

Universidad de las Ciencias Informáticas
Facultad 3



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

Título: Desarrollo del Portal Web de la Organización Nacional
de Bufetes Colectivos

***Autores:** Mirileydis Martínez Ajete*


Dayana Francia Ofarrill

***Tutores:** MSc. Yarina Amoroso Fernández*

Ing. Reinier Fernández Coello

La Habana 2014

“Año 56 de la Revolución”



“Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio, consciente de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.”

Ernesto “Che” Guevara de la Serna

Declaración de autoría

Declaramos ser autoras de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Dayana Francia Ofarrill
Firma del autor

Mirileydis Martínez Ajete
Firma del autor

MSc. Yarina Amoroso Fernández
Firma del tutor

Ing. Reinier Fernández Coello
Firma del tutor

Agradecimientos

A la persona más importante de mi vida, esa persona fuerte que ha sabido ser madre y padre durante 20 años para mi hermana y para mí, a ti mi mamá bella que adoro.

A papá que quiero con la vida por siempre exigir el máximo en los estudios.

A mi confidente de toda la vida a pesar de las peleas, mi hermana Marielys.

A mis tíos Roberto, Ester, Eufemio, La china (que aunque no es mi tía la quiero como tal), Mena, en fin a todos.

A mi abuela Ada por estar siempre conmigo y a mi abuela Coco que aunque no está siempre la recuerdo con mucho cariño.

A toda mi familia que ha estado siempre para apoyarme.

A mi novio Hamdy que ha estado conmigo en los momentos más difíciles en estos 3 cursos, te amo.

A mis tutores por la confianza y apoyo, en especial a Reinier que no solo supo guiarnos, sino también supo hacernos sonreír cuando pasábamos por tiempos de tensión y estrés.

A mi compañera de tesis, que la vida quiso que naciera una amistad de dos historias similares en esta universidad.

A Carlos Cordoví por ayudarnos con el documento.

A mis amistades en estos años de carrera que han sabido ganarse mi cariño, los llevo en mi corazón Dayana (Nana), Elizabeth, Noraivis, Claudia Salcedo, Claudia Guerra, Angélica, Liseth Schery, a todas gracias.

A Daymaris por estar al tanto de todas las cosas de la tesis y por apoyarme en los momentos en los no podía pensar en otra cosa que no fuera la tesis.

A mis dos madres aquí en la UCI, ustedes que siempre me han apoyado y aconsejado a lo largo de mi carrera las quiero mucho, Isabel y Made.

A todas las personas que hicieron posible este momento gracias.

A todos muchas gracias.

Miry

*A mi ángel de la guarda por estar siempre a mi lado y protegerme en cada paso que doy en esta vida mi
mamita.*

A mi papa, hermano, abuelo y familia en general por su apoyo.

*A mis viejas amistades que en estos momentos no se encuentran cerca pero que de una manera u otra seguimos la
comunicación (Liu, Trina, Rosalia, Fiset, Fadi, Moatira, Josbelito, Javier, Fohamis).*

*A mis nuevas amistades que en los últimos 2 cursos de carrera dios quiso que compartiera con ellos de una
manera u otra (Rosmary, Alfonso, Ledier, Miguel Ángel)*

A mis compañeros de aula en especial a mi compañera de tesis, Gabriel y Juan.

*A mi amiga o hermanita Olivia por aconsejarme, regañarme y a su familia por acogerme como si fuera un
miembro más de la familia.*

*A mis vecinos por su preocupación en especial a Beatriz, Maira, Ibdelisa, Dalia, Ramel, Mexci y
Lazara.*

A mis suegros por apoyarme y preocuparse por mí en todo momento, por su gran ayuda.

*A mi querido novio Quirola (mi cosita) por su amor, paciencia, por estar siempre a mi lado en los momentos de
alegría y tristeza y por soportarme en estos 5 años jajajaja, te quiero mucho.*

A los tutores por la ayuda en la realización de este sueño.

A todo en el que me preguntó y la tesis!!! ¿Cómo va?

A todos ustedes gracias.

Dayana

Dedicatoria

A mi madre por ser mi apoyo incondicional a lo largo de todos los años.

A mi padre por exigir más cada día.

A mi hermana que a pesar de pelear bastante, siempre me ha apoyado en todo.

A toda mi familia: tíos abuelos, primos, en fin a todos.

A mi novio para servirle de inspiración en lo que le queda de carrera.

A Daimarys, Angélica y Claudia para que vean que si yo pude ellas también van a poder llegar con esfuerzo, entrega y dedicación.

A Thalía y Dailyn para que de una forma u otra sigan mi ejemplo de superación y de ser independientes.

Miry

A todos los que permitieron lograr este sueño.

Dayana

Resumen

La Organización Nacional de Bufetes Colectivos (ONBC) es una organización que tiene dentro de sus objetivos: prestar un servicio o servicios jurídicos con calidad a personas naturales y jurídicas, contando con profesionales calificados en todas las ramas del Derecho (ONBC, 2009). Actualmente la ONBC posee un portal web en Internet el cual es estático que no cuenta con un sistema de actualización que permita conocer la cuantía de la obsolescencia de la información, no cuenta con los servicios en línea y presenta problemas a la hora de autenticarse o crear nuevas cuentas de usuario. Además, este no satisface sus necesidades debido a que las fallas que poseen dichos servicios limitan la administración del sitio, no considerándose lo suficientemente útil como medio de comunicación.

Por esta razón en la presente investigación se tiene como propuesta de solución un portal web que contribuirá a la mejora del canal de comunicación y servicios a la comunidad de usuarios de dichas entidades.

El resultado obtenido facilitará la gestión de los servicios que brinda y ofrecerá un mayor número de estos a los usuarios.

Palabras clave: Organización Nacional de Bufetes Colectivos, portal web, servicios, canal de comunicaciones.

Índice de contenidos

Introducción	1
Capítulo 1: Fundamentación teórica.....	5
1.1 Introducción	5
1.2 Conceptos fundamentales.....	5
1.3 Tendencias actuales	6
1.3.1 Portal Web del Bufete Internacional de la Habana (La Habana, 2009)	7
1.3.2 Portal de la Abogacía Española (España, 2010).....	7
1.3.3 Portal Web Alpha Abogados Perú (Perú, 2010).....	7
1.3.4 Resultados arrojados del estudio de sistemas homólogos.....	7
1.4 Metodología de Desarrollo de Software	9
1.4.1 Programación extrema (XP)	9
1.4.2 SCRUM	10
1.4.3 SXP	10
1.5. Herramientas para el desarrollo de la solución.....	12
1.5.1. Sistema gestor de contenidos.....	12
1.5.1.1. Joomla 1.7.3	13
1.5.1.2. Drupal 7.24	13
1.6. Lenguaje de Modelado UML 2.0	14
1.7. Lenguajes de programación.....	15
1.7.1. PHP 5.3.8.....	15
1.7.2. HTML 5.....	16
1.7.3. CSS (Hojas de Estilos en Cascada) 3.....	16

1.7.4.	JavaScript 1.8.5.....	17
1.8.	Herramienta para modelado de prototipos web.....	17
1.9.	Herramienta CASE.....	18
1.9.1.	Herramienta CASE para UML.....	18
1.10.	Sistema Gestor de Base de Datos.....	19
1.10.1.	MySQL 5.5.16.....	19
1.10.2.	PostgreSQL.....	20
1.11.	Servidor Web Apache 2.2.21.....	21
1.12.	Entorno de Desarrollo Integrado.....	22
1.12.1.	Netbeans 7.3.....	22
1.13.	Conclusiones parciales.....	23
	Capítulo 2: Planificación-Definición del Portal Web.....	24
2.1.	Introducción.....	24
2.2.	Descripción del negocio.....	24
2.3.	Flujo actual de procesos.....	24
2.4.	Propuesta de Solución.....	26
2.5.	Personal relacionado con el sistema.....	26
2.6.	Requisitos.....	27
2.6.1.	Lista de Reserva del producto.....	28
2.6.2.	Requisitos no funcionales.....	29
2.7.	Historias de Usuario.....	30
2.8.	Estimación de esfuerzos por HU.....	31
2.9.	Plan de Iteraciones.....	32

2.10.	Plan de duración de las iteraciones.....	34
2.11.	Arquitectura del sistema.....	35
2.11.1.	Arquitectura del CMS Drupal.....	35
2.11.2.	Arquitectura de Información del sistema de la ONBC.....	39
2.12.	Patrones de diseño	40
2.12.1.	Patrones GOF	40
2.13.	Diagrama de paquetes.....	41
2.14.	Conclusiones parciales	43
Capítulo 3: Implementación y prueba		44
3.1.	Introducción	44
3.2.	Técnicas de validación de requisitos.....	44
3.3.	Métricas aplicadas a los requisitos.....	45
3.4.	Implementación.....	47
3.4.1.	Iteración 1.....	47
3.4.2.	Iteración 2.....	48
3.4.3.	Iteración 3.....	48
3.4.4.	Plan de liberación	49
3.4.5.	Estándares de codificación empleados.....	50
3.5.	Diagrama de despliegue	51
3.6.	Estrategia de prueba.....	53
3.6.1.	Método de caja negra.....	54
3.6.2.	Método de caja blanca.....	55
3.6.3.	Usabilidad.....	57

3.7. Conclusiones parciales	58
Conclusiones generales.....	59
Recomendaciones	60
Referencias bibliográficas	61
Anexos	64
Anexo 1. Glosario de términos	64
Anexo 2. Fases y artefactos de la metodología SXP	65
Anexo 3. Requisitos funcionales del sistema para la ONBC.....	66
Anexo 4: Requisitos no funcionales.....	71
Anexo 5: Descripción de los elementos de la estructura del sistema.....	73
Anexo 6: Preguntas de chequeo para la validación de los requisitos	74
Anexo 7: Tareas de ingeniería	75
Anexo 8: Descripción del caso de prueba de aceptación número 01 de la HU # 05	78
Anexo 9: Interfaz “Certificación de documentos”	79
Anexo 10: Formulario para crear Certificación de matrimonio en Cuba	80
Anexo 11: Resultado del caso de prueba 0401	83

Índice de tablas

Tabla 1: Personas relacionadas con el sistema	26
Tabla 2: Descripción de la Historia de usuario	30
Tabla 3: Historia de usuario "Catálogo en línea"	31
Tabla 4: Estimación de esfuerzo por HU	31
Tabla 5: Plan de duración de las iteraciones	34
Tabla 6: Tareas generadas por la historia de usuario Gestionar usuario.....	47
Tabla 7: Tareas generadas por la historia de usuario Servicio de información.....	48
Tabla 8: Tareas generadas por la historia de usuario Gestionar encuesta.....	48
Tabla 9: Plan de liberación	49

Índice de imágenes

Imagen 1: Modelo de dominio.....	25
Imagen 2: Flujo de ejecución del patrón MVC.....	36
Imagen 3: Modelo de arquitectura de Drupal	37
Imagen 4: Diagrama de Paquetes del sistema.....	42
Imagen 5: Diagrama de despliegue	52
Imagen 6: Resultados de la prueba	54
Imagen 7: Función que devuelve la tabla donde se recogen los registros una vez agregados a la cesta.	56
Imagen 8: Grafo de flujo asociado al método uc_cart_view_table.....	56
Imagen 9: Resultados de la prueba de Usabilidad.....	58

Introducción

Con la aparición de las Tecnologías de la Información y las Comunicaciones (TIC) se ha ido cambiando la manera de trabajar y gestionar recursos. Ellas son un elemento fundamental para hacer que el trabajo sea más productivo, pues agilizan las comunicaciones y aumenta el trabajo en equipo, además permiten realizar constantes cambios en los productos de software.

Las TIC están transformando la sociedad y en particular los procesos educativos, para responder a estos desafíos que demandan cambios en los sistemas educativos y promueven experiencias innovadoras en los procesos de enseñanza y aprendizaje. Ellas cumplen un papel determinante en la producción y en el desarrollo social. En países desarrollados y algunos en desarrollo, los gobiernos han visualizado como instrumentos que contribuyen al logro de amplios objetivos nacionales y como soporte de políticas nacionales y programas de desarrollo. Las TIC han sido un factor determinante para afrontar problemas que surgen de la complejidad del mundo actual.

Cuba, por ser un país subdesarrollado tiene que hacer un gran esfuerzo para lograr informatizar su sociedad mediante el uso de las TIC. La Universidad de las Ciencias Informáticas (UCI) es un eslabón importante en este aspecto debido a la fuerte vinculación que tienen sus estudiantes con las TIC. Además, se encuentra inmersa en el desarrollo de productos para informatizar la sociedad cubana e impulsar las exportaciones de software.

En la actualidad es una necesidad para todas las empresas o instituciones tener un sitio o portal web en el cual puedan mostrar y gestionar todo tipo de información relacionadas con sus productos, servicios o negocios en general. Algunas instituciones en Cuba se han propuesto la actualización de sus portales web haciendo uso de estándares web actuales con el propósito de agilizar y controlar los procesos; una de las instituciones involucradas en dicho perfeccionamiento es la Organización Nacional de Bufetes Colectivos (ONBC).

Actualmente la ONBC posee un portal web en Internet el cual es estático y presenta problemas como:

No permite autenticarse correctamente ni crear nuevas cuentas de usuarios, lo que dificulta saber qué tipo de información va a estar disponible para los distintos tipos de usuarios. Los cursos que se ofrecen

carecen de información para los usuarios y no cuentan con un sistema de actualización que permita conocer la cuantía de la obsolescencia de dicha información, por lo que los usuarios de la institución no saben qué tan obsoleta es la información que brinda el sistema. No muestra información sobre los abogados con los que cuenta un bufete colectivo ni de los casos que atienden en determinado momento, dificultando que los clientes que visiten el portal sepan a qué abogados recurrir en el caso que lo necesiten o qué abogado tienen asignado a su caso judicial. No cuenta con los servicios en línea de:

- ✓ Asesoría, representación y defensa de los derechos de las personas naturales y jurídicas nacionales y extranjeras, en todas las materias del Derecho con facultades para actuar ante los tribunales de justicia, órganos administrativos en el territorio nacional y ante órganos, organismos y organizaciones extranjeras o internacionales.
- ✓ Tramitación de documentos y gestiones ante instancias administrativas.
- ✓ Consulta gratuita de tal forma se brinde dicho servicio, en un mínimo de tiempo, por un grupo de especialistas que garantizarán una respuesta segura y confiable.
- ✓ Servicios especializados a Bufetes de Servicios Especializados (BES) y a Bufete Especializado en Recurso de Casación (BEC).

Teniendo en cuenta la situación planteada, el **problema a resolver** queda expresado de la siguiente forma: ¿Cómo mejorar el canal de comunicación y de servicios a la comunidad de usuarios de la ONBC?

Teniendo como **objeto de estudio**: Los sistemas de gestión de la información en entorno Web, con el **objetivo general**: Desarrollar un Portal Web que gestione la información referente a la ONBC, de manera que contribuya al establecimiento de un canal de comunicación y servicios a la comunidad de usuarios de dichas entidades.

Enmarcándose en el **campo de acción**: Desarrollo de portales web basados en sistema de Gestión de la información legal de la ONBC.

Teniendo como referencia el problema a resolver y el objetivo general se plantea la siguiente **Idea a defender**: Con el desarrollo del nuevo Portal Web de la ONBC se contribuirá al establecimiento de un canal de comunicación y servicios a la comunidad de usuarios de dichas entidades.

Para dar cumplimiento al objetivo propuesto se han derivado un conjunto de **objetivos específicos** orientados fundamentalmente a proveer los elementos necesarios para la implementación de la solución, los cuales se listan a continuación:

- ✓ Definir el marco teórico de la investigación para conceptualizar los elementos fundamentales que serán utilizados.
- ✓ Realizar el análisis y diseño del Portal Web de la ONBC para tener una representación técnica de la solución a implementar.
- ✓ Implementar la solución para obtener un Portal Web aplicando la tecnología definida y las buenas prácticas del desarrollo de software.
- ✓ Validar la solución propuesta mediante pruebas de software y el cumplimiento de los objetivos de la investigación para comprobar la correcta funcionalidad del sistema obtenido.

Los **métodos científicos** utilizados para validar metodológicamente la investigación son:

- ✓ **Métodos teóricos:**
 - Histórico-lógico: a través del mismo se realizó un estudio de los diferentes portales web que existen en Cuba y el mundo, homólogos al sistema que se desea desarrollar. Además, permitió indagar en la historia y el comportamiento de los portales para servicios ciudadanos en la actualidad.
 - Analítico-Sintético: se utilizó en el análisis de la bibliografía donde se resumen los elementos más importantes de la gestión del aprendizaje teniendo en cuenta el objeto de la investigación.
- ✓ **Método empírico:**

- Entrevista: mediante intercambios con la ONBC se logró obtener información sobre el negocio de la organización y los requisitos que debe cumplir el portal web.

