

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**FACULTAD 4**



**Título:** Sistema para la gestión de documentos electrónicos correspondientes al área de Calidad de la Dirección de Producción Audiovisual (DPA), perteneciente al Centro de Tecnologías para la Formación (FORTES) de la Facultad 4.

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

***Autor(as):***

Zuleyda Navarro Pérez.

Anett Redondo González.

***Tutor:***

Ing. Luis Felipe Díaz Barrios.

***Co-tutor(es):***

Ing. Yordanis Rodríguez Rodríguez.

Ing. Yaylén Carrazana del Llano.

**LA HABANA, 2011**

**“AÑO DEL 53 ANIVERSARIO DEL TRIUNFO DE LA REVOLUCIÓN”**

## DECLARACIÓN DE AUTORÍA

---

Declaramos que somos las únicas autoras del presente Trabajo de Diploma que lleva como título: Sistema para la gestión de documentos correspondientes al área de Calidad de la Dirección de Producción Audiovisual (DPA), perteneciente al Centro de Tecnologías para la Formación (FORTES) de la Facultad 4 y autorizamos a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Firma del Autor  
Zuleyda Navarro Pérez

---

Firma del Autor  
Anett Redondo González

---

Firma del Tutor  
Luis Felipe Díaz Barrios

## DATOS DE CONTACTO

---

Ing. Luis Felipe Díaz Barrios, correo: [lf Diaz@uci.cu](mailto:lf Diaz@uci.cu). Graduado de Ingeniería Informática en el año 2010 en la Universidad de las Ciencias Informáticas (UCI). Actualmente se desempeña como desarrollador en el proyecto “Laboratorio para la Producción de Recursos Didácticos” perteneciente al Centro FORTES de la Facultad 4.

Ing. Yaylén Carrazana del Llano, correo: [yaylen@uci.cu](mailto:yaylen@uci.cu). Graduada de Ingeniería Industrial en el año 2004 en la Universidad de Holguín “Oscar Lucero Moya”. Actualmente se desempeña como jefa del área de Calidad de la DPA del Centro FORTES de la Facultad 4.

## OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado **Sistema para la gestión de documentos electrónicos correspondientes al área de Calidad de la Dirección de Producción Audiovisual (DPA)**, perteneciente al **Centro de Tecnologías para la Formación (FORTES) de la Facultad 4**, fue realizado en la **Universidad de las Ciencias Informáticas (UCI)**. El área de Calidad de la DPA considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente.

Parcialmente en un \_\_\_\_ %.

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

- Herramienta que facilita la rapidez y mejora la calidad de los procesos internos que se llevan a cabo durante el manejo de los documentos en la entidad.
- Interacción centralizada con los documentos y trabajadores de la institución contribuyendo a la disminución de la demora en la gestión de los documentos.
- Consulta y búsqueda de documentos en un tiempo mínimo.
- Extensibilidad para nuevas funcionalidades que permitan realizar mejoras futuras al sistema.
- Al constituir un software libre, no se incurre en costo alguno para el aprovechamiento de la solución.

Y para que así conste, se firma la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_

Yaylén Carrazana del Llano

Representante de la entidad

Jefa del área de Calidad de la DPA

Cargo

Firma

Cuño



**“El futuro de Cuba tiene que ser necesariamente un futuro de hombres de ciencia, de hombres de pensamiento.”**

**Fidel Castro Ruz**

## DEDICATORIA

---

*Zuleyda Navarro Pérez:*

*A mi mamita linda por brindarme todo su amor incondicionalmente y no abandonarme nunca en estos años de mi carrera.*

*A mi papito por ser tan preocupado por mis estudios y creer siempre en mí.*

*A mi segundo papito Yaya por ser tan especial.*

*Anett Redondo González:*

*A la persona que con su amor, confianza y sabiduría me ha hecho ser quien soy hoy, todo te lo debo a ti, mi mamita, gracias por permitirme ser la persona más dichosa del mundo al tenerte como madre.*

*A toda mi familia, en especial a mi tía Ildelisa y mi primo Georvanys por el apoyo y cariño brindados y a mi padrastro Nelso, pues ha sabido ser un padre para mí.*

*A mis abuelos porque sé que donde quiera que estén estarán muy orgullosos de mí.*

