roger S. *Ingeniería de Software. Un enfoque Práctico*. 2005.



35. Fernández Escribano, Gerardo . *Introducción a Extreme Programming. Ingeniería de Software II*. 2002.
36. Rodríguez Corbea, Maité y Ordoñez Pérez, Meilín. *La Metodología XP aplicada al desarrollo de software educativo en Cuba*. La Habana : s.n., 2007.
37. Lerner, P. *La Arquitectura de la Información como estrategia comunicacional*. 2005.
38. Ronda León, P. *La Arquitectura de la Información y las Ciencias de la Información*. 2005.
39. Kruchten, Philippe. *Architectural Blueprints. The "4+1" View Model of Software Architecture*. 1998.
40. *Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web*. Guerrero, Carlos A. y M. Suárez, Johnna. 3, Colombia : La Serena, 2013, Vol. 24.
41. Mühlrad, Daniel. *Patrones de Diseño*. 2008.
42. Larman, Craig. Cap. VI Fase de Diseño. *UML y Patrones. Introducción al análisis y diseño*.
43. Craig Larman, Prentice Hall. *El modelo de diseño. UML y patrones 2da Edición*. 2003.
44. Bakken, Stig S. Pear. *Pear*. [En línea] 14 de Marzo de 2007. [Citado el: 25 de Febrero de 2013.] <http://pear.php.net>.
45. Sánchez Fornaris, Maité y Alcantara Rabí, Dayanis. Propuesta de una guía de métricas para evaluar el desarrollo de los Sistemas de Información Geográfica. [En línea] 2010. [Citado el: 9 de Mayo de 2014.] http://vinculando.org/articulos/sociedad_america_latina/propuesta_guia_de_medidas_para_evaluacion_sistemas_informacion.html.
46. McGraw-Hill. *Ingeniería del Software. Un enfoque práctico. 6ta edición*. 2007.
47. CIBERTEC. *Pruebas de software*. 2011.
48. Apache Software Foundation. Apache Jmeter. *Apache Jmeter*. [En línea] 2013. [Citado el: 05 de 05 de 2014.] <http://jakarta.apache.org/jmeter/>.
49. Martínez, Ander. *Apache Jmeter. Manual de Usuario v1.2*. 2009.
50. Nielsen, Jakob y Kauffman, Morgan. *Usability Engineering (Interactive Technologies)*. 1994. ISBN 0-12-518406-9.
51. Charlton, Samuel G. y O'Brien, Thomas G. *Handbook of Human Factors Testing and Evaluation*. s.l. : LEA Inc., 2001. ISBN 0-8058-3290-4.
52. Canós, J. *Metodologías Ágiles*. 2005.



53. Pérez, Javier Eguíluz. *Introducción a JavaScript*. 2008.
54. M. A. Álvarez. *Introducción a PHP 5. Introducción a PHP 5*. [En línea] 2004.
<http://www.desarrolloweb.com/articulos/1696.php>.
55. P., Lerner. *La Arquitectura de la Información como estrategia comunicacional*. 2005.
56. Java. Java. [En línea] [Citado el: 25 de Octubre de 2013.]
http://www.java.com/es/download/faq/java_javascript.xml.

Glosario

HTTP (Protocolo de transferencia de hipertexto): establece un esquema de comunicación cliente-servidor. El cliente es el navegador web que realiza las peticiones a las que el servidor se encarga de dar respuesta.

TCP/IP (Protocolo de Internet): se utiliza en la comunicación entre el servidor y la base de datos para realizar operaciones sobre la información de las tablas.

Servidor web apache: representa una estación donde se instala el servidor sobre el cual se despliega la aplicación.

Servidor de base datos MySQL: representa el servidor donde se instala este y como responde a las peticiones hechas por la aplicación.

ORM: en sus siglas en inglés Object Relational Mapping, en español Mapeo de Objetos Relacionales. Es un componente de software que permite trabajar con datos persistentes que forman parte de una base de datos.