El presente trabajo de investigación está estructurado en 3 capítulos, de los cuales se brinda una breve descripción a continuación:

Capítulo 1 - Fundamentación Teórica: se describen aspectos importantes sobre el proceso de sistemas legales. Se realiza un estudio del estado del arte relativo a los sistemas informáticos que gestionan información referente a los sistemas legales para ciudadanos. Además, se efectúa un breve análisis de los lenguajes de programación, herramientas, tecnologías y metodología a emplear en el desarrollo del portal web.

Capítulo 2 - Planificación-Definición del Portal Web: se identifican y describen los requisitos funcionales y los requisitos no funcionales. Se hace un análisis del sistema a través de las historias de usuarios, además de generarse los artefactos que propone la metodología.

Capítulo 3 - Implementación y Prueba: se especifican los elementos relacionados con la implementación del sistema para dar cumplimiento a las funcionalidades propuestas en el capítulo anterior. Se detalla la estrategia de prueba a seguir para validar la calidad de la solución.

Capítulo 1: Fundamentación teórica

1.1 Introducción

En este capítulo se realiza un estudio sobre los elementos que fundamentan la base teórica conceptual del proceso de sistemas legales para ciudadanos en Cuba. Se incluye una breve descripción de la metodología, los lenguajes, las herramientas y las tecnologías que son utilizadas para todo el proceso de desarrollo del sistema.

1.2 Conceptos fundamentales

Con el objetivo de que el lector pueda tener una comprensión mayor de los temas que se relacionan en el capítulo, se describen a continuación un grupo de conceptos asociados al dominio del problema, entre los que se destacan Organización Nacional de Bufetes Colectivos, Sistemas legales, Sistema de gestión de la información entre otros.

Al abordar temas relacionados con la gestión de la información es casi imposible no vincular a la tecnología informática como factor clave facilitador de este proceso, es por ello que se define **Sistema de gestión de la información**: como conjunto de políticas y normas relacionadas entre sí que se establecen para el acceso y tratamiento de los recursos de información, incluye los registros administrativos y los archivos, el soporte tecnológico de los recursos y el público a que se destina. En su evolución el sistema puede manejar la función de inteligencia corporativa y generar productos de inteligencia (Moreiro, 1998).

En la actualidad la web se ha convertido en un eslabón importante para la comunicación, algunas instituciones que manejan información sobre procesos legales han querido informatizar sus procesos como lo es la **Organización Nacional de Bufetes Colectivos**: entidad autónoma, de interés social y carácter profesional, con patrimonio propio y personalidad jurídica amparada en el Decreto-Ley No. 81 del 8 de junio de 1984 del Consejo de Estado. Está integrada voluntariamente por juristas, que se especializan en el asesoramiento y representación de las personas naturales o jurídicas en todas las materias del Derecho (ONBC, 2013).

Siendo los **Sistemas legales**: el ordenamiento jurídico de un país o jurisdicción. El ordenamiento jurídico es el conjunto de normas jurídicas que rigen en un lugar determinado en una época concreta. En el caso de los estados democráticos, el ordenamiento jurídico está formado por la Constitución del Estado, que se rige como la norma suprema, por las leyes como normas emanadas del poder legislativo (en sus diversos tipos y clases), las manifestaciones de la potestad reglamentaria del poder ejecutivo, tales como los reglamentos y otras regulaciones, tales como los tratados, convenciones, contratos y disposiciones particulares. No se debe confundir el ordenamiento jurídico con el orden jurídico, que se traduce en el conjunto de normas que rigen una determinada área del ordenamiento jurídico (Bobbio, 1997).

Siendo un **Portal web**: un sitio web cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse buscadores, foros, documentos, aplicaciones, compra electrónica. Principalmente están dirigidos a resolver necesidades específicas de un grupo de personas o de acceso a la información y servicios de una institución pública o privada (Bobbio, 1997).

Los desarrolladores de los portales web utilizan generalmente lenguajes interpretados del lado del cliente para obtener una mayor funcionalidad, así como tecnologías para al lado del servidor para no tener que recargar la página en su totalidad, algo que molesta mucho a los usuarios constituyendo lo anterior en un **Servicio web** definido como: conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web (W3C, 2013).

1.3 Tendencias actuales

Con el objetivo de ganar experiencia en cuanto a las funcionalidades que requiere el sistema se visitaron varios sitios de Gestión de Información. A continuación se muestra un resumen de algunos de los portales visitados con sus respectivas características:

1.3.1 Portal Web del Bufete Internacional de la Habana (La Habana, 2009)

El portal del Bufete Internacional de La Habana es informacional, se debe acceder a cada sección para saber qué contiene. Tiene la opción de cambiar el idioma (inglés, francés, portugués, italiano, alemán). No cuenta con un formulario de búsqueda. Ofrece varios servicios en línea 100 % rápido y seguro y da la posibilidad de acceder a otras secciones de interés para el bufete desde la página principal.

1.3.2 Portal de la Abogacía Española (España, 2010)

El portal de la Abogacía Española es dinámico, muestra varias secciones con noticias e informaciones sobre servicios a ciudadanos. Se anuncian publicidades sobre los eventos y actividades que se programan, manteniendo a los usuarios informados. El portal permite que los usuarios se unan a las redes sociales, posee enlaces hacia otros portales de abogados. El mismo brinda a los usuarios foros de discusión para que opinen e intercambien información y les permite inscribirse para recibir los boletines mediante el correo electrónico. Los usuarios pueden cambiar el idioma del portal y realizar búsquedas simples y avanzadas. Este portal tiene como desventaja que está muy cargado de noticias, lo que provoca que el usuario no sepa o se pierda al buscar alguna información que necesite.

1.3.3 Portal Web Alpha Abogados Perú (Perú, 2010)

El portal en general es informacional, cuenta con poco número de **banners** publicitarios y anuncios, muestra algunas noticias asociadas al bufete y a los servicios que este brinda. No posee un buscador que permita a los usuarios buscar alguna información que necesite.

1.3.4 Resultados arrojados del estudio de sistemas homólogos

De los homólogos analizados en la investigación se pudo observar la fuerte tendencia que existe actualmente en los portales de establecer diferencias entre información y servicios. Por tanto, se hizo un estudio de estos dos aspectos esenciales por separado y se obtuvo el siguiente resultado:

Elementos informativos:

A continuación se detallan los elementos informativos encontrados con mayor frecuencia en los portales visitados.

- ✓ Quiénes somos (presente en 2 de los 3 sitios visitados): es la sección donde se recogen datos esenciales sobre la organización a la cual representa el portal web.
- ✓ Enlaces (presente en 2 de los 3 sitios visitados): los enlaces son dirigidos por lo general a sitios que se relacionan con el portal visitado. Estos enlaces pueden residir en el pie de página del sitio o en cualquier menú.
- ✓ Noticias (presente en los 3 sitios visitados): la publicación de noticias ha abierto una nueva puerta como medio de comunicación con carácter global, posibilitando mayores niveles de información y difusión que lo que permiten los medios tradicionales. En los portales estudiados las noticias constituyen la vía esencial para transmitir el acontecer sobre los servicios a ciudadanos.

Servicios:

- ✓ Seguimiento en redes sociales (presente en 1 de los 3 sitios visitados): son enlaces que permiten acceder a diferentes redes sociales como Facebook o Twitter.
- ✓ Comentar - Compartir (presente en 1 de los 3 sitios visitados): una de las nuevas tendencias de los portales actuales es precisamente posibilitar una mayor interacción con los usuarios, por lo cual se prestan servicios como este que posibilitan al usuario dejar su comentario sobre una noticia del sitio o enviar la misma por correo a un amigo.
- ✓ Contactos (presente en 2 de los 3 sitios visitados): por lo general se representa con un formulario que permite enviar un mensaje de contacto a los directivos de la organización aunque también puede contener solamente las direcciones (correo) de los mismos.
- ✓ Buscador (presente en 1 de los 3 sitios visitados): este servicio permite la búsqueda dentro del sitio, puede estar dada por letras, palabras o frases.
- ✓ Banner (presente en los 3 sitios visitados): estos están referidos por lo general a las noticias de mayor relevancia dentro del portal. Pueden tener diferentes dimensiones, en dependencia del nivel de importancia que se le quiera dar a la información que representan por encima del resto de los contenidos del portal.

- ✓ Servicios en línea (presente en los 2 de los 3 sitios visitados): estos están referidos a poder hacer uso de este servicio vía web sin tener que llegar al bufete a pedir dicho servicio.

A partir de los resultados arrojados en el análisis de sistemas homólogos realizado se decide desarrollar el portal web para la ONBC teniendo en cuenta algunas de las características más frecuentes presentes en dichos portales, además de los requisitos funcionales que sean incorporados a solicitud del cliente y de acuerdo a las necesidades de la organización.

1.4 Metodología de Desarrollo de Software

Una metodología de desarrollo de software se puede definir como un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda en la construcción de un software (Pressman, 2001).

Todo desarrollo de software es riesgoso y difícil de controlar, de ahí la importancia de contar con una metodología que indique paso a paso las actividades que deben desarrollarse, las personas involucradas en la realización de las mismas, así como el papel que desempeñan estas en aras de lograr un proceso de software exitoso.

Metodologías Ligeras/Ágiles: metodologías orientadas a la interacción con el cliente y el desarrollo incremental del software, mostrando versiones parcialmente funcionales del software al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando, requiere de pequeños grupos de trabajo y son apropiadas para entornos volátiles. Son a diferencia de las tradicionales, más adaptables a los cambios (Prezi, 2013).

Dentro de las metodologías ágiles que se caracterizan por dividir el desarrollo de software en pequeñas iteraciones y ser el cliente parte del equipo de desarrollo, se encuentran SCRUM, XP y SXP.

1.4.1 Programación extrema (XP)

Es una metodología ágil que se basa fundamentalmente en promover el trabajo en equipo. El principal objetivo que persigue, es lograr que el producto se termine en el tiempo requerido por los clientes,

obteniendo la plena satisfacción del mismo. Es una metodología en la cual existe comunicación entre el cliente y los desarrolladores, pues el cliente forma parte del equipo de trabajo y este tiene el derecho a decidir qué es lo que se debe implementar, hacer cambios en los requisitos y exigir la entrega de una parte del trabajo cada cierto tiempo. Esto demuestra la retroalimentación existente entre ambas partes, la correcta comunicación y forma de trabajo. Para la especificación de requisitos se utilizan las historias de usuarios, las cuales pueden ser totalmente flexibles a cambios o reemplazos. La programación puede llevarse a cabo por parejas de programadores, disminuyendo la cantidad de errores y también se realizan pruebas unitarias. Las cuatro fases que se definen en esta son: planificación, diseño, desarrollo y prueba (Kniberg, 2006).

1.4.2 SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados de forma rápida, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se requiera capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de productos.

1.4.3 SXP

Es una metodología que consiste en la integración de los aspectos más eficientes de las otras dos metodologías ágiles, la metodología SCRUM y la metodología XP. La unión de estas metodologías está dividida en cuatro fases principales: Planificación-Definición, Desarrollo, Entrega, Mantenimiento. Cada

fase está compuesta por una serie de actividades donde se generan los artefactos que quedan incluidos en el expediente de proyecto (Leyva Samada, 2009).

SXP, es un híbrido cubano de metodologías ágiles, que ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo. Consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Basada completamente en los valores y principios de las metodologías ágiles expuestos en el Manifiesto Ágil. Como método de estimación utiliza la opinión de expertos y constan con métricas o indicadores para lograr una eficiente calidad (Leyva Samada, 2009).

Fundamentación de la metodología seleccionada

SCRUM puede ser visto como una interfaz entre el equipo y los clientes. SCRUM deja un agujero en el medio, ejemplo: como el equipo debe hacer su trabajo diario, en el cual XP encaja muy bien. Una vez que SCRUM es puesto en práctica el equipo es auto administrado y puede escoger como hacer el trabajo. El desarrollo de software ágil es todo sobre círculos de retroalimentación. SCRUM y XP se complementan el uno al otro de manera muy positiva. Las dos metodologías dan un círculo de retroalimentación de unos pocos segundos, donde los defectos son detectados y corregidos en segundos. Es muy práctico implantar SCRUM como metodología de gestión, y luego ir introduciendo las prácticas de ingeniería de XP, para lograr mayor productividad, mayores resultados y satisfacciones por parte del cliente y del equipo de trabajo (Romero, 2008).

Teniendo en cuenta los elementos analizados anteriormente referidos a las características de XP, SCRUM y SXP, se selecciona como metodología de desarrollo de software SXP, la cual consta de 4 fases (Ver Anexo 2) (Romero, 2008).

Esta metodología ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas a realizar, de forma que los jefes pueden ver día a día cómo progresa el trabajo (Leyva Samada, 2009).

1.5. Herramientas para el desarrollo de la solución

Hay muchas herramientas disponibles para ayudar a que los proyectos de desarrollo web sean más rápidos y productivos. Hoy en día es fácil encontrar una gran variedad de herramientas y utilidades que puede aumentar la velocidad de desarrollo, depuración y reducir el tiempo de prueba, y mejorar la calidad de producción de sitios web. Las herramientas que se describen a continuación son una variedad de servicios públicos, optimizadores, verificación, depuración y herramientas destinadas a ayudar a los desarrolladores a crear sitios web de manera más eficiente.

1.5.1. Sistema gestor de contenidos

Un sistema gestor de contenidos (CMS) es un programa que permite crear una estructura de soporte para la creación y administración de contenidos de páginas web. Es una 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.

Los CMS aportan herramientas para definir la estructura, el formato de las páginas y el aspecto visual, el sistema se encarga de gestionar muchos otros aspectos como son los menús de navegación. Permite manejar de manera independiente el contenido y el diseño o presentación.

Es un programa que sirve de plataforma para crear, editar, eliminar contenido por parte de los administradores, utilizando un ambiente gráfico para la administración, muchos de ellos necesitan una base de datos donde se almacena la información y son fáciles de instalar y configurar. El contenido será visto por los usuarios finales que visiten el CMS, y puede cambiarse el diseño sin alterar el contenido, finalmente, un software que ofrece utilidades específicas (Rojas, 2010). Dentro de los CMS más usados se encuentran:

1.5.1.1. Joomla 1.7.3

Joomla es un sistema de administración de contenidos de código abierto construido con PHP bajo una licencia General Public License (GPL). Dentro de las características principales de Joomla están (Dhar, 2010):

- ✓ Software libre (Licencias GNU¹/GPL), ampliable al disponer el código fuente.
- ✓ Completa y fácil administración por la web.
- ✓ Creación y administración rápida de una comunidad on-line².
- ✓ Creación de la web por inserción de módulos y componentes independientes.
- ✓ Creación y actualización dinámica de secciones, subsecciones y contenidos (públicos y privados).
- ✓ Creación de perfiles y privilegios con niveles jerárquicos para diferentes niveles de usuarios (súper administrador, administrador, gerente, editor y usuario registrado).
- ✓ Plantillas para transformar el diseño visual de la Web de forma automática en pocos minutos.
- ✓ Estadísticas de acceso a los contenidos.

1.5.1.2. Drupal 7.24

Drupal es un CMS de código libre y abierto, escrito en PHP y se distribuye bajo la GNU/GPL. Se deriva de un proyecto del estudiante universitario holandés Dries Buytaert.

Drupal se volvió en un proyecto de código abierto en el año 2001, la comunidad ha adoptado de manera fácil su concepto expandiéndolo hasta crearlo en uno de los CMS más poderosos en la web. Los individuos, equipos y las comunidades de usuarios de Drupal pueden con facilidad publicar, administrar y

¹GNU: GNU es un acrónimo recursivo que significa GNU No es Unix.

²*On – line*: en línea, conectados

organizar el contenido desde **blogs** personales, grandes empresas y centros gubernamentales mediante los sitios web (Douglass, 2010).

La liberación estándar de Drupal, conocido como el núcleo de Drupal, contiene funciones básicas que pueden ser utilizadas para crear un sitio web, un **blog** usuarios, un foro de Internet o de una comunidad con contenidos generados por usuarios. Dentro de las características que se encuentran en el núcleo de Drupal, se incluye la capacidad para crear y administrar contenido, usuarios, menús, foros y encuestas, gestionándolo a través de un navegador web mediante una interfaz de administración. Drupal ha sido diseñado para ser mejorado con nuevas características y el comportamiento personalizado mediante la descarga de módulos adicionales. Existe un gran número de módulos adicionales disponibles (conocidos como contribuidos) que amplían las funcionalidades básicas de este CMS, estas cubren un amplio espectro de capacidades, incluido el comercio electrónico, Web 2.0, la integración con aplicaciones de terceros y multimedia (Tomlinson, 2010).

Fundamentación del CMS seleccionado

Se decide seleccionar el CMS Drupal porque distingue los contenidos en distintas categorías: páginas, entradas de **blog**, noticias, artículos, encuestas, foros; teniendo cada uno de estos sus propias características, lo que le aporta una estructura potente y más organizada. Además, está apoyado por una enorme comunidad de usuarios bastante activa que tiene implementada una gran cantidad de plantillas y módulos para el desarrollo de portales basados en este CMS, dentro los que se destacan un gran número enfocados a la Web 2.0 y las comunidades multiusuarios. Las nuevas versiones del CMS están integradas plenamente con las funcionalidades de la Web 2.0 convirtiendo a Drupal en un CMS extremadamente fácil de usar por cualquier persona con conocimientos básicos de informática.