*A todos ellos, a Fidel y a la Revolución va dedicado nuestro trabajo.*

## AGRADECIMIENTOS

*Un especial agradecimiento a las personas que me dieron la vida, que me inculcaron los valores de honestidad, honradez, sencillez, responsabilidad... mi mamita Zulema y mi papito Gabriel, los quiero con la vida, gracias a ustedes soy quien soy, ustedes son mi faro guía y lo más grande y especial que tengo en mi vida, los amo muchísimo, si me faltaran les juro que no sabría qué hacer. Gracias a los dos por su apoyo incondicional y hacerme creer siempre que sí se puede, sé que ustedes también me quieren como yo, y soy muy feliz por eso.*

*Agradezco también a mi segundo papito Yayi gracias por estar siempre con mi mamita en estos años lejos de ella, aunque no te lo diga debes saber que te adoro y ocupas un lugar muy especial en mi corazón.*

*A Marifolis, Anita, Gabriela y María mi familia de Rolo las quiero mucho.*

*A mi abuelita Nena, a Adriancito, a mi tía Magdalenita, a mi prima Ailín, en fin a toda mi familia gracias por ser parte de mi vida.*

*A mis vecinas Eliza, Marcy y Ángela por preocuparse por mí.*

*A mis amigas de Moa Niurkis, Milagros, Maidelis, Yamilet son muy especiales las 4 y las quiero a todas tal y como son, hemos compartido momentos muy especiales, a pesar de que hemos tomado caminos diferentes nunca las olvidaré y siempre las tengo presente.*

*A mis primitas Ana Delia y Yara las quiero incondicionalmente.*

*A mis compañeros del 8101, 8202, 8307, 8405 y 4505, a todos los que aportaron un granito de arena en la realización de esta tesis, a Leonardo Alexander, a mi novio Roney, gracias por soprtarme como soy, por acompañarme y brindarme tu apoyo en esta etapa tan difícil de mi vida.*

*A las niñas del apartamento 142 303 Yane, Lia, Jessy, Lisett, Ana y en especial a Idalmita o Nani como la llamamos cariñosamente, gracias por las comiditas eres una amiga muy buena y especial a la que se llega a querer en muy poco tiempo, te quiero mimar.*

*También a los niños del 142 304 con los que he compartido momentos muy especiales. A mis mejores amigos desde primer año Ancel y Tinito, los adoro.*

*Y por supuesto a mis amigas incondicionales Mily, Clary y Yoly que estamos juntas desde primer año en las buenas y en las malas, también las quiero con el corazón, a Rosy que fue incorporada al clan en tercero.*

*A mi compañera de tesis y mejor de las mejores amigas Anett, que nunca, nunca nos hemos separado a lo largo de estos cinco años, te deseo lo mejor del mundo, eres muy especial y la vida sabrá recompensarte por eso, te quiero mucho, mucho.*

*Al tribunal por su exigencia, sus consejos y sus recomendaciones, lo que permitió que este trabajo de diploma se realizara con la calidad requerida, al tutor, al oponente, a todos los profesores que han aportado un granito de arena a lo largo de toda mi vida estudiantil en fin a todos los que han hecho posible que llegara este momento tan importante de mi vida.*



## AGRADECIMIENTOS

*A mi mamita de mi alma Maribel González López, porque has inundado mi mundo con tu amor incondicional, porque has guiado mis pasos hacia el bienestar y la felicidad, haciendo de mí lo que soy hoy, gracias por estar siempre ahí cuando lo necesité, gracias por las noches de desvelo y preocupación, porque haces tuyos mis triunfos y derrotas, por creer fielmente que siempre saldré adelante. Te quiero mucho.*

*A mi tía Ildelisa y mi primo Georvanys, por preocuparse siempre por mí, por estar en las buenas y en las malas, ofreciéndome sabios consejos cuando los necesito, por velar por mi salud y bienestar, por ser como son.*

*A Nelso, porque has sido un padre para mí, gracias por tu apoyo y cariño incondicional.*

*Al resto de mi familia porque en un momento u otro me han brindado su ayuda, y a mis vecinos porque siempre se han preocupado por mí.*

*A mis amistades de siempre, por crecer juntos y enfrentar los problemas a mi lado: Dayana, Zuremis, Dairelís y Anniel. A las del preuniversitario Betty, Rebeca, Darlys y Luis Carlos, porque fueron tiempos difíciles y me alentaron a salir adelante, por las noches de estudio y también de karaokes, por su amistad que jamás podré olvidar.*

*A mis amigas de esta vida loca de universidad: Zuleyda, Mileydís, Claribel, Rosana, Idalmis, Yanetsi y Yoslane, porque han sido las hermanas que siempre quisiera tener conmigo, gracias por tantos momentos lindos vividos, por acompañarme en las noches de estudio y de locuras y por cuidarme siempre, en especial a ti Zuly por ser mi mejor amiga desde primer año, por brindarme tu hombro para desahogarme cuando me hacía falta, por ser mi co-piloto preocupada en tareas extraclases, seminarios, incluso hasta en la realización de esta tesis.*

*A todas mis compañeras del apartamento 142 303 y a los locos del 142 304 por alegrarme cada día con sus ocurrencias y escándalos, porque han sido mi familia en la universidad.*

*A Agustín y Ancel por ser mis amigos incondicionales, porque conforman el Dream Team que siempre llevaré en un rinconcito especial de mi corazón.*

*A todos mis compañeros de los grupos 8101, 8202, 8307, 8405 y 4505, a los que están y a los que no, a todos por enseñarme que con la voluntad de hacer las cosas bien siempre se sale adelante. A Leonardo Alexander por las palabras de aliento, ayuda y amistad brindadas.*

*A mis maestros y profesores a lo largo de mi vida, porque cada cual aportó su pedacito en mi formación. Al tribunal, al tutor y al oponente porque con su trabajo han hecho posible que esta investigación resultara provechosa. A todos aquellos que de una forma u otra han contribuido a hacer realidad este sueño.*



El área de Calidad de la Dirección de Producción Audiovisual (DPA), perteneciente al Centro de Tecnologías para la Formación (FORTES) de la Facultad 4 requiere un sistema que gestione los documentos que se manejan en la misma, debido a que en la actualidad se dificulta la interacción y el control de dichos documentos. Por estas razones la presente investigación se centró en el desarrollo de un sistema encargado de la gestión y control de los documentos pertenecientes a dicha área.

Se realizó un estudio de diferentes herramientas gestoras de documentos en los ámbitos internacional, nacional y universitario, las cuales no cumplen con las demandas especificadas por los clientes. Además se efectuó un análisis de las diferentes tecnologías y herramientas posibles a utilizar, escogiendo a Drupal como sistema gestor de contenidos, NetBeans como entorno de desarrollo integrado, Visual Paradigm como herramienta de modelado y Rational Unified Process como metodología para guiar el proceso de desarrollo de la aplicación atendiendo a los distintos flujos de trabajo que propone.

La aplicación de la metodología escogida permitió la obtención de los requisitos funcionales y no funcionales, la modelación de los diagramas de clases del sistema, del análisis y del diseño, la definición de la estructura de la base de datos, la implementación para el desarrollo de la aplicación y la realización de pruebas para validar el sistema comprobando que cumple con todos los requisitos plasmados en sus inicios.

<b>INTRODUCCIÓN:</b> .....	1
<b>CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA:</b> .....	6
1.1- Introducción. ....	6
1.2- Sistemas de gestión documental. ....	6
1.2.1- Definiciones. ....	6
1.2.2- Ventajas de los sistemas de gestión documental. ....	7
1.2.3- Sistemas de gestión documental en el ámbito internacional.....	7
1.2.3- Sistemas de gestión documental en el ámbito nacional. ....	9
1.2.4- Sistemas de gestión documental en la UCI. ....	10
1.2.5- Sistema de gestión documental a desarrollar.....	11
1.3- Metodologías de desarrollo.....	12
1.3.1- Metodologías tradicionales. ....	12
1.3.2- Metodologías ágiles. ....	15
1.3.3- Justificación de la metodología escogida. ....	16
1.4- Herramientas de desarrollo.....	17
1.4.1- Herramientas de modelado.....	17
1.4.2- Justificación de la herramienta de modelado escogida.....	19
1.4.3- Sistemas gestores de base de datos. ....	19
1.4.4- Justificación de la herramienta gestora de base de datos escogida.....	21
1.4.5-Entornos de desarrollo integrado (IDEs).....	22
1.4.6-Justificación del IDE escogido.....	23
1.5 - Tecnologías de desarrollo.....	23
1.5.1- Sistemas gestores de contenidos (CMS). ....	23
1.5.2- Lenguajes de programación a utilizar. ....	25
1.5.3- Servidores web. ....	28
1.5.4- Justificación de las tecnologías escogidas.....	29