1.6. Lenguaje de Modelado UML 2.0

El lenguaje de modelado empleado por la metodología seleccionada SXP, es el Lenguaje Unificado de Modelado (UML por sus siglas en inglés Unified Modeling Language), el cual es un lenguaje de modelado visual que se emplea para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se emplea para

entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas (Rumbaugh, 2007).

Proporciona un conjunto de elementos de modelado, anotaciones, relaciones y normas que pueden aplicarse a una actividad de desarrollo de software. Sin embargo, UML también puede ser empleado para modelar otros dominios, tales como el modelado de sistemas y el modelado de negocio (Leffingwell, 2003).

Se puede aplicar de diversas formas para dar soporte a una metodología de desarrollo de software pero, en sí mismo, no especifica qué metodología o proceso emplear.

1.7. 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 sintáctica y semánticamente correctas (Rodríguez Sala, 2003).

1.7.1. PHP 5.3.8

Acrónimo de Hypertext Preprocessor es un lenguaje del lado del servidor, multiplataforma, de código abierto y gratuito, fue creado especialmente para el desarrollo de páginas web dinámicas. Con facilidad se inserta dentro del código HTML, permitiéndole el desarrollo de un gran número de funcionalidades. Puede soportar numerosas visitas con escaso mantenimiento y su integración con múltiples bases de datos como MySQL, Oracle o PostgreSQL es sencilla. Cuenta con la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente. La programación en PHP es segura y confiable. Además, aporta grandes facilidades a partir de una serie de ventajas con las que cuenta como:

- ✓ Una biblioteca de funciones sumamente amplia e incluida.
- ✓ Amplia documentación en su página oficial, en la que sobresale que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.

- ✓ Posee una amplia capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad.
- ✓ Es de código abierto, además no tiene que pagar para obtener las actualizaciones.
- ✓ No requiere definición de tipos de variables (Sócrates, 2000).

1.7.2. HTML 5

HTML es un lenguaje artificial que los ordenadores son capaces de interpretar y diseñado para que los programadores redacten instrucciones que los navegadores ejecutan para originar la página web. Es decir, HTML es un lenguaje de programación o un idioma que la máquina entiende y procesa para dar una respuesta (Vértice, 2009).

Sirve para estructurar documentos (títulos, párrafos, listas y otros), pero no describe la apariencia o el diseño de estos, sino que ofrece las herramientas necesarias para dar formato, según la capacidad del servidor web en el que se almacenan las páginas web y la capacidad del navegador, dígase resolución de la pantalla, fuentes de texto instaladas y otros. Dentro de sus ventajas sobresale la compatibilidad y facilidad que plantea su aprendizaje debido al reducido número de etiquetas en las que se apoya (Vértice, 2009).

1.7.3. CSS (Hojas de Estilos en Cascada) 3

Las Hojas de Estilo en Cascada (**Cascading Style Sheets** conocido como CSS, por sus siglas en inglés) son un lenguaje formal usado para definir la presentación de un documento estructurado, escrito en HTML o XML (Lenguaje de Marcas Extensible) permite dar formato y estilo de presentación. Estandarizado por la organización W3C, es la simplificación del mantenimiento de las páginas web generadas en HTML que suponen una mejora de dicho lenguaje en cuanto a presentación, mantenimiento y estandarización del contenido (Sierra, 2012).

Las CSS permiten a los desarrolladores web llevar a cabo el control sobre el estilo y formato de múltiples páginas web, al mismo tiempo cualquier cambio en el estilo o formato de un elemento en las CSS, beneficiará a todas las páginas vinculadas a dichas CSS en las que aparezca ese elemento.

- ✓ Los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad.
- ✓ Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario.
- ✓ El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea) (López 2010).

1.7.4. JavaScript 1.8.5

JavaScript es uno de los lenguajes de **script** u orientado a documento (González Boticario, 2001) más usados en Internet, creado por la empresa Netscape, para añadir interactividad a las páginas web. Es compacto y basado en objetos, diseñado para el desarrollo de aplicaciones cliente-servidor a través de Internet (Sánchez Maza, 2012).

Además, es un lenguaje interpretado e independiente de la plataforma que permite dar respuestas a eventos iniciados por el usuario tales como la entrada de datos en un formulario o la elección de algún enlace del documento HTML. La principal ventaja de JavaScript es que eso sucede sin ningún tipo de transmisión de datos a través de internet, de tal forma que cuando un usuario escribe algo en un formulario, no es necesario que sea enviado al servidor, verificado y devuelto (González Boticario, 2001).

1.8. Herramienta para modelado de prototipos web

Para el modelado de los prototipos web se decide emplear Axure RP Pro 5.5 pues, a diferencia de la herramienta CASE Visual Paradigm permite crear prototipos amigables, con mejor calidad visual, brinda además un conjunto de librerías muy atractivas y la posibilidad de crearlas nuevas para satisfacer las necesidades del modelado. Permite generar los diseños de prototipos a HTML permitiendo la interacción de cualquier miembro del equipo de desarrollo o las partes interesadas con estos sin necesidad de instalar Axure; de esta forma se puede comprobar si el sistema de navegación de la aplicación web es el deseado y además el cliente podrá tener una idea clara y precisa de cómo será la aplicación en un futuro. Otra de

las características de Axure que influyeron en su selección es la facilidad de crear elementos de diseño reutilizables como encabezados, pies de página y plantillas.

1.9. Herramienta CASE

Las herramientas CASE comprenden un conjunto de programas de diferentes tipos empleadas para ayudar a las actividades del proceso del software como el análisis de requerimientos, el modelado de sistemas, la depuración y las pruebas.

“Permiten incrementar la productividad, comunicarse de manera más eficiente con los usuarios, integrar las actividades y proporcionar continuidad de una fase a la siguiente durante todo el ciclo de vida del desarrollo de sistemas e integrar el trabajo que desempeñan en el sistema”, reduciendo de esta forma el costo en términos de tiempo y dinero del proceso de desarrollo del software.

1.9.1. Herramienta CASE para UML

Existen diversas herramientas CASE que soportan el lenguaje de modelado UML, dentro de las que se destacan Visual Paradigm for UML (VP-UML) y Rational Modeler por ser bastante completas, disponer de varios diagramas como los de clases, de objetos, de casos de uso del negocio y diagramas de paquetes, por la facilidad de uso para la modificación y creación de estos y por generar código a partir de dichos diagramas. Sin embargo, Rational Modeler es un software propietario, mientras que Visual Paradigm es gratuito aunque se encuentra bajo una licencia que no permite su modificación o venta, ajustándose esta licencia a las políticas de soberanía tecnológica trazadas por el país.

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 sus características, se selecciona esta herramienta en su versión 8.0, para realizar el diseño UML del sistema para la ONBC.

Visual Paradigm 8.0

Es una herramienta CASE multiplataforma muy potente y fácil de utilizar que hace uso del lenguaje unificado de modelado (UML). Soporta las últimas versiones de UML. Mediante Visual Paradigm se puede realizar el modelado, la captura de requisitos, diseño de base de datos, modelado de procesos de

negocio. Ofrece al analista y desarrollador de software un ambiente para analizar, diseñar y mantener aplicaciones de software en una manera disciplinada. Además, aumenta la producción automática de código, bases de datos y generación de informes. Visual Paradigm para UML es apoyado por un conjunto de idiomas, tanto en la generación del código como en la ingeniería inversa, por mencionar algunos ejemplos los cuales tienen la capacidad de soporte se pueden mencionar: Java, C++, CORBA IDL, PHP, Ada y Python. Para maximizar la interoperabilidad de los productos de Visual Paradigm con otras aplicaciones se introdujo la importación y exportación de modelos de proyecto a un formato XML. Los usuarios y proveedores de tecnología pueden integrar Visual Paradigm en cada uno de sus modelos para utilizarlos en sus soluciones con un mínimo esfuerzo.

Esta herramienta proporciona una plataforma ampliable para que los desarrolladores puedan agregar funciones al mismo, ellos pueden hacer referencia a los plugin de guiar el desarrollo, a construir sus propios plugin para leer, actualizar, recuperar y eliminar los diagramas y los elementos del modelo.

1.10. Sistema Gestor de Base de Datos

Un Sistema Gestor de Bases de Datos (SGBD) es un conjunto de programas que administran y gestionan la información que se encuentra en una base de datos. Permiten definir datos, mantener la integridad de estos dentro de la base de datos, controlar la seguridad y privacidad de los mismos.

1.10.1. MySQL 5.5.16

MySQL ha pasado de ser una pequeña base de datos a una completa herramienta. Su rápido desarrollo se debe en gran medida a la contribución de muchas personas al proyecto, así como la dedicación del equipo de MySQL. Lo que en un tiempo se consideró como un sencillo juguete para uso en sitios web, se ha convertido en la actualidad en una solución viable y de misión crítica para la administración de datos. Es un Sistema de Administración de Bases de Datos Relacional (RDBMS). Se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y de distribuirlos para cubrir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos. Este gestor compite con sistemas RDBMS propietarios como: Oracle, SQL Server. Incluye todos los elementos necesarios para instalar el programa, preparar diferentes

niveles de acceso de usuario, administrar el sistema y proteger los datos. Puede desarrollar sus propias aplicaciones de bases de datos en la mayor parte de lenguajes de programación utilizados en la actualidad y ejecutarlos en casi todos los sistemas operativos, utiliza el lenguaje de consulta estructurado (SQL) es rápido y una solución accesible para administrar correctamente los datos de una empresa. Son muchas las razones para escoger a MySQL como una solución de misión crítica para la administración de datos:

- ✓ **Funcionalidad:** dispone de muchas de las funciones que exigen los desarrolladores profesionales, como compatibilidad completa con ACID (por sus siglas en inglés **Atomicity, Consistency, Isolation and Durability**), volcados en línea, duplicación e integración con la mayor parte de los entornos de programación.
- ✓ **Portabilidad:** se ejecuta en la inmensa mayoría de sistemas operativos y la mayor parte de los casos, los datos se pueden transferir de un sistema a otro sin dificultad.
- ✓ **Facilidad de uso:** resulta fácil de utilizar y de administrar. Las herramientas son potentes y flexibles, sin sacrificar su capacidad de uso.

1.10.2. PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde 1977. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. PostgreSQL proporciona un gran número de características que normalmente solo se encontraban en base de datos comercial como Oracle. A continuación se muestra brevemente una lista de características de este gestor:

- ✓ **Soporte SQL comprensivo:** soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (**joins**) SQL92.
- ✓ **Integridad referencial:** soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.

- ✓ DBMS Objeto-Relacional: aproxima los datos a un modelo objeto-relacional y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-version, soporte multi-usuario, optimización de consultas, herencia y arreglos.
- ✓ Altamente extensible: soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- ✓ Cliente/Servidor: usa una arquitectura proceso-por-usuario cliente/servidor.

Fundamentación del SGBD seleccionado

El sistema gestor de base de datos seleccionado es MySQL debido a que es muy rápido, confiable, robusto y fácil de usar tanto para volúmenes de datos grandes como pequeños. Es muy utilizado para el desarrollo de aplicaciones y páginas web. Es software libre, por tanto, se puede obtener y trabajar sin preocupaciones, utiliza la licencia GPL para definir qué se puede o no hacer. Se debe mencionar además que es de fácil integración con el lenguaje de programación que se utilizará en la solución del problema.

1.11. Servidor Web Apache 2.2.21

Apache es un programa de servidor web de código libre, robusto, cuya implementación se ha realizado y se sigue realizando de forma colaborativa, con prestaciones, características y funcionalidades equivalentes a las de cualquier servidor comercial. Algunas de sus características:

- ✓ Es una tecnología gratuita de código fuente abierta.
- ✓ Trabaja con PHP y otros lenguajes de script. Teniendo todo el soporte que se necesita para desarrollar páginas dinámicas.
- ✓ Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.

1.12. Entorno de Desarrollo Integrado

Un Entorno de Desarrollo Integrado (IDE, **Integrated Development Environment**) es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Puede dedicarse en exclusiva a un solo lenguaje de programación o puede utilizarse para varios.

Un IDE es un programa informático compuesto por un conjunto de herramientas de programación que les permite a los programadores escribir, compilar, depurar y ejecutar programas (Netbeans, 2012).

1.12.1. Netbeans 7.3

Existen diversos IDE, cada uno de ellos brinda la posibilidad de trabajar con uno o más lenguajes de programación; dentro de ellos sobresalen NetBeans y Eclipse. 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, depuración, etc. Sin embargo, a pesar de que NetBeans realiza mejor el autocompletado, Eclipse posee mayor rendimiento pues inicia y reacciona más rápido y hace menor uso de memoria (Software Talk, 2012).

No obstante, NetBeans es un producto libre y gratuito sin restricciones de uso, teniendo el código fuente disponible para su reutilización bajo las licencias CDDL (Licencia Común de Desarrollo y Distribución) y la GPL (**General Public License**), presenta una interfaz muy amigable e intuitiva y tiene todas las herramientas para crear aplicaciones profesionales ya sean de escritorio, empresariales, web, móviles y aplicaciones SOA (Arquitectura Orientada a Servicios), no solo en Java sino también en C/C++ y Ruby.

Debido a las razones antes expuestas y teniendo en cuenta que la herramienta NetBeans es libre, ajustándose a las políticas de soberanía tecnológica trazadas por el país, se selecciona para el desarrollo del sistema para la ONBC en su versión 7.3.

1.13. Conclusiones parciales

En este capítulo se realizó un análisis de los conceptos fundamentales relacionados con los sistemas legales, lo que permitió dominar terminologías importantes involucradas en la solución del problema propuesto. Se estudiaron los tipos de sistemas de información existentes y las tendencias actuales de estos, lo que facilitó una serie de requisitos funcionales que pudieran incluirse en la propuesta de solución. Además, se analizaron las posibles metodologías, lenguajes, herramientas y tecnologías a utilizar para el desarrollo del sistema, lo que garantizará la construcción de un sistema que resuelva la problemática planteada.

Capítulo 2: Planificación-Definición del Portal Web

2.1. Introducción

En el presente capítulo se abordarán temas relacionados con el funcionamiento del sistema, siendo guiados por las dos primeras fases en las que se divide la metodología SXP: Planificación-Definición. En estas fases se abordarán temas relacionados con las necesidades del software y conceptos introducidos por la metodología como son las Historias de Usuario. Se definirán las funcionalidades y se generarán los artefactos propuestos por la metodología seleccionada.

2.2. Descripción del negocio

El sitio web que representa a la ONBC no posee servicios en línea, por lo que los usuarios tienen que dirigirse hasta la entidad para solicitar el servicio. Estos deben ser automatizados como es el caso del proceso de certificación de documentos ya sea de nacimiento, matrimonio, divorcio y defunción. Este proceso está dirigido a todos los usuarios que deseen certificar cualquier documento de los antes mencionados, el mismo deberá permitir a estos contactar con los especialistas encargados de este tema, a través de una solicitud de certificación. Otro de los puntos a tener en cuenta para automatizar es la consulta gratuita que realizan los usuarios sobre temas legales. Además de realizar el servicio de BEC, BES y el Catálogo en línea. También se automatizarán dos directorios uno para los bufetes de todos el país y el otro con los abogados que atienden casos relacionados con la vivienda y que pertenecen a un determinado bufete.

2.3. Flujo actual de procesos

Se decide describir el funcionamiento del sistema a desarrollar mediante una serie de conceptos, entidades y sus relaciones, agrupándose en un Modelo de dominio para facilitar un mejor entendimiento.

Clases del Modelo de dominio

- ✓ **Portal:** aplicación web encargada de gestionar los contenidos.
- ✓ **Usuarios:** personal que interactúa con el sistema o que está registrada en ella.

- ✓ **Administrador:** una o varias personas que se encargan de trabajar en la actualización y soporte del sistema.
- ✓ **Registrador:** encargado de publicar las respuestas a las inquietudes de los usuarios sin relevar su identidad y registrar recurso de casación.
- ✓ **Invitado:** personal que se encuentra autenticado en el sistema y tiene acceso a los servicios brindados en el portal.
- ✓ **Contenidos:** todas aquellas noticias, avisos, informaciones, artículos, etc. con formatos que van desde un simple texto plano a animaciones, videos o fotografías, concebidos para informar, formar o entretener.
- ✓ **Noticias:** información que se brinda en el sistema sobre cualquier hecho ocurrido o próximo a ocurrir, en el mundo o en Cuba.
- ✓ **Información:** cualquier elemento dentro del sistema que brinde al usuario algún conocimiento acerca de un tema.
- ✓ **Servicios:** son funciones que se le brindan al usuario con el objetivo de responder a una o más necesidades del mismo.