<b>CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA:</b> .....	30
2.1- Introducción. ....	30
2.2- Modelo del negocio. ....	30
2.2.1- Actores del negocio.....	30
2.2.2- Casos de uso del negocio (CUN). ....	31
2.2.3- Trabajadores del negocio.....	31
2.2.4- Entidades del negocio. ....	31
2.2.5- Descripción del CUN.....	32
2.2.6- Diagrama de CUN. ....	33
2.2.7- Diagrama de clases del modelo de objetos del negocio.....	33
2.3 - Propuesta de solución. ....	33
2.4- Requerimientos del sistema.....	34
2.4.1- Requerimientos funcionales (RF). ....	35
2.4.2- Requerimientos no funcionales (RNF). ....	36
2.5- Modelo del sistema.....	39
2.5.1- Actores del sistema.....	39
2.5.2- Casos de uso del sistema (CUS).....	41
2.5.3- Diagrama de CUS. ....	42
<b>CAPÍTULO 3- ANÁLISIS Y DISEÑO:</b> .....	44
3.1- Introducción. ....	44
3.2- Modelo de análisis. ....	44
3.2.1- Diagramas de clases del análisis.....	44
3.3- Arquitectura del CMS Drupal. ....	45
3.4- Modelo de diseño. ....	46
3.4.1- Patrones de diseño. ....	46
3.4.2- Diagramas de clases del diseño con estereotipos web. ....	47
3.4.3- Diseño de la base de datos. ....	50

3.4.4- Diagrama de clases persistentes.....	50
3.4.5- Modelo de datos.....	51
3.5- Prototipos de interfaz de usuario.....	52
<b>CAPÍTULO 4- IMPLEMENTACIÓN Y PRUEBA:</b> .....	54
4.1- Introducción.....	54
4.2- Modelo de implementación.....	54
4.2.1- Diagrama de despliegue.....	54
4.2.2- Diagrama de componentes.....	55
4.3- Estándares de codificación.....	56
4.3.1- Estándar de codificación para la base de datos.....	56
4.3.2- Estándar de codificación para PHP.....	56
4.4-Modelo de prueba.....	57
4.4.1- Registro de defectos y dificultades detectadas.....	58
<b>CONCLUSIONES GENERALES:</b> .....	62
<b>RECOMENDACIONES:</b> .....	64
<b>REFERENCIAS BIBLIOGRÁFICAS:</b> .....	65

## **INTRODUCCIÓN:**

El hombre, como ser cognitivo, desde que comenzó a manipular la información de los procesos que acontecían a su alrededor comprendió la necesidad de guardar, realizar copias, modificar y eliminar la misma; en sus inicios, se recurrió al formato duro para preservarla, dando paso a la gestión documental la cual alcanzó gran aplicación en las distintas empresas debido al continuo aumento de documentos referentes a los distintos procesos llevados a cabo en las mismas.

Con el surgimiento de las denominadas Tecnologías de la Información y las Comunicaciones (TIC) se comienza a sustituir los documentos en formato duro por los documentos electrónicos y con el auge de dichas tecnologías cada día se generan, reciben, envían y almacenan documentos con este tipo de formato en un número cada vez mayor de empresas de toda clase en el mundo. Esto ha hecho necesario la creación de áreas de trabajo que se ocupen fundamentalmente del manejo de los documentos.

Con este objetivo la industria informática ha desarrollado sistemas para la gestión de documentos electrónicos cuyo progreso ha tenido un auge notable en el desarrollo de las tecnologías en la última década. Específicamente en Cuba las TIC han tenido un gran impacto en todas las esferas y una de ellas es la esfera empresarial en la cual se ha tomado un papel protagónico en el uso de estas nuevas tecnologías.