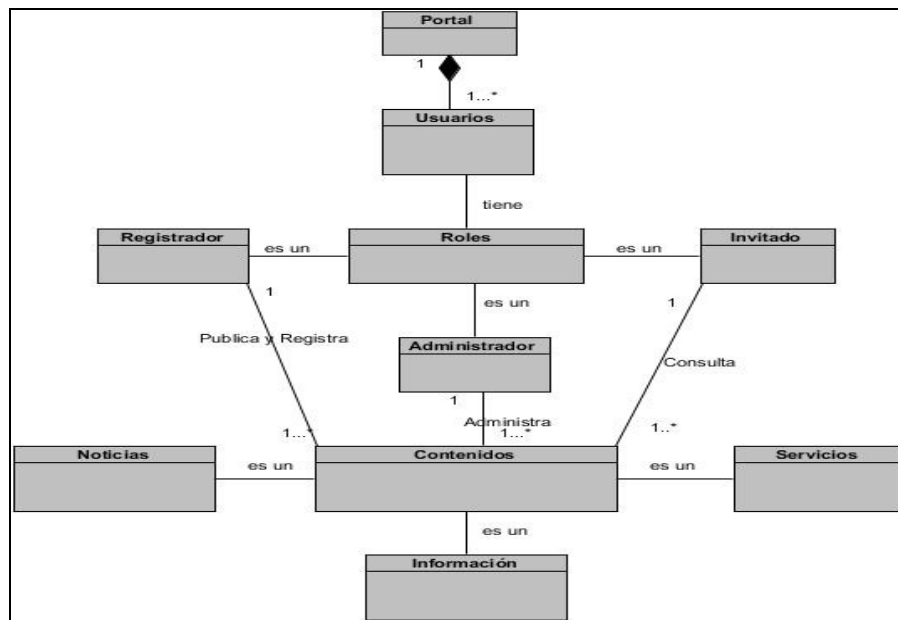


Imagen 1: Modelo de dominio

2.4. Propuesta de Solución

Una vez analizadas las dificultades que posee el portal web de la ONBC, en cuanto a apariencia, gestión de servicios, actividades y contenido, además de tener en cuenta el resultado del análisis de sistemas existentes y la entrevista con el cliente se propone modelar un sistema con las siguientes características.

El sistema contará con una interfaz principal y será capaz de mostrar cada proceso que caracteriza a la organización como es el caso de gestionar el flujo de noticias referentes a esta. Deberá ser capaz de brindar servicios interactivos e informativos así como mostrar todo el flujo de información de la ONBC en cuanto a estructura y documentación oficial, además de mostrar enlaces hacia otras páginas que sea de interés para la organización. Los usuarios podrán realizar la acción que estos deseen en el sistema en dependencia del nivel de autorización o acceso que posean, por lo que solo podrán hacer las modificaciones que le sean permitidas teniendo en cuenta la responsabilidad que estos desempeñen.

2.5. Personal relacionado con el sistema

Las personas relacionadas con el sistema serán aquellas que interactúen de una forma u otra con el mismo, obteniendo algún beneficio de los diferentes procesos que se ejecuten. Su implementación será a través de módulos donde se establecerán un grupo de roles para asignar los diferentes permisos para su acceso. Tanto los roles como los permisos determinan el nivel de acceso a las funciones del sistema de cualquier usuario que interactúe con el portal. El sistema propuesto dispondrá de los siguientes roles de usuario.

Tabla 1: Personas relacionadas con el sistema

Personas relacionadas con el sistema	Descripción
Registrador	Es la persona que se va a encargar de registrar todo tipo de documento que se realice en el sistema.
Administrador	Son aquellos usuarios que tienen permiso para administrar distintas funcionalidades del sistema.

	Gestionan los contenidos, usuarios y todos los servicios que se brindan.
Invitado	Persona que se encuentra autenticado en el sistema y tiene acceso a los servicios brindados en el portal.

2.6. Requisitos

Realizar un buen levantamiento de requisitos es fundamental para el éxito de un proyecto porque estos constituyen la voz del cliente en el proceso de desarrollo del software y por tanto la garantía de que el producto final satisfaga las necesidades del mismo. Por su importancia en dicho proceso deben ser viables, medibles, comprobables, relacionados con las necesidades u oportunidades del negocio identificadas y definidas con eficiencia para lograr la correcta implementación del sistema. Existen diferentes técnicas para lograr las especificaciones mencionadas anteriormente en este caso se pusieron en práctica las siguientes:

Entrevistas

Las entrevistas se emplearon con el fin de reunir de forma más específica todo tipo de información relacionada con la Organización, así como las actividades y el flujo de contenido que los futuros usuarios finales consideran se deben mostrar.

Sistemas existentes

Esta técnica se utilizó para analizar distintos portales ya desarrollados que son homólogos del que se desea desarrollar. Se analizaron las interfaces de usuario, observando el tipo de información manejada, la estructura de navegación y el contenido que se muestra en dichos sistemas para establecer estándares a seguir y propiciar el surgimiento de nuevas ideas.

Tormenta de ideas

Se utilizó con el objetivo de generar la máxima cantidad posible de requisitos para el sistema. Se llevó a cabo en reuniones, entrevistas y encuentros con el cliente y especialistas en el tema.

2.6.1. Lista de Reserva del producto

Los requisitos funcionales no son más que capacidades o condiciones a cumplir por el sistema. La realización de la Lista de Reserva del Producto (LRP) es muy importante para el proceso de desarrollo del software pues de esta dependen las diferentes funcionalidades con que contará el software así como la calidad del mismo atendiendo al grado de satisfacción del cliente con el producto final. A continuación se muestran los requisitos funcionales que debe cumplir el sistema (Ver Anexo 3):

RF01 Autenticar usuario

RF02 Cambiar contraseña

RF03 Adicionar información

RF04 Editar información

RF05 Eliminar información

RF06 Adicionar usuario

RF07 Editar usuario

RF08 Eliminar usuario

RF09 Adicionar enlace

RF10 Eliminar enlace

RF11 Solicitar certificación de nacimiento

RF12 Solicitar certificación de defunción

RF13 Solicitar certificación de matrimonio

RF14 Solicitar certificación de divorcio

RF15 Solicitar consulta gratuita

RF16 Notificar las respuestas emitidas por los especialistas

RF17 Publicar las preguntas de los ciudadanos y las respuestas de los especialistas

RF18 Buscar Información

RF19 Mostrar cursos

RF20 Solicitar curso

RF21 Emitir respuesta de aceptación o no del curso

RF22 Adicionar Catálogo

RF23 Mostrar Catálogo

RF24 Editar Catálogo

RF25 Eliminar registro de la cesta

RF26 Mostrar cuantía de obsolescencia de las noticias creadas

RF27 Mostrar contador de tiempo en el formato (dd/mm/aa)

RF28 Mostrar Información sobre los bufetes que radican en cada provincia y municipios

RF29 Eliminar Información sobre los bufetes que radican en cada provincia y municipios

2.6.2. Requisitos no funcionales

Los requisitos no funcionales (RnF) se refieren a las características o cualidades que debe tener el producto para hacerlo atractivo, usable, rápido y confiable. Para el desarrollo del sistema de la ONBC se definieron 24 RnF, los que se clasificaron en RnF de usabilidad, confiabilidad, soporte, estándares aplicables, seguridad, portabilidad, apariencia o interfaz externa y requisitos legales (Ver Anexo 4).

2.7. Historias de Usuario

La metodología SXP utiliza la técnica de las Historias de Usuario (HU) para sustituir a los documentos de especificación funcional y a los casos de uso. Estas HU son escritas por el cliente en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. El tratamiento de las HU es muy dinámico y flexible, permite que en cualquier momento se puedan romper, reemplazar por otras más específicas o generales, añadirse nuevas o ser modificadas. Las HU deben tener el detalle mínimo para permitir que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Para ser implementadas las HU, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia. El tiempo de desarrollo ideal para una HU es entre unas 10 horas y un par de semanas. A continuación se muestra una de las HU que representa una de las funcionalidades críticas que fueron implementadas para la propuesta de solución.

Tabla 2: Descripción de la Historia de usuario

Historia de Usuario	
No: número consecutivo a partir del 1.	Nombre: identifica la HU.
Usuario: ¿Quién ejecuta la HU?	
Prioridad de Negocio: define la relevancia e impacto de la historia de usuario para el negocio de acuerdo con las necesidades del usuario.	Puntos Estimados: permite estimar la duración de la implementación, representando con 1, una semana de trabajo.
Nivel de Complejidad: define la dificultad técnica que supone desarrollar la historia de usuario desde el punto de vista del programador.	Iteración Asignada: precisa la iteración en la que será desarrollada la HU.
Descripción: explica en qué consiste la HU, teniendo en cuenta las acciones realizadas por el usuario y la respuesta brindada por el sistema.	
Observaciones: brinda información extra que se estime agregar para hacer más comprensible la HU. Por	

ejemplo: conceptos, post-condiciones, relación con otros requisitos, entre otras.

Tabla 3: Historia de usuario "Catálogo en línea"

Historia de Usuario	
No: 09	Nombre: Catálogo en línea
Usuario: Administrador	
Prioridad de Negocio: Alta	Puntos Estimados: 3
Nivel de Complejidad: Alta	Iteración Asignada: 1
Descripción: Se podrá adicionar, mostrar y actualizar un catálogo, además se podrán eliminar registros existentes en la cesta de registros.	
Observaciones: El usuario tiene que estar autenticado en el sistema y poseer los privilegios para realizar la acción indicada. La HU hace referencia a los requisitos funcionales: RF26 Adicionar Catálogo RF27 Mostrar Catálogo RF28 Actualizar Catálogo RF29 Eliminar registro de la cesta	

2.8. Estimación de esfuerzos por HU

De acuerdo a la prioridad asignada por el cliente a cada HU y teniendo en cuenta la complejidad y riesgo determinado por el programador, se realiza la estimación de cada una de las HU identificadas. La unidad de estimación es el punto, un punto equivale a una semana ideal de programación. Si la estimación supera los 3 puntos la HU deberá ser dividida hasta que pueda ser desarrollada en un tiempo factible. En caso de que el esfuerzo sea menor de 1 punto la HU será combinada con otra.

Tabla 4: Estimación de esfuerzo por HU

NO.	HU	Estimación (días)	Puntos de estimación
-----	----	-------------------	----------------------

1	Autenticar usuario	2	3
2	Gestionar información	2	
3	Gestionar usuario	2	
4	Gestionar enlace a otros sitios	1	
5	Servicio de solicitud de certificación	7	
6	Servicio de consulta gratuita	3	
7	Buscar Información	1	
8	Gestionar información de los Cursos	3	
9	Catálogo en línea	21	3
10	Mostrar cuantía de obsolescencia de las noticias creadas	1	3
11	Servicio de Información	6	
12	Gestionar noticia	2	
13	Servicio especializado en recursos de casación	3	
14	Servicio de bufete especializado	2	
15	Servicio de asesoría, representación y defensa	2	
16	Políticas de Seguridad	1	
17	Términos de Uso	1	
18	Gestionar Encuesta	3	

2.9. Plan de Iteraciones

En el plan de iteraciones se especifican exactamente cuáles son las HU definidas por el cliente que serán implementadas en cada iteración del sistema. Cada HU se traduce en tareas específicas de programación y se definen las pruebas de aceptación las cuales son realizadas al final de cada ciclo. Al terminar cada iteración la aplicación tendrá implementadas funcionalidades para dar cumplimiento a los objetivos propuestos. Además, se le mostrará al cliente la primera versión de prueba con el objetivo de obtener retroalimentación para el grupo de trabajo.

Iteración 1: en la primera iteración se entregarán las funcionalidades que tienen prioridad alta para el cliente las cuales son:

HU03 Gestionar usuario

HU05 Servicio de solicitud de certificación

HU06 Servicio de consulta gratuita

HU09 Catálogo en línea

HU12 Gestionar noticia

HU13 Servicio especializado en recursos de casación

HU14 Servicio de bufete especializado

HU15 Servicio de asesoría, representación y defensa

Iteración 2: en esta iteración se entregarán las funcionalidades que tienen prioridad media para el cliente las cuales son:

HU02 Gestionar información

HU04 Gestionar enlaces a otros sitios

HU08 Gestionar información de los Cursos

HU11 Servicio de Información

Iteración 3: en esta iteración se entregarán las funcionalidades que tienen prioridad baja para el cliente pero no menos importante que las demás, las cuales son:

HU01 Autenticar usuario

HU07 Buscar Información

HU10 Mostrar cuantía de obsolescencia de las noticias creadas

HU16 Políticas de Seguridad

HU17 Términos de Uso

HU18 Gestionar Encuesta

2.10. Plan de duración de las iteraciones

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

Tabla 5: Plan de duración de las iteraciones.

Iteración	Orden de las HU	Duración Total(semanas)
1	HU03 Gestionar usuario HU05 Servicio de solicitud de certificación HU06 Servicio de consulta gratuita HU09 Catálogo en línea HU12 Gestionar noticia HU13 Servicio especializado en recursos de casación HU14 Servicio de bufete especializado HU15 Servicio de asesoría, representación y defensa	6
2	HU02 Gestionar información HU04 Gestionar enlaces a otros sitios HU08 Gestionar información de	1.7

	los cursos HU11 Servicio de Información	
3	HU01 Autenticar usuario HU07 Buscar Información HU10 Mostrar cuantía de obsolescencia de las noticias creadas HU16 Políticas de Seguridad HU17 Términos de Uso HU18 Gestionar Encuesta	1.2

2.11. Arquitectura del sistema

Jerrold Graschow define la arquitectura de un sistema como el marco conceptual completo que describe su forma y estructura (sus componentes y la manera en que se integran). A continuación se hace referencia a la arquitectura del CMS Drupal y a la arquitectura de información del portal de la ONBC.

2.11.1. Arquitectura del CMS Drupal

El sistema está implementado sobre Drupal, por lo cual su arquitectura hereda de este CMS: una arquitectura modular que permite ampliar sus funcionalidades a través de métodos uniformes de desarrollo e integración de nuevos módulos, pues es la herramienta seleccionada para dar cumplimiento al objetivo general del presente trabajo. A continuación se muestra la descripción de cada una de las capas de la composición arquitectónica del sistema.

- ✓ **Los Datos:** son los elementos básicos en que Drupal almacena la información, los contenidos. Así a medida que el sistema crece, lo va haciendo el número de Nodos los cuales van formando un depósito de Nodos cada vez mayor.

- ✓ **Los Módulos:** son los elementos que operan sobre los nodos y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada portal web. Son complementos que se instalan en el sistema proporcionándole nuevas funcionalidades.
- ✓ **Los Bloques y Menú:** permiten estructurar y organizar los contenidos en la página web.
- ✓ **Roles y Permisos:** actualmente, la mayor parte de sitios web son multiusuario, por lo que la seguridad y control de los usuarios es un punto clave para garantizar la integridad de la información almacenada. Con esta finalidad Drupal dispone de un registro de usuarios y de roles que permiten especificar qué tareas pueden realizar y a qué contenidos puede acceder cada tipo de usuario.
- ✓ **Las Plantillas:** son las que establece la apariencia gráfica o estilo de la información que se le muestra al usuario. Esta separación entre información y aspecto gráfico permite cambiar el diseño u apariencia del sistema sin necesidad de modificar los contenidos.

Patrón de arquitectura Modelo Vista Controlador (MVC)

Uno de los patrones que se ajustan a Drupal es el Modelo Vista Controlador. A grandes rasgos este tipo de arquitectura busca distribuir la aplicación en tres partes diferentes que se comunican entre sí, garantizando aplicaciones de fácil mantenimiento y escalables la siguiente imagen muestra el flujo de ejecución del patrón MVC.



Imagen 2: Flujo de ejecución del patrón MVC

El **modelo** está constituido por la base de datos (BD). El modelo indica al controlador que actualice la vista, con los datos de respuesta a la solicitud del usuario.

Por su parte el **controlador** lo constituyen el Núcleo de Drupal y los diferentes Módulos (Los que se encuentran en el núcleo y los que son agregados luego de instalado Drupal). Las peticiones se envían al controlador que indica al modelo la funcionalidad a ejecutar y ordena la actualización de la vista con la información resultante.

La **vista** la constituyen el conjunto de ficheros HTML, JavaScript y las hojas de estilo en cascada (CSS) representando los Temas y Lógica de los Temas que visualizan la información sobre la interfaz gráfica de usuario. Además es aquí donde se definen los manejadores de eventos que invocan a las funciones del controlador.

La siguiente imagen muestra la arquitectura del sistema basada en la arquitectura de Drupal. Algunos de los módulos contribuidos por la comunidad internacional de Drupal utilizados en la propuesta de solución son los siguientes: las **Vistas** y **Reglas**, siendo la **Cesta** y **Enlaces de la cesta** los módulos implementados por los desarrolladores.