Existe un gran número de instituciones encargadas de informatizar los diferentes procesos llevados a cabo en una empresa; la Universidad de las Ciencias Informáticas (UCI) como única en su tipo, se ha integrado a la informatización de estos procesos, para ello se han creado varios Centros de desarrollo que pertenecen a cada una de las facultades, los cuales son los responsables de desarrollar tecnologías que permitan ofrecer servicios y productos para la implementación de soluciones informáticas.

La Facultad 4 cuenta con uno de estos centros de desarrollo denominado Centro de Tecnologías para la Formación (FORTES) del cual forma parte la Dirección de Producción Audiovisual (DPA), que tiene como objetivo la edición y/o grabación de DVDs (Digital Versatile Disc traducido al español Disco Versátil Digital) de contenido pedagógico e histórico fundamentalmente, respondiendo a las necesidades de clientes tanto nacionales como internacionales.

El proceso de edición y/o grabación de los DVDs sigue la serie de pasos siguiente: captura de requisitos, planificación, revisión primaria de medias, diseño, edición- codificación de video, edición de sonido,

subtitulaje, autoría, visionaje y masterización, dichos pasos tienen asociado varios documentos que los rigen y que llevan la información de los resultados obtenidos.

La DPA cuenta con un área de Calidad responsable de verificar que el proceso de edición y/o grabación de un DVD se realice según lo establecido y cumpliendo las normas de calidad.

Actualmente no se encuentra automatizada la gestión documental en dicha área, por lo que se dificulta la interacción y el trabajo con la información recogida en los documentos producidos en el proceso descrito anteriormente, además el proceso de seguimiento y control de esta información resulta engorroso al carecer de una solución que lo optimice. Asimismo cada trabajador es el encargado de gestionar sus propios documentos, ya que no existe un lugar donde permanezca centralizada toda la información referente al flujo de trabajo del control de la calidad de los productos, igualmente la comunicación entre los trabajadores se realiza por vía e-mail lo que trae consigo que el proceso de modificación o de solicitud de consulta de un documento resulte lento y deficiente.

Se evidencian las limitaciones siguientes: se carece de una notificación cuando son actualizados los documentos, para saber cuál es la última versión existente de los mismos es necesario consultar al responsable de esta versión, no existen restricciones de acceso para los documentos por lo que cualquier trabajador puede modificarlos aunque no esté autorizado, la localización y/o búsqueda de un documento específico se torna tediosa por encontrarse dispersa la información. Por estas razones se producen deficiencias e inconformidades cuando se prueba y/o entrega el producto final.

Teniendo en cuenta lo anteriormente planteado el **problema a resolver** queda formulado de la siguiente manera: La insuficiente gestión de los documentos electrónicos correspondientes al área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4 provoca que la información esté descentralizada, desactualizada y de difícil acceso para el trabajo con la misma.

La presente investigación tiene como **objeto de estudio** el proceso de gestión de documentos electrónicos en la UCI.

El **objetivo general** de esta investigación está encaminado a: desarrollar un sistema para la gestión de documentos electrónicos correspondientes al área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4.

El **campo de acción** se define como el proceso de gestión de documentos electrónicos correspondientes al área de Calidad de la DPA, perteneciente al Centro FORTES de la Facultad 4.

Para complementar el objetivo general planteado anteriormente surgen los siguientes **objetivos específicos**:

- Efectuar el estudio del estado del arte sobre la gestión documental, para definir posición de las investigadoras.
- Realizar el proceso de análisis del sistema de gestión documental.
- Diseñar el sistema de gestión documental.
- Implementar la aplicación.
- Realizar pruebas al sistema.

Los objetivos planteados se cumplirán mediante las siguientes **tareas** a realizar:

- Estudio de las deficiencias existentes en la gestión de documentos del área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4.
- Identificación y análisis de las tendencias actuales sobre los sistemas de gestión de documentos a nivel internacional, nacional y en la UCI.
- Análisis y elección de la metodología que guiará el proceso de desarrollo de la aplicación.
- Selección de la herramienta de modelado para el diseño técnico del sistema.
- Identificación y selección de las herramientas y tecnologías para la implementación del sistema.
- Capacitación personal en las tecnologías para implementar el sistema.
- Realización de entrevistas al cliente para saber cómo quiere que funcione el sistema a desarrollar.
- Identificación y especificación de los requerimientos del sistema.
- Modelado de los artefactos del análisis del sistema.
- Definición de la arquitectura del sistema.
- Modelado de los artefactos del diseño del sistema.
- Diseño de una interfaz amigable para los usuarios utilizando CMS (Content Management System, sistema gestor de contenidos en español).
- Implementación de las funcionalidades generales del sistema.
- Implementación de un sistema de búsqueda.