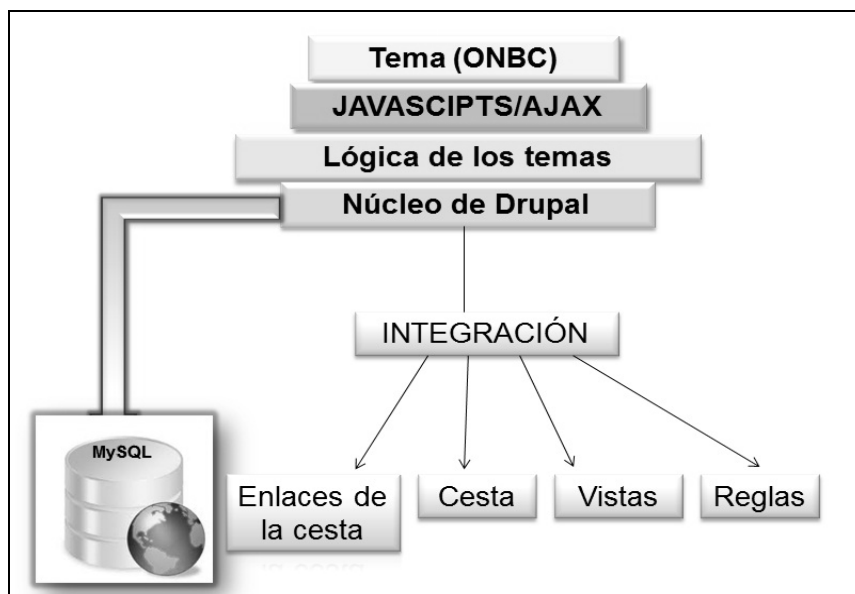


Imagen 3: Modelo de arquitectura de Drupal

El CMS Drupal está desarrollado en PHP siguiendo un modelo de programación estructurada y haciendo uso de un sistema de bases de datos relacional. El núcleo de Drupal es el encargado de la integración de los módulos que sean instalados. Este le envía la información procesada de los módulos al componente de la lógica del tema que se encarga de la visualización de la información en las vistas del sistema. Finalmente los temas que se desarrollen utilizan los datos, permitiendo pre-procesar o alterar los mismos antes de ser mostrados. A su vez estos pueden utilizar las funciones del API de JavaScript y AJAX para brindar una interfaz más intuitiva a los usuarios. Drupal incluye además un conjunto de bibliotecas que brindan un grupo de servicios como conexión y administración de la base de datos, gestión de procesos de envío de correos e internacionalización. Drupal cuenta además con un entorno de integración de utilidades que posibilita ampliar las funcionalidades del sistema a través de métodos uniformes de desarrollo e integración de nuevos módulos. La propuesta de solución se basa en la utilización de módulos previamente incluidos por el núcleo de Drupal y por otros que son contribuidos por la comunidad de Drupal internacional. Además se implementan módulos para lograr que los componentes sean portables y reutilizables en otros sistemas funcionando de manera independiente (FG, 2012).

Descripción de los Módulos principales de la propuesta de solución:

Módulos contribuidos de la comunidad de Drupal utilizados:

- ✓ **Vista:** la acción de este módulo es definir páginas, temas dándole una vista detallada de cada elemento seleccionado.
- ✓ **Rules:** permite definir las reglas que se necesiten para un determinado problema, en este caso se definen reglas para la cantidad máxima de registros que se puede agregar a la cesta de registros y si el registro se encuentra adicionado en la cesta.

Módulos desarrollados:

- ✓ **Cesta:** contiene la cesta de registros en la cual desde el Catálogo en línea se pueden adicionar los registros.

- ✓ **Enlaces de la cesta:** crea enlaces especializadas para visualizar la cesta de registros como **Ver cesta** y los enlaces de los registros que en encuentran adicionados en ella.

2.11.2. Arquitectura de Información del sistema de la ONBC

La Arquitectura de Información (AI) es un término que se utilizó por primera vez en 1975 por Richard Saul Wurman que lo definió como: el estudio de la organización de la información con el objetivo de permitir al usuario encontrar su vía de navegación hacia el conocimiento y la comprensión de la información.

La arquitectura de la información de un sitio web, como resultado de la actividad, comprende los sistemas de organización y estructuración de los contenidos, los sistemas de rotulado o etiquetado de dichos contenidos, y los sistemas de recuperación de información y navegación que provea el sitio web.

Las técnicas de arquitectura de información, llevadas a cabo para obtener el esbozo del sistema fueron:

Técnicas de interacción con el usuario

Técnicas mediante las que se obtiene información relacionada con los usuarios del producto final. Esta información constituye la base para lograr un diseño centrado en el usuario, sobre la que sustentan las posteriores etapas de producción.

- ✓ **Reunión:** encuentros que se hacen durante diferentes etapas del proceso de producción. Se aconseja realizarlas con un moderador, o un representante de las partes implicadas (los productores y los usuarios del producto).
- ✓ **Entrevista y encuesta:** contactos personalizados con usuarios, ya sea de manera oral o escrita.

Técnicas de interacción con el contexto

Son las técnicas que buscan información de productos similares o productos que le hacen competencia al que se está realizando. El objetivo principal de la técnica es conocer qué cualidades tienen los productos similares o de la competencia, para poder mejorar y superar; o qué dificultades tienen estos productos revisados, para no repetirlos en el sistema, dándole solución y obteniendo una ventaja competitiva.

- ✓ **Evaluación de productos similares:** se hace necesaria la revisión de productos similares al que se va a realizar, similitud que puede estar sustentada en aspectos de contenido, diseño o programación.

Después de realizar el proceso de captura de requisitos y teniendo en cuenta el análisis de portales homólogos, se obtuvo el esbozo de la estructura del sistema y una breve descripción de cada uno de los elementos de la estructura (Ver Anexo 5).

2.12. Patrones de diseño

El núcleo de Drupal 7 está diseñado haciendo uso de una arquitectura modular, las características estructurales presentes en este CMS son resultado de la aplicación de patrones de diseño. A continuación se especifica cómo se evidencian los patrones de diseños **Gang of Four** (GOF) en español (La Pandilla de los Cuatro).

2.12.1. Patrones GOF

Solitario o instancia única (Singleton): este patrón se evidencia si se asocian los módulos y los temas como objetos, cada uno de ellos en si no almacena datos, sino que representan un conjunto de funciones que extienden el núcleo para añadirle funcionalidades al mismo, cada uno por separado. De esta forma se pueden considerar estos como una clase con una instancia única.

Decorador (Decorator): la evidencia de este patrón se encuentra en la definición de **hooks** (ganchos) por parte de los módulos del núcleo de Drupal y otros módulos contribuidos, que posibilitan que otros puedan extender el comportamiento de dichas funciones. Los **hooks** son funciones que permiten la comunicación con todo el núcleo de Drupal, pueden ser eventos u operaciones con campos. De esta forma se brinda la flexibilidad de que nuevos módulos puedan modificar el comportamiento del núcleo en cuanto al tratamiento de los datos y en cada uno de los eventos del sistema. Este comportamiento podría ser imitado por el uso de decoradores, por ejemplo con el **hook_views_exposed_form_alter** definido por el módulo **Views** de Drupal, el módulo de Catálogo en línea implementa ese **hook** para alterar la presentación de los filtros en la vista de búsqueda de registro(s).

Observador (Observer): el comportamiento anterior descrito en la utilización del patrón decorator, puede ser visto también como evidencia de la aplicación del patrón Observador. Este patrón se evidencia en el sistema en la implementación del **hook_entity_delete** para eliminar las referencias de los elementos de las listas a la hora de borrar registro(s).

Puente (Bridge): la capa de abstracción de datos de Drupal se encuentra implementada siguiendo el patrón Puente. Cada módulo debe ser programado de manera que sea independiente del motor de base de datos que utiliza el sistema. Esto se logra por la capa de abstracción de base de datos, sobre la que se pueden desarrollar nuevas funcionalidades siguiendo la API definida, para añadir soporte para nuevos sistemas gestores de base de datos, sin necesidad de modificar el código de los módulos.

Comando (Command): muchos de los hooks definidos por el núcleo de Drupal usan el patrón Comando para reducir número de funciones que son necesarias implementar, pasando la operación como parámetro conjuntamente con los argumentos. De esta forma se evidencia en el sistema de **hooks**, que cada módulo no necesariamente debe definir cada **hook**, sino solo los que tienen que ver con su propósito y funcionamiento. En el desarrollo de la solución propuesta se evidencia este patrón ya que los módulos implementados no tienen que definir cada **hook**, sino los que se vayan a aplicar.

Cadena de responsabilidades (Chain of responsibility): el sistema de menús de Drupal es la evidencia del patrón cadena de responsabilidades. En cada petición de una página, el sistema de menús de Drupal determina si hay algún módulo para responder la petición, si el usuario tiene acceso al recurso solicitado y que función se debe llamar para procesar la petición. En este proceso se transmite el mensaje de la petición por cada uno de los componentes que se encuentran inmersos. De esta forma se continúa la cadena hasta que un módulo atienda la petición, hasta que un módulo deniegue el acceso o hasta que la cadena se agote. Todas las peticiones de rutas definidas por los módulos desarrollados evidencian este patrón.

2.13. Diagrama de paquetes

Los paquetes son elementos que permiten dividir un modelo en partes manejables mediante la agrupación de clases u otros paquetes, de ahí que se usen para reflejar la organización de paquetes y los elementos que los componen (Falgueras, 2008).

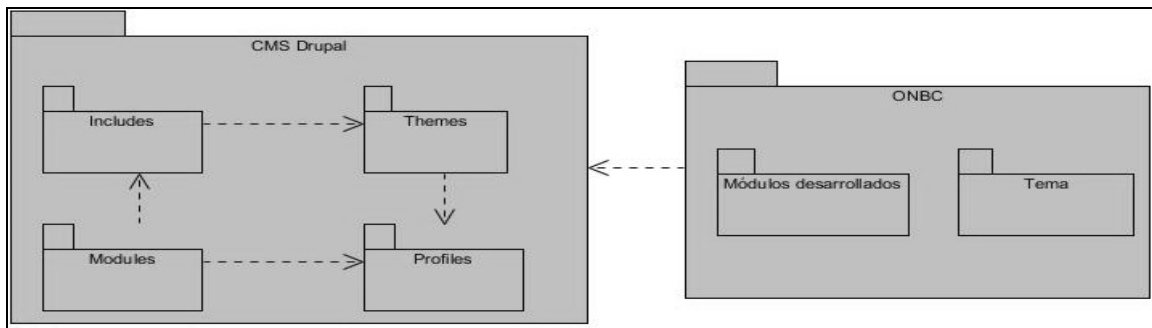


Imagen 4: Diagrama de Paquetes del sistema.

En el diagrama se representa un paquete nombrado “CMS Drupal”, que incluye los paquetes contenedores de los componentes del CMS y otro paquete de los más generales es el denominado “ONBC”, que incluye los elementos que conforman la aplicación. A continuación se describe el contenido de cada uno:

✓ **Paquete CMS Drupal:**

- **Includes (incluidos):** contiene un conjunto de ficheros indispensables para el funcionamiento de los módulos y el sistema Drupal, como las API de conexión a la base de datos.
- **Modules (módulos):** incluye todos los módulos, que permiten las distintas funcionalidades del CMS, cuando desee incorporar un nuevo módulo sólo se tiene que copiar en dicha carpeta.
- **Themes (temas):** incluye los temas que tiene de forma predefinida Drupal, permiten cambiar el estilo de la vista que se le muestra al usuario, cuando desee incorporar un nuevo tema sólo se tiene que copiar en dicha carpeta.
- **Profiles (perfiles):** es un conjunto de instrucciones que permite definir qué módulos y opciones se quieren activar durante el proceso de instalación de Drupal.

✓ **Paquete ONBC:**

- **Tema desarrollado:** contiene el tema desarrollado para el sistema.
- **Módulos desarrollados:** contiene todos los módulos que fue necesario implementar para el desarrollo de la aplicación.

2.14. Conclusiones parciales

El sistema diseñado garantiza y facilita la gestión de la información relacionada con los servicios, cursos, ediciones, consultas y otras informaciones que maneja la ONBC. La arquitectura modular del sistema Drupal permite la ampliación de las funcionalidades del sistema y la aplicación de los patrones de diseño garantizan lograr un esquema del mismo de fácil entendimiento, extensión y mantenimiento, además de la reutilización de componentes en aplicaciones. La planificación de las iteraciones a realizar durante el proceso de desarrollo del sistema proporciona una mejor organización del trabajo.

Capítulo 3: Implementación y prueba

3.1. Introducción

El presente capítulo contempla los elementos relacionados con la implementación del sistema, reflejándose en este sentido el diagrama de despliegue del mismo, así como el plan de releases y los estándares de codificación empleados. Además, se hace referencia a las pruebas que se realizan para validar el correcto desempeño del sistema.

3.2. Técnicas de validación de requisitos

Con el objetivo de ratificar que los requisitos del software obtenidos definen el sistema que el cliente desea, se llevó a cabo un proceso de validación de los mismos en la etapa Planificación-Definición, para el cual se emplearon las siguientes técnicas:

Revisión técnica formal

El primer mecanismo para la validación de los requisitos es la revisión técnica formal (RTF). El equipo de revisión incluye ingenieros del sistema, clientes, usuarios, y otros intervinientes que examinan la especificación del sistema buscando errores en el contenido o en la interpretación, áreas donde se necesitan aclaraciones, información incompleta, inconsistencias (es un problema importante), requisitos contradictorios, o requisitos imposibles o inalcanzables (Pressman, 2001).

Aunque la validación de requisitos puede guiarse de manera que se descubran errores, es útil chequear cada requisito con un cuestionario. Las siguientes cuestiones representan un pequeño subconjunto de las preguntas que pueden plantearse (Ver Anexo 6):

- ✓ ¿Está el requisito claramente definido?
- ✓ ¿Pueden interpretarse mal?
- ✓ ¿Está identificado el origen del requisito (por ejemplo: persona, norma, documento)?
- ✓ ¿El planteamiento final del requisito ha sido contrastado con la fuente original?

- ✓ ¿El requisito está delimitado en términos cuantitativos?

Las preguntas planteadas en la lista de comprobación ayudan asegurar que el equipo de validación dispone de lo necesario para realizar una revisión completa de cada requisito (Pressman, 2001).

Las RTF constituyen el mecanismo primario para la validación de los requisitos. Desarrollada a partir de las revisiones que realice el equipo destinado a ello, en este caso, este equipo fue compuesto por:

- ✓ Cliente:
 - MSc Yarina Amoroso Fernández
- ✓ Equipo de desarrollo del sistema

Los diferentes encuentros y entregas de los artefactos para su revisión estuvieron encaminados a la examinación de la especificación y a la búsqueda de errores en el contenido, a la interpretación, a la información faltante, inconsistencias, conflictos entre los requisitos y la existencia de requisitos irreales (inalcanzables).

Construcción de prototipos

Mediante los prototipos se le mostró al cliente un modelo ejecutable del portal web que le permitió tener una visión preliminar de cómo sería el sistema y a través de la interacción con los mismos se comprobó si satisfacía sus necesidades, los mismos fueron aprobados por el cliente.

3.3. Métricas aplicadas a los requisitos

Con el objetivo de medir la calidad de la especificación de los requisitos se aplicó una de las métricas propuestas para la metodología SXP, Calidad de la especificación (CE).

Para obtener cuán entendibles y precisos son los requisitos, primeramente se calcula el total de requisitos de la especificación como se muestra a continuación:

Nr.: el total de requisitos de especificación.

Nf: cantidad de requisitos funcionales.

Nnf: cantidad de requisitos no funcionales.

$$Nr = Nf + Nnf$$

$$Nr = 44 + 24$$

$$Nr = 68$$

Como resultado de la sustitución de los valores, para el sistema de la ONBC se obtiene: 68 requisitos en total.

Para determinar, finalmente, la Especificidad de los Requisitos (ER) o ausencia de ambigüedad en los mismos se realiza la siguiente operación:

$$ER = Nui / Nr$$

Donde *Nui* es el número de requisitos para los cuales todos los revisores tuvieron interpretaciones idénticas. Mientras más cerca de 1 esté el valor de ER, menor será la ambigüedad.

Para el caso de los requisitos obtenidos para el sistema solo uno produjo contradicción en las interpretaciones. Sustituyendo las variables se obtiene:

$$ER = 67/68$$

$$ER = 0.98$$

Arrojando un resultado final satisfactorio, indicando que el grado de ambigüedad de los requisitos es sumamente bajo (2%) ya que el 98% es entendible. El requisito ambiguo fue modificado y validado para garantizar su correcta comprensión.

3.4. Implementación

Las historias de usuarios seleccionadas para ser implementadas en cada iteración se van realizando durante el transcurso de la iteración a la cual pertenecen. Como parte de este plan se descomponen las historias de usuario en tareas de desarrollo, asignando a un grupo de desarrollo o a una persona responsable de la implementación. Teniendo como premisa la planificación realizada anteriormente, se llevaran a cabo tres iteraciones de desarrollo sobre el sistema, permitiendo que al final se logre un producto con todas las restricciones y características deseadas por el cliente. A continuación se detallan cada una de las iteraciones.

3.4.1. Iteración 1

En esta primera iteración se desarrollan las historias de usuarios de mayor prioridad en el sistema con el objetivo de obtener una primera versión del producto con las principales características o funcionalidades para ser mostrado al cliente (Ver Anexo 7).

Tabla 6: Tareas generadas por la historia de usuario Gestionar usuario

Tarea de ingeniería	
Número de tarea: 1	Nombre de historia: Gestionar usuario.
Nombre de la tarea: Configurar la interfaz para la autenticación de los usuarios que viene por defecto con el CMS.	
Tipo de tarea: Agregar campos nuevos.	Puntos Estimados(días): 2
Fecha inicio: 29/2/2014	Fecha fin: 30/2/2014
Programador responsable: Mirileydis Martínez Ajete	
Descripción: Se plasma mediante el formulario los elementos necesarios para que se realice la autenticación en el sistema de los usuarios.	

3.4.2. Iteración 2

En el transcurso de esta iteración se implementan las historias de usuarios con nivel de prioridad medio para el sistema. Esto permite agregar nuevas funcionalidades al sistema por lo que se obtiene una versión más completa y el cliente podrá observar algunas de las facilidades en cuanto a las funcionalidades del sistema en su culminación.

Tabla 7: Tareas generadas por la historia de usuario Servicio de información

Tarea de ingeniería	
Número de tarea: 1	Nombre de historia: Servicio de información.
Nombre de la tarea: Crear a través de la interfaz un tipo de contenido Bufetes.	
Tipo de tarea: Agregar campos.	Puntos Estimados(días): 2
Fecha inicio: 11/3/2014	Fecha fin: 13/3/2014
Programador responsable: Dayana Francia Ofarrill	
Descripción: El usuario podrá visualizar este servicio a través de una vista que muestra información sobre los bufetes de todo el país.	

3.4.3. Iteración 3

En esta iteración se implementan las historias de usuarios con nivel de prioridad bajo para el sistema. Esto permite agregar nuevas funcionalidades al sistema por lo que se obtiene una versión más completa y el cliente podrá observar algunas de las facilidades en cuanto a las funcionalidades del sistema en su culminación.

Tabla 8: Tareas generadas por la historia de usuario Gestionar encuesta

Tarea de ingeniería

Número de tarea: 1	Nombre de historia: Gestionar encuesta.		
Nombre de la tarea: Utilizar las funcionalidades que brinda el módulo Encuesta.			
Tipo de tarea: Agregar datos.		Puntos Estimados(días): 1	
Fecha inicio: 14/3/2014		Fecha fin: 14/3/2014	
Programador responsable: Dayana Francia Ofarrill			
Descripción: El usuario podrá visualizar las encuestas publicadas en el sitio.			

3.4.4. Plan de liberación

El propósito de una iteración es transformar un conjunto de la LRP en un incremento en la funcionalidad del producto que sea potencialmente entregable a los usuarios. Para ello, previamente a cada iteración, se realiza una reunión de planificación de iteración para determinar en qué funcionalidad del producto trabajará el equipo. Las iteraciones a realizar, sus características y el orden de las historias de usuarios con su planificación estimada para ser implementadas se recogen en la plantilla Plan de liberación, la cual se muestra a continuación:

Tabla 9: Plan de liberación

Liberación	Descripción de la iteración	Orden de la Historia de Usuario a implementar	Duración total (En semanas)
1	Realizar los servicios en línea.	<ol style="list-style-type: none"> 1. HU05 Servicio de solicitud de certificación 2. HU06 Servicio de consulta gratuita 3. HU13 Servicio especializado en recursos de casación 4. HU14 Servicio de bufete especializado 	2.4

		5. HU15 Servicio de asesoría, representación y defensa	
2	Implementación del módulo Catálogo.	6. HU09 Catálogo en línea	3
3	Realización del resto de las funcionalidades acordadas con el cliente.	7. HU01 Autenticar usuario 8. HU02 Gestionar información 9. HU03 Gestionar usuario 10. HU04 Gestionar enlaces a otros sitios 11. HU07 Buscar Información 12. HU08 Gestionar información de los Cursos 13. HU10 Mostrar cuantía de obsolescencia de las noticias creadas 14. HU11 Servicio de Información 15. HU12 Gestionar noticia 16. HU16 Políticas de Seguridad 17. HU17 Términos de Uso 18. HU18 Gestionar Encuesta	3.5

3.4.5. Estándares de codificación empleados

Drupal cuenta con estándares de codificación que permiten lograr una implementación limpia y organizada siendo esta entendible para cualquier desarrollador que utilice el código desarrollado (FG, 2012).

Indentación: la indentación consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. En programación se utiliza la indentación para anidar elementos. En Drupal la indentación debe ser de 2 espacios.

Etiquetas de apertura y cierre de PHP: el código PHP se debe escribir siempre utilizando las etiquetas `<?php` y `?>` y en ningún caso la versión corta `<? y ?>`.

Estructuras de control: con respecto a las estructuras de control, se debe tener en cuenta las siguientes normas:

- ✓ Debe haber un espacio entre el comando que define la estructura (if, while, for) y el paréntesis de apertura permitiendo no confundir las estructuras de control con la nomenclatura de las funciones.
- ✓ La llave de apertura ({) se sitúa en la misma línea que la definición de la estructura, separada por un espacio.
- ✓ Se recomienda usar siempre las llaves { } aún en los casos en que no sea obligatorio su uso.
- ✓ Las estructuras else y elseif se deben escribir en la línea siguiente al cierre de la sentencia anterior.

Funciones: los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guión bajo. Además, se debe incluir siempre como prefijo el nombre del módulo o tema, para evitar así duplicidad de funciones. En su declaración, después del nombre de la función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior.

Arreglos: los valores dentro de un arreglo se deben separar por un espacio (después de la coma que los separa).

Variables globales: estas se deben declarar utilizando un guión bajo inicial, seguido del nombre del módulo o tema y otro guión bajo antes del nombre de la variable.

3.5. Diagrama de despliegue

Un diagrama de despliegue modela la arquitectura en tiempo de ejecución de un sistema. Este muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se instalan en esos nodos (Falgueras, 2008).

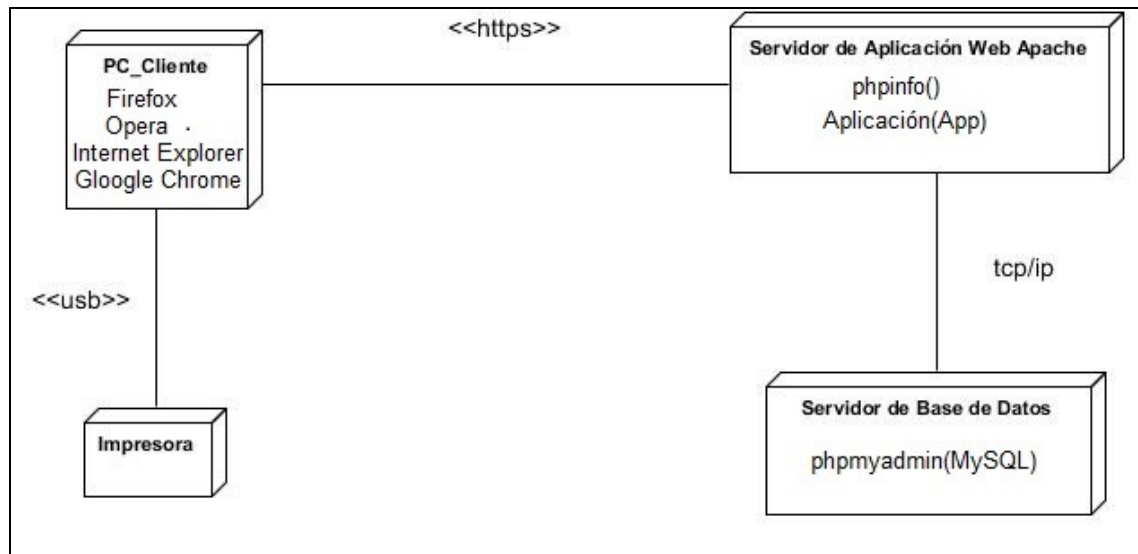


Imagen 5: Diagrama de despliegue

Nodos:

- ✓ PC Cliente: representa una computadora desde la cual el usuario accede a la aplicación.
- ✓ Servidor de Aplicación Apache: representa una estación donde se instala el servidor Apache sobre el cual se despliega la aplicación.
- ✓ Servidor de Base de Datos MySQL: representa el servidor donde se instala el SGBD MySQL que responde a las peticiones hechas por la aplicación.
- ✓ Impresora: Representa una estación donde se puede imprimir todas las páginas del sistema.

Protocolos:

- ✓ TCP/IP (Protocolo de comunicación y transmisión/ 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.

- ✓ HTTPS (Protocolo seguro de transferencia de hipertexto): es un protocolo de aplicación basado en el protocolo HTTP, destinado a la transferencia segura de datos de Hipertexto, es decir, es la versión segura de HTTP.

Puerto:

- ✓ USB (Conector de alimentación eléctrica) diseñado para estandarizar la conexión de periféricos, como mouse, teclados, memorias USB, escáneres, cámaras digitales, teléfonos móviles, impresoras.

Elementos que componen los nodos:

- ✓ Firefox, Opera, Internet Explorer y Google Chrome son los navegadores web desde donde los usuarios pueden acceder a la aplicación.
- ✓ Phpinfo () muestra información sobre la configuración de PHP.
- ✓ Aplicación es la que gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación.
- ✓ Phpmyadmin es la herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web.

3.6. Estrategia de prueba

Las pruebas son un instrumento adecuado para determinar el estado de la calidad de un producto en las cuales un sistema o componente es ejecutado bajo condiciones o requisitos especificados. Una de las principales fortalezas de la metodología de desarrollo SXP es el proceso de pruebas, el cual permite asegurar el éxito del producto al realizarse de manera continua, proporcionando la obtención de un producto de mayor calidad, pues los errores son detectados en un corto plazo de tiempo y se corrigen de una manera más sencilla.

Se le realizaron al sistema pruebas de unidad, las que consisten en pruebas formales que permiten declarar que un módulo está listo y terminado. De ellas, se aplicó el método de Caja Negra con el objetivo

de comprobar que las funciones son operativas a través de la interfaz del software, que la entrada se acepta de forma adecuada y se produce un resultado correcto, manteniendo la integridad de la información externa. Además se aplicó el método de caja blanca utilizando la técnica de camino básico y por último la prueba de Usabilidad utilizando la lista de chequeo de la empresa de CALISOFT.

3.6.1. Método de caja negra

Dentro del método de Caja Negra la técnica de la Partición de Equivalencia es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar.

Con el objetivo de comprobar que las funcionalidades del sistema se realizan de forma correcta y responden a las necesidades del cliente, se realizaron un total de 3 iteraciones del método de caja negra, obteniéndose los siguientes resultados:

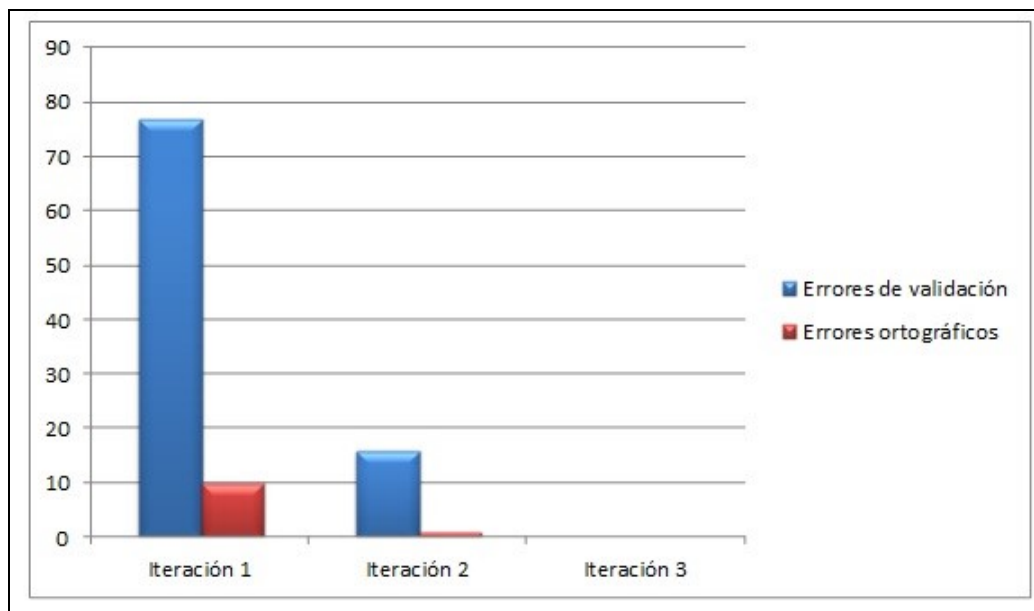


Imagen 6: Resultados de la prueba

En una primera iteración se detectaron un total de 87 no conformidades entre todas las funcionalidades implementadas, generalmente errores de validación y de escritura. En la segunda iteración se obtuvieron 17 no conformidades donde aún existían errores de validación. Finalmente en una tercera iteración se obtuvieron resultados satisfactorios al no detectarse no conformidades.

A continuación se muestran los resultados obtenidos en la primera iteración para el caso de prueba 0501 asociado a la Historia de Usuario Servicio de solicitud de certificación (Ver Anexo 8. Descripción del caso de prueba de aceptación número 01 de la HU # 05).

Registrado el usuario administrador, el cual es administrador del sistema para la ONBC, se selecciona la opción Certificación de documentos y el sistema muestra correctamente la interfaz visualizando los diferentes documentos que se pueden certificar (Ver Anexo 9: Interfaz “Certificación de documentos”), luego se selecciona la opción Certificación de matrimonio en Cuba (Ver Anexo 10: Formulario para crear certificación de matrimonio en Cuba) en la que se detectaron 4 no conformidades las cuales se correspondían a errores de validación.

Luego de introducidos los datos correctamente (Ver Anexo 11. Resultado del caso de prueba 0501) y seleccionada la opción Aceptar, el sistema muestra el mensaje “Ya ha enviado este formulario” y la opción de ver otros envíos anteriores, luego se muestra información del envío realizado, obteniéndose para este caso una respuesta satisfactoria del sistema.

Para cada uno de los casos de pruebas se verificaron las variantes con datos correctos, incorrectos o incompletos, este último se refiere a dejar algunos de los campos obligatorios en blanco.

3.6.2. Método de caja blanca

Al sistema desarrollado se le aplicó el método de caja blanca, haciendo uso de la técnica del camino básico, con el objetivo de evaluar la complejidad lógica de un diseño procedimental y usar esta medida como guía para la definición de un conjunto básico de caminos de ejecución (Pressman, 2001). Esta prueba permite garantizar que en los casos de prueba obtenidos a través del camino básico se ejecute cada sentencia del programa por lo menos una vez. La generación del grafo de flujo tiene que descomponer las condiciones compuestas en condiciones sencillas.

A continuación se analizan y enumeran las sentencias de código del método `uc_cart_view_table` contenido en el módulo `cart` en el fichero `uc_cart.module`.

```
function uc_cart_view_table($table) {  
    $table['#columns']=array( //1  
        'desc' =>array(  
            'cell' => 'Nombre del registro',  
            'weight' => 2,  
        ),  
        'qty' => array(  
            'cell'=>'Cantidad',  
            'weight' => 3,  
        ),  
        'remove' => array(  
            'cell' => t(' Acción'),  
            'weight' => 0,  
        ),  
    );  
  
    $subtotal = 0; //2  
    foreach (element_children($table) as $i) { //3  
        $subtotal += $table[$i]['#total']; //4  
        $table[$i]['desc']['#cell_attributes'] = array('class' => array('desc'));  
        $table[$i]['qty']['#cell_attributes'] = array('class' => array('qty'));  
        $table[$i]['remove']['#cell_attributes'] = array('class' => array('remove'));  
    }  
    return $table; //5  
}
```

Imagen 7: Función que devuelve la tabla donde se recogen los registros una vez agregados a la cesta.

Después de este paso 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.

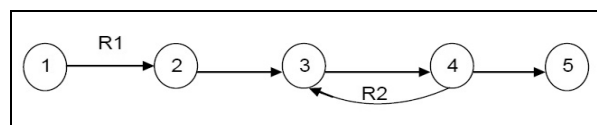


Imagen 8: Grafo de flujo asociado al método `uc_cart_view_table`.

Una vez construido el grafo de flujo asociado al procedimiento anterior se determina la complejidad ciclomática, la cual es una métrica de software útil pues proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado como complejidad ciclomática define el número de caminos independientes del conjunto básico de un programa y da un límite superior para el número de pruebas que se deben realizar.

Cálculo de la complejidad ciclomática para el grafo de flujo de la imagen 7

- ✓ $V(G) = a - n + 2$
 - $V(G) = 5 - 5 + 2 = 2$
- ✓ $V(G) = p + 1$
 - $V(G) = 1 + 1 = 2$
- ✓ $V(G) = r$
 - $V(G) = 2$

Siendo “V (G)” el valor de la complejidad ciclomática, “a” la cantidad total de aristas, “n” la cantidad total de nodos, “p” la cantidad total de nodos predicados (nodos de los cuales parten dos o más aristas) y “r” la cantidad total de regiones, se incluye el área exterior del grafo como una región más.

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:

Camino básico #1: 1-2-3-4-5

Camino básico #2: 1-2-3-4-3-4-5

3.6.3. Usabilidad

Evaluar la Usabilidad de un sitio, permiten identificar los elementos y sus niveles de compromisos con el cumplimiento de los requisitos de efectividad, eficiencia y satisfacción para un contexto de uso determinado.

Las pruebas de usabilidad se realizaron mediante la lista de chequeo (diseñada por el Laboratorio Industrial de Pruebas de Software perteneciente a la empresa CALISOFT) relacionando el cumplimiento

de aspectos incluidos en áreas o indicadores como se muestran en la figura siguiente. Para el cálculo del por ciento definido se responde una serie de preguntas en función de los aspectos analizados, y se le otorga una evaluación, el mismo se evalúa de 1 en caso de mal y se le otorga el valor 0 en caso de que el elemento revisado no presente errores, luego se calcula el porcentaje de preguntas evaluadas de bien en función del total y si el resultado es superior al 70 % se considera que la prueba arrojó resultados positivos. El resultado de estas pruebas indicó una buena usabilidad del sistema, arrojó más del 75% en los indicadores de cada área.

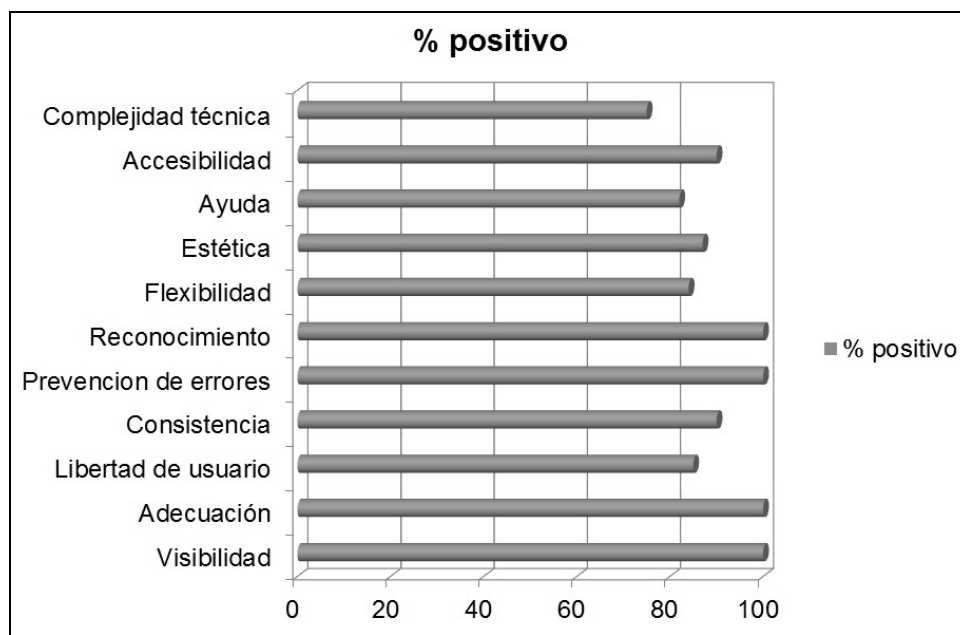


Imagen 9: Resultados de la prueba de Usabilidad.

3.7. Conclusiones parciales

La utilización de los estándares de codificación con que cuenta Drupal, permitió la obtención del código del sistema de manera limpia y organizada, siendo esta entendible para cualquier programador que utilice el código desarrollado. La especificación del diagrama de despliegue permitió contar con una vista de cómo quedará el sistema en el momento de su explotación por parte de los usuarios de la ONBC. La realización de las pruebas de software al sistema web permitió encontrar los errores existentes para poder solucionarlos, posibilitando determinar y asegurar la calidad, robustez y fiabilidad del mismo.

Conclusiones generales

Al finalizar el presente trabajo investigativo se arribó a las siguientes conclusiones:

- ✓ El estudio realizado sobre las diferentes tecnologías de desarrollo permitió seleccionar la metodología y herramientas más adecuadas a utilizar.
- ✓ La implementación de cada una de las funcionalidades definidas para el sistema, la inclusión de un diseño de interfaz sencillo y adaptado a cualquier tipo de usuario, así como la definición de roles para garantizar la seguridad en la administración de la información, permitieron obtener un producto de software a la altura de las necesidades de la ONBC.
- ✓ La aplicación de las pruebas permitió corregir las no conformidades de manera satisfactoria y determinar la usabilidad del sistema, arrojando los resultados esperados por el equipo de desarrollo y el cliente.
- ✓ Con el empleo del sistema desarrollado, la ONBC mejorará la gestión de los servicios que brinda y ofrecerá un mayor número de estos a los usuarios.

Recomendaciones

Una vez concluido el presente trabajo se recomienda:

- ✓ Implementar el servicio de cambio de idioma para el sistema.
- ✓ Implementar en la vista de la Cesta de registro(s) que estos puedan ser enviados por correo como referencia bibliográfica para después ser cargadas en cualquier gestor bibliográfico.

Referencias bibliográficas

- A, Venete. 2010.** Introducción a los Patrones de Diseño. [En línea] 2010. <http://mahara.uji.es/artefact/file/download.php?file=54536&view=4648>.
- Benchimol, Daniel. 2011.** *HACKING DESDE CERO: Manuales Users (Spanish Edition)*. Argentina : Creative Andina Corp, 2011. 987177303X, 9789871773039.
- Bobbio, Norberto. 1997.** *Teoría general del derecho*. Madrid : PublidisaS.L, 1997.
- Dhar, Santonu Kumar. 2010.** *Building job sites with Joomla*. s.l. : Akshara Aware, 2010. ISBN 978-1-849512-22-0.
- Douglass, Robert T. 2010.** *Building Online Communities with Drupal*. 2010. ISBN:1-59059-562-9.
- Drupal. 2010.** Drupal Hispano. [En línea] 2010. <http://drupal.org.es/Drupal>.
- Española, Abogacia. 2010.** Abogacia Española. *Abogacia Española*. [En línea] 2010. <http://www.abogacia.es>.
- Falgueras. 2008.** *Ingeniería del software*. 3. Barcelona : s.n., 2008. 423-84-909410-2-2.
- FG, RODRÍGUEZ. 2012.** *Experto en Drupal*. s.l. : Forcontu S.L, 2012. Vol. 3. 978-84-939410-2-4.
- González Boticario, Jesús y Gaudioso Vázquez, Elena. 2001.** *Aprender y formar en Internet*. Madrid, España : International Thomson Editors Spain Paraninfo, S. A, 2001. 84-283-2743-2.
- Habana, Bufete Internacional de la. 2009.** Bufete Internacional de la Habana. *Bufete Internacional de la Habana*. [En línea] 2009. <http://www.bufeteinternacional.cu>.
- Hejlsberg, Anders y de Icaza, Miguel. 2008.** *Framework Design Guideline*. España : s.n., 2008. ISBN: 978-0-321-54561-9.
- Kendall, Kenneth E. y Kendall, Julie E. 2005.** *Análisis y Diseño de Sistemas. Sexta Edición*. México : Pearson Educación de México, S.A.de C.V, 2005. 970-26-0577-6.
- Kniberg. 2006.** lastrincheras. *lastrincheras*. [En línea] 12 de 4 de 2006. [Citado el: 28 de 11 de 2013.] <http://infoq.com/minibooks/scrum-xp-from-the-trenches>. 2.
- Leffingwell, Dean y Widrig, Don. 2003.** *Using Software Engineering Techniques for Business Modeling. The Unified Modeling Language". Managing software requirements*. United States of America : Pearson Education, 2003. 0-321-12247-X.

- Leyva Samada, Lisandra Isabel. 2009.** Flujo de Investigación para la Metodología Ágil SXP. Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas. [En línea] 2009. http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_2435_09. TD_2435_09.
- Martín Fernández, Francisco J. y Hassan Monter, Yusef. 2003.** *Qué es la Arquitectura de la Información*. 2003. Vol. 2. 1886-8592.
- Moreiro, G. 1998.** *Introducción al estudio de la información y la documentación*. Antioquía : Medellín, 1998.
- Netbeans. 2012.** Bienvenido a NetBeans. [En línea] 2012. http://netbeans.org/index_es.html.
- ONBC. 2013.** Organización nacional de bufetes colectivos. [En línea] 2013. <http://www.onbc.cu/inicio>.
- Perú, Abogados. 2010.** Alpha Abogados Perú. [En línea] 2010. http://www.abogados.com.uy/_es/.
- Pressman, Roger S. 2001.** *Ingeniería del Software: una tecnología estratificada. Ingeniería del Software. Un enfoque práctico. Quinta Edición*. España : McGraw Hill, 2001.
- Prezi.** Metodologías. *Metodologías*. [En línea] <http://prezi.com/rwp5dkmrmets/metodologias-pesadas>.
- . **2013.** Prezi. [En línea] 2013. [Citado el: 10 de 12 de 2013.] <http://prezi.com/rwp5dkmrmets/metodologias-pesadas/>.
- Puertas Ortega, Juan y Orellana Zubieta, Francisco Javier. 2010.** Un-paseo-por-PHP. *Un-paseo-por-PHP*. [En línea] 2010. [Citado el: 4 de 12 de 2013.] <http://es.scribd.com/doc/51830143/Un-paseo-por-PHP>. DOI: 51830143.
- Rojas, Nohemi. 2010.** Framework vs CMS. [En línea] 2010. [Citado el: 11 de 2 de 2014.] <http://nohemirojas.wordpress.com/2010/03/25/cms-y-framaework-dos-conceptos-distintos/>.
- Romero, Gladys Marsi Peñalver. 2008.** *Metodología ágil para proyectos de software libre*. 2008. Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas. MA-GMPR-UR2.
- Rumbaugh, James , Jacobson, Ivar y Booch, Grady. 2007..** *El Lenguaje Unificado de Modelado Manual de Referencia*. s.l. : s.l. : Addison-Wesley Iberoameri, 2007. 9788478290871.
- Sánchez Maza, Miguel Ángel. 2012.** *JavaScript*. s.l. s.l. : IC Editorial, 2012. 978-8495733184.
- Sierra, Manue. 2012.** Qué es y para qué sirve el lenguaje CSS (Cascading Style Sheets - Hojas de Estilo). *Qué es y para qué sirve el lenguaje CSS (Cascading Style Sheets - Hojas de Estilo)*. [En línea] 2012. http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=546.
- Sócrates. 2000.** PHP. [En línea] 2000. <http://www.slideshare.net/socrates12854/libro-4-18955488>.

Software Talk. 2012. Netbeans vs Eclipse: An IDE comparison. [En línea] 2012. <http://software-talk.org/blog/2012/01/netbeans-vs-eclipse-an-ide-comparison/>.

Sommerville, Ian. 2005. *Ingeniería del software. Séptima Edición.* Madrid. España : Pearson Educación. S. A, 2005. 84-7829-074-5.

Tomlinson, Todd. 2010. *Beginning Drupal 7.* 2010. ISBN: 978-1-4302-2859-2.

Vértice, Equipo. 2009. *Diseño Básico de Páginas Web en HTML.* España : Publicaciones Vértice S. L, 2009. 978-84-9931-034-3.

W3C. 2013. Guía Breve de Servicios Web. *Guía Breve de Servicios Web.* [En línea] 2013. <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>.

Wordpress. 2009. Comunidad CodeIgniter. *Sitio web de Wordpress.* [En línea] 2009. [Citado el: 5 de 12 de 2013.] <http://comunidadcodeigniter.wordpress.com>.

Anexos

Anexo 1. Glosario de términos

- ✓ **API:** Acrónimo del inglés Application Programming Interface, traducido al español como Interfaz de programación de aplicaciones.
- ✓ **Aplicación:** En informática es un tipo de software diseñado para facilitar al usuario la realización de una determinada tarea o trabajo.
- ✓ **Aplicación web:** Se denomina aplicación web a aquellas aplicaciones que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.
- ✓ **GNU/GPL:** Licencia orientada principalmente a proteger la libre distribución, modificación y uso de software. Garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software.
- ✓ **HTML:** HyperText Markup Language, por sus siglas en inglés, en español Lenguaje de Marcado de Hipertexto.
- ✓ **JavaScript:** Lenguaje de script multiplataforma orientado a objetos, pequeño y ligero; no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores Web.
- ✓ **Librerías:** Conjunto de subprogramas utilizados para desarrollar software.
- ✓ **Módulo:** Es un software que agrupa un conjunto de subprogramas y estructuras de datos.
- ✓ **TCP/IP:** Se le denomina conjunto de protocolos TCP/IP, en referencia a los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP), que fueron dos de los primeros en definirse.

- ✓ **Multiplataforma:** Término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

Anexo 2. Fases y artefactos de la metodología SXP

- ✓ **Planificación-Definición:** fase en la que se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.

- **Artefactos:**

- Concepción del sistema.
- Lista de reserva del producto (LRP).
- Lista de riesgos.
- Historia de usuario.
- Modelo de Historia de Usuario del negocio.
- Modelo de diseño.

- ✓ **Desarrollo:** fase en la que se realiza la implementación del sistema hasta que esté listo para ser entregado.

- **Artefactos:**

- Tareas de ingeniería
- Cronograma de producción
- Estándar de programación
- Caso de prueba de aceptación

- Plan de releases
- ✓ Entrega: fase donde se pone en marcha el producto desarrollado y se hace la entrega al cliente.
 - Artefactos:
 - Manual de desarrollo
 - Manual de identidad
 - Manual de usuario
- ✓ Mantenimiento: fase en la que se realiza el soporte para el cliente.
 - Artefactos:
 - Gestión de cambios

Anexo 3. Requisitos funcionales del sistema para la ONBC

No.	Nombre	Descripción	Complejidad	Prioridad para el cliente
01	Autenticar usuario	Comprueba si los datos introducidos son igual que los datos almacenados.	Baja	Baja
02	Cambiar contraseña	Permite a un usuario registrado cambiar su contraseña.	Baja	Baja
03	Adicionar información	Permite al administrador crear nueva información en el sistema.	Baja	Media
04	Editar información	Permite al administrador modificar algunos campos	Baja	Media

		en la información ya publicada.		
05	Eliminar información	Permite al administrador eliminar información publicada en el sistema.	Baja	Media
06	Adicionar usuario	Permitirá al administrador crear un usuario que no exista en la base de datos.	Baja	Alta
07	Editar usuario	Permitirá al administrador modificar los datos de un usuario ya creado.	Baja	Alta
08	Eliminar usuario	Permitirá al administrador eliminar un usuario de la base de datos.	Baja	Alta
09	Adicionar enlace	Permitirá agregar un enlace al sistema.	Bajo	Media
10	Eliminar enlace	Permitirá eliminar un enlace del sistema.	Bajo	Media
11	Solicitar certificación de nacimiento	Permitirá al usuario solicitar certificación de nacimiento en Cuba.	Alta	Alta
12	Solicitar certificación de defunción	Permitirá al usuario solicitar defunción de nacimiento en Cuba.	Alta	Alta
13	Solicitar certificación de matrimonio	Permitirá al usuario solicitar defunción de matrimonio en Cuba.	Alta	Alta
14	Solicitar certificación de divorcio	Permitirá al usuario solicitar defunción de	Alta	Alta

		divorcio en Cuba.		
15	Solicitar consulta gratuita	Permitirá al usuario llenar un formulario con la consulta a realizar.	Alta	Alta
16	Notificar las respuesta emitidas por los especialistas	El especialista enviará la respuesta a la consulta realizada por el usuario.	Alta	Alta
17	Mostrar las preguntas de los ciudadanos y las respuestas de los especialistas	Se publicarán las consultas realizadas por los usuarios y sus debidas respuestas.	Alta	Alta
18	Buscar Información	Permitirá al usuario buscar información publicada en el sistema.	Baja	Baja
19	Mostrar cursos	Se mostrarán a los usuarios los cursos que están ofertando en ese momento.	Baja	Media
20	Solicitar cursos	El usuario podrá solicitar inscribirse en el curso que desee.	Baja	Media
21	Notificar respuesta de aceptación o no del curso	Se mostrará la información de los usuarios matriculados en los distintos cursos.	Baja	Media
22	Adicionar catálogo	Permitirá crear un registro en el catálogo.	Media	Alta
23	Mostrar catálogo	Permitirá mostrar los registros del catálogo.	Media	Alta

24	Editar catálogo	Permitirá modificar los registros del catálogo.	Media	Alta
25	Eliminar registro de la cesta	Permitirá eliminar los registros de la cesta.	Media	Alta
26	Mostrar cuantía de obsolescencia de las noticias creadas	Se mostrará que tan obsoletas son las noticias publicadas en el sistema.	Baja	Baja
27	Mostrar contador de tiempo en el formato (dd/mm/aa)	Se mostrará el día, mes y año de publicación de cada noticia.	Baja	Baja
28	Mostrar información sobre los bufetes que radican en cada provincia y municipios	Se mostrará la información sobre los bufetes que radican en cada provincia y municipios.	Baja	Media
29	Eliminar información de los bufetes que radican en cada provincia y municipios	Se podrá eliminar información sobre los bufetes que radican en cada provincia y municipios.	Baja	Media
30	Editar información de los bufetes que radican en cada provincia y municipios	Se podrá modificar la información sobre los bufetes que radican en cada provincia y municipios.	Baja	Media
31	Mostrar información de los abogados	Se mostrar información de los abogados con los que cuenta un bufete y que caso atienden en determinado momento.	Baja	Media

32	Adicionar noticia	El administrador podrá crear una noticia nueva en el sistema.	Baja	Alta
33	Editar noticia	El administrador podrá modificar una noticia ya creada.	Baja	Alta
34	Eliminar noticia	El administrador podrá eliminar una noticia ya creada.	Baja	Alta
35	Mostrar noticia	Se mostrarán las noticias en el sistema.	Baja	Alta
36	Adicionar recurso de casación	Se podrá registrar recursos de casación.	Alta	Alta
37	Adicionar escrito de oposición	Se podrá registrar escritos de oposición.	Alta	Alta
38	Solicitud de asesoría, representación y defensa	Se podrá crear el formulario del servicio de asesoría, representación y defensa.	Alta	Alta
39	Mostrar Políticas de privacidad	Mostrar Políticas de privacidad del sistema.	Baja	Baja
40	Mostrar Términos de uso	Mostrar Términos de uso del sistema.	Baja	Baja
41	Adicionar encuesta	Permitirá crear una nueva encuesta.	Baja	Baja
42	Mostrar encuesta	Permitirá mostrar las encuestas en el sistema.	Baja	Baja
43	Editar encuesta	Se podrá modificar las encuestas publicadas.	Baja	Baja

44	Eliminar encuesta	Se podrá eliminar una encuesta publicada.	Baja	Baja
----	-------------------	---	------	------

Anexo 4: Requisitos no funcionales

✓ **Apariencia o interfaz externa**

- El sistema debe estar diseñado para una resolución de 1024 x 768 píxeles o superior.
- El sistema debe estar estructurado en bloques con una correcta organización de la información.
- Deberá visualizarse de manera correcta en los navegadores: "Internet Explorer, Mozilla Firefox, Opera y Netscape Navigator".

✓ **Usabilidad**

- El sistema debe estar concebido para que sea utilizado por usuarios con conocimientos avanzados de informática y por usuarios inexpertos con conocimientos básicos, este aspecto se logra a partir de una correcta estructura de la información.

✓ **Soporte**

- Servidor Web Apache 2.2 o superior.
- Versión de PHP 5.x. o superior.
- Se requiere un servidor de bases de datos con soporte para grandes volúmenes de información y que permita una alta concurrencia.
- El sistema debe ofrecer la posibilidad de ser mejorado, así como de incorporarle nuevos servicios en caso de ser necesarios.
- En el lado del cliente se requiere un intérprete de aplicaciones web ("Internet Explorer, Mozilla Firefox, Opera, Google Chrome").

✓ **Estándares aplicables**

- El nombre de las funciones debe estar escrito de la siguiente forma:
nombre_del_módulo_nombre_de_la_función
- Cada sentencia condicional debe llevar su respectiva llave.
- Los arreglos deben ser formateados dejando un espacio entre cada elemento:
\$variable = array (elemento1, elemento2);

- Las especificaciones de las hojas de estilo que servirán de estándar a los navegadores se formulan a través del Consorcio W3C. Para garantizar un diseño correcto se hace imprescindible el uso de CSS en su versión 3 y las clases deben estar definidas con el siguiente formato:

.clase-definida {}

#identificador-definido {}

✓ **Portabilidad**

- El sistema debe ser multiplataforma.

✓ **Seguridad**

- El sistema debe reflejar los principios fundamentales de la seguridad informática: confidencialidad, integridad y disponibilidad de la información:
 - **Confidencialidad:** el acceso a la información debe ser solo para las personas autorizadas.
 - **Integridad:** la modificación de los datos la deben realizar las personas autorizadas.
 - **Disponibilidad:** las personas autorizadas deben tener acceso a la información solo en el momento requerido.
- El sistema debe incluir seguridad a nivel de usuarios, para garantizar la disponibilidad de la información.
- La administración del contenido debe realizarse únicamente por la persona que posea los permisos necesarios.
- Las funcionalidades del sistema tienen que mostrarse de acuerdo al rol del usuario que se encuentre autenticado.
- Otro mecanismo de seguridad son las consultas parametrizadas que define la capa de abstracción de base de datos de Drupal, que evitan los ataques de inyecciones SQL.
- El módulo **Database Logging** el cual se encarga de guardar registro de la actividad de los usuarios en el sistema, lo que permite auditar los fallos que ocurran en el sistema.
- Configuración de los permisos de uso de los filtros tales como el “PHP Filter” y el “Full HTML”, de esta manera se logra evitar también las inyecciones SQL.

✓ **Legales**

- Las licencias que se utilizarán son:
 - Licencia GNU/GPL para el CMS Drupal.
 - Licencia “Apache Software”.
 - Licencia “PHP License”.
- Todas consideradas licencias libres y flexibles aprobadas por la Fundación de Software Libre (“Free Software Foundation”).

✓ **Confiabilidad**

- Realizar salvallas periódicas para evitar que ocurra la pérdida de la información, ante cualquier situación.

Anexo 5: Descripción de los elementos de la estructura del sistema

- ✓ **Quiénes Somos:** se publica información referente a lo que representa la ONBC para las personas ya sean naturales o jurídicas.
- ✓ **Estructura:** permite conocer la estructura de la ONBC a varios niveles.
- ✓ **Misión, Visión y Objeto Social:** permitirá conocer la misión, la visión y objeto social de la ONBC.
- ✓ **Historia de la ONBC:** en este espacio se podrá conocer la historia de la ONBC.
- ✓ **Catálogo en línea:** en este espacio se podrá consultar la documentación de las diferentes ediciones de la ONBC.
- ✓ **Sitios informativos:** son sitios web que son de interés para la organización.
- ✓ **Sitios con documentación legal:** sitios que puede visitar el usuario para documentarse sobre procesos legales llevados en Cuba.
- ✓ **Autenticación de usuario:** muestra el formulario que permite al usuario autenticarse en el sistema.

- ✓ **Registro de usuario:** muestra el formulario que permite al usuario registrarse en el sistema.
- ✓ **Buscador:** permite realizar búsquedas de contenidos en el sistema.
- ✓ **Comentar:** permite al usuario comentar una noticia, imagen, etc.
- ✓ **Redes sociales (Facebook, Twitter):** permite al usuario establecer vínculos con diferentes redes sociales.
- ✓ **Contactos:** permite al usuario enviar (e-mail) un mensaje de contacto a algunas de las áreas de la ONBC.
- ✓ **Banner:** muestra todas las imágenes de banner una por una, de manera aleatoria.
- ✓ **Solicitud de certificación:** en esta sección el usuario puede solicitar una certificación, ya sea de nacimiento, matrimonio, divorcio y defunción.
- ✓ **Consulta gratuita:** en esta sección el usuario puede hacer alguna pregunta para que un especialista se la responda.
- ✓ **Bufete de Servicios Especializados:** en esta sección el usuario puede obtener y legalizar documentos que surtirán efectos en el extranjero.
- ✓ **Bufete Especializado en Recurso de Casación:** en esta sección el usuario puede registrar un recurso de casación y según el caso se registra el recurso de oposición.
- ✓ **Noticias:** muestra las noticias en la página principal.
- ✓ **Políticas de privacidad:** permite mostrar las políticas de privacidad del sistema.
- ✓ **Términos de uso:** permite mostrar los términos de uso del sistema.

Anexo 6: Preguntas de chequeo para la validación de los requisitos

¿Qué otros requisitos hacen referencia al requisito estudiado?

¿Están claramente identificados por medio de una matriz de referencias cruzadas o por cualquier otro mecanismo?

¿El requisito incumple alguna restricción definida?

¿El requisito es verificable? Si es así, ¿podemos efectuar pruebas (algunas veces llamadas criterios de validación) para verificar el requisito?

¿Se puede seguir el requisito en el modelo del sistema que hemos desarrollado?

¿Se puede localizar el requisito en el conjunto de objetivos del sistema/producto?

¿Está el requisito asociado con los rendimientos del sistema o con su comportamiento y han sido establecidas claramente sus características operacionales?

¿El requisito está implícitamente definido?

Anexo 7: Tareas de ingeniería

Tarea de ingeniería	
Número de tarea: 2	Nombre de historia: Gestionar usuario.
Nombre de la tarea: Garantizar el acceso de los usuarios.	
Tipo de tarea: Desarrollo.	Puntos Estimados(días): 1
Fecha inicio: 29/2/2014	Fecha fin: 30/2/2014
Programador responsable: Mirileydis Martínez Ajete	

Descripción: El usuario y contraseña son verificados para otorgarle los permisos según el rol que cumpla dentro de la aplicación.

Tarea de ingeniería

Número de tarea: 1

Nombre de historia: Catálogo en línea.

Nombre de la tarea: Mostrar las ediciones de la ONBC.

Tipo de tarea: Desarrollo.

Puntos Estimados(días): 4

Fecha inicio: 3/3/2014

Fecha fin: 7/3/2014

Programador responsable: Mirileydis Martínez Ajete

Descripción: El usuario visualizará lo(s) registro(s) publicados en el catálogo.

Tarea de ingeniería

Número de tarea: 3

Nombre de historia: Catálogo en línea.

Nombre de la tarea: Crear funcionalidad para eliminar registro(s).

Tipo de tarea: Desarrollo.	Puntos Estimados(días): 2
Fecha inicio: 8/3/2014	Fecha fin: 10/3/2014
Programador responsable: Mirileydis Martínez Ajete	
Descripción: El usuario podrá eliminar registro(s) añadidos en la cesta.	

Tarea de ingeniería	
Número de tarea: 2	Nombre de historia: Catálogo en línea
Nombre de la tarea: Crear funcionalidad Añadir a la cesta.	
Tipo de tarea: Desarrollo	Puntos Estimados(días): 2
Fecha inicio: 11/03/2014	Fecha fin: 13/03/2014
Programador responsable: Dayana Francia Ofarrill	
Descripción: El usuario podrá añadir a la cesta el o los registros del catálogo.	

Anexo 8: Descripción del caso de prueba de aceptación número 01 de la HU # 05

Caso de Prueba de Aceptación		
Código Caso de Prueba: 0501		Nombre Historia de Usuario: Servicio solicitud de certificación.
Nombre de la persona que realiza la prueba: Mirileydis Martínez Ajete		
Descripción de la prueba: se realiza la prueba a la funcionalidad Solicitar certificación de matrimonio.		
Condiciones de ejecución: El usuario tiene que estar registrado en el sistema y ser administrador de este.		
Entrada/Pasos de ejecución		Resultado esperado:
Entrada	Salida	
Se selecciona la opción Certificación de documentos y posteriormente la opción Certificación de matrimonio en cuba.		El sistema debe mostrar un formulario para introducir los datos de la certificación que desea hacer.
Se introducen los datos necesarios para la certificación.	Datos obligatorios y opcionales.	El sistema debe mostrar un mensaje indicando si la acción se ha realizado correctamente o no, en caso de ser negativa la respuesta del sistema, debe especificar en el mensaje los motivos por los que no se pudo realizar la acción.
Evaluación de la prueba: Satisfactoria.		

Anexo 9: Interfaz “Certificación de documentos”

[Inicio](#) » [Certificación de documentos](#)

Certificación de documentos

Obtención y Legalización 100% seguro de Certificados y Documentos en Cuba.

Ofrecemos el servicio de obtención y/o legalización de certificaciones y documentos de todo tipo en Cuba, además de realizarle cualquier tipo de subsanación de estos. Le realizamos el proceso de manera 100% segura y le enviamos todos los documentos hasta su casa. Contamos con un grupo de experimentados profesionales en todas las ramas del derecho que pueden atenderlo, evacuar sus dudas y ayudarlo a satisfacer sus necesidades, ya sean empresariales o personales, privadas o colectivas, propias o de terceros, actuales o futuras. Le aseguramos siempre una asesoría legal fácil y rápida a través de Internet. De esta manera Ud. siempre con nosotros ahorrará tiempo y dinero.

Certificaciones en línea:

[Certificación de Defunción en Cuba](#)

[Certificación de Divorcio en Cuba](#)

[Certificación de Nacimiento en Cuba](#)

[Certificación de Matrimonio en Cuba](#)

Anexo 10: Formulario para crear Certificación de matrimonio en Cuba

[Inicio](#) » [¿Que servicios brindamos?](#) » [Certificación de documentos](#) » [Certificación de Matrimonio](#)

Certificación de Matrimonio en Cuba

En Cuba se expiden Certificaciones de Nacimiento con toda la garantía que establecen las leyes nacionales e internacionales y que a su vez pueden ser legalizadas en el Ministerio de Relaciones Exteriores y en el consulado (en Cuba) del país donde resida para surtir efectos en el extranjero.

Ud. debe enviar los siguientes datos, para evaluar su solicitud de certificado de nacimiento.

Datos del solicitante de la certificación de matrimonio

Nombre(s) y apellidos del solicitante *

Correo electrónico *

Dirección particular del solicitante (no incluir ciudad, ni país, ni código postal) *

Ciudad *

País de residencia *

- Seleccionar - ▼

Código postal *

Datos de los contrayentes

Nombre(s) y apellidos de la contrayente *

Nombre(s) y apellidos del contrayente *

Fecha del matrimonio *

Día ▼ Mes ▼ Año ▼

Lugar del matrimonio *

Provincia del matrimonio *

- Seleccionar - ▼

Registro civil *

Tomo *

Folio *

Otros datos que puedan ser útiles

Trámites del documento

Obtención en el registro civil del certificado *

Obtener el certificado de matrimonio. ▾

Legalización en consulado en Cuba *

Si ▾

Especificar el consulado en Cuba *

- Seleccionar - ▾

Si la dirección de envío del certificado es distinta a la del solicitante especificarla a continuación:

El documento se utilizará fuera de Cuba *

Si ▾

País donde se utilizará el documento *

- Seleccionar - ▾

Legalización del certificado en el MINREX *

Sí ▾

Enviar solicitud

[Inicio](#) » [Certificación de Matrimonio en Cuba](#)

Certificación de Matrimonio en Cuba

Su formulario ha sido enviado correctamente. Gracias por confiar en nosotros.


[Volver al formulario](#)

Anexo 11: Resultado del caso de prueba 0401

The screenshot displays the website interface for the National Organization of Collective Law Offices. The header features the organization's logo and name, along with navigation links for 'Portada', 'Servicios', 'Certificación de documentos', 'Catálogo en línea', and 'Noticias'. The main content area shows a confirmation message: 'Ya ha enviado este formulario. Ver sus envíos anteriores.' Below this, the title 'Certificación de Matrimonio en Cuba' is displayed with several tabs: 'Vista', 'Editar', 'Formulario web', 'Índice', 'Resultados', and 'Seguimiento'. A paragraph explains that in Cuba, birth certifications are issued with national and international guarantees and can be legalized in the Ministry of Foreign Relations or the consulate of the country of residence. A note states: 'Ud. debe enviarnos los siguientes datos, para evaluar su solicitud de certificado de nacimiento.' Below this is a form field labeled 'Datos del solicitante de la certificación de matrimonio'. On the right side, there is a sidebar with a shopping cart containing one item, a search bar with a 'Buscar' button, and a list of notices including 'ABOGACÍA 2014' and 'Cursos'.

Panel de control Tienda Contenido Estructura Tienda Apariencia Personas Módulos Configuración CTools plugin example Informes
Advanced help Ayuda Example help Buscar 0 / 1 Bienvenido, **Administrador** Cerrar sesión
Imprimir Reenviar correos electrónicos

Información del envío
Form: [Certificación de Matrimonio en Cuba](#)
Enviado por **Administrador**
Jueves, 8 Mayo, 2014 - 11:30
127.0.0.1



Datos del solicitante de la certificación de matrimonio

Nombre(s) y apellidos del solicitante
mirileydis martinez ajete

Correo Electrónico
majete@estudiantes.uci.cu

Repetir Correo Electrónico
majete@estudiantes.uci.cu

Dirección Particular del solicitante (no incluir ciudad, ni país, ni código postal)
57102

Ciudad
.

Datos de los contrayentes

Nombre(s) y apellidos de la contrayente

mirileydis martina ajete

Nombre(s) y apellidos del contrayente

hamdy mohamed martinez

Fecha del matrimonio

27 Abril, 2014

Lugar del matrimonio

artemisa

Provincia del matrimonio

Artemisa

Municipio

Artemisa

Registro Civil

artemisa

Tomo

1,204

Folio

- -

Trámites del Documento

Obtención en el registro Civil del Certificado

Obtener el certificado de matrimonio.

Legalización en Consulado en Cuba

Sí

Especificar el Consulado en Cuba

Cuba

Si la dirección de envío del certificado es distinta al la del solicitante especificarla a continuación:

El documento se utilizará fuera de Cuba

Sí

País donde se utilizará el documento

Perú

Legalización del Certificado en el MINREX

Sí

[Envío Anterior](#) [Envío Siguiente](#)