- Adaptación del foro del CMS escogido a las características de la aplicación.
- Implementación del control de versiones de los documentos.
- Realización de pruebas a la aplicación.
- Redacción del informe de tesis.

Para desarrollar la investigación se plantea la siguiente **idea a defender**: con la realización de un sistema de gestión documental para el área de Calidad de la DPA, perteneciente al Centro FORTES de la Facultad 4 se tendrá centralizada toda la información permitiéndose un mejor acceso a la misma, facilitando la realización eficiente de los procesos llevados a cabo en dicha área, lo que posibilita que el producto final resulte con la calidad requerida atendiendo a las necesidades del cliente.

Para el desarrollo de la investigación se utilizaron algunos métodos tradicionales investigativos tanto teóricos como empíricos, a continuación se menciona cada uno de ellos y cómo se ponen de manifiesto en la investigación:

### **1- Métodos Teóricos:**

- **Analítico-Sintético:** Este método permite estudiar la influencia de cada factor en particular que tributa a que exista una ineficiente gestión documental en el área de Calidad de la DPA, perteneciente al Centro FORTES de la Facultad 4, a su vez posibilita descubrir las relaciones que guardan entre sí dichos factores. Además a través del mismo se realiza un análisis de los principales conceptos, metodologías, tecnologías y herramientas posibles a utilizar en el desarrollo de la investigación.
- **Histórico-Lógico:** Mediante este método se estudia la teoría y los conceptos sobre la gestión documental y los sistemas de gestión de documentos electrónicos a nivel internacional, nacional y de la UCI, además de las tecnologías y herramientas para su posterior selección, entiéndase estos como lenguajes de programación, CMS, sistemas gestores de base de datos (SGBD), así como la investigación de la metodología de desarrollo a seguir.
- **Modelación:** Con la aplicación de este método en la investigación se realiza un análisis de la realidad mediante diversos modelos y diagramas que permiten reflejar las necesidades de los clientes logrando un mejor entendimiento y comprensión del problema a resolver en toda su amplitud, para ello se utiliza una herramienta de modelado que facilita el proceso de modelación.

## **2-Métodos Empíricos:**

- **Observación:** A través de este método es posible recopilar una serie de datos confiables que ayudan a la comprensión del fenómeno y a la definición de lo primordial de la problemática, al mismo tiempo esto posibilita el planteamiento del problema permitiendo enmarcar el objeto de estudio y el campo de acción lo que propicia enfocar la investigación hacia lo que se necesita alcanzar y cómo alcanzarlo.

El contenido del presente trabajo de diploma está estructurado por 4 capítulos:

En el **Capítulo 1** se hace referencia a la fundamentación teórica donde se realiza un estudio de los sistemas de gestión de documentos electrónicos existentes y las principales tendencias actuales en los mismos, así como de las herramientas, las tecnologías, lenguajes de programación y la metodología a utilizar en el desarrollo del sistema.

En el **Capítulo 2** se hace referencia a las características del sistema donde se describen los procesos identificados en el negocio, así como el modelo de negocio correspondiente. Se definen los requisitos funcionales y no funcionales que debe cumplir el sistema, se identifican los actores y casos de uso del mismo realizándose una descripción de estos últimos, además se elabora el diagrama de casos de uso del sistema.

En el **Capítulo 3** se abordan los principales aspectos referentes al análisis y diseño de la aplicación y se obtienen una serie de artefactos que brindan una visión detallada del sistema y una arquitectura más robusta. Estos constituyen la base para una posterior implementación.

El **Capítulo 4** está destinado a la implementación y prueba de la aplicación mediante los resultados obtenidos en el análisis y diseño del sistema se desarrollará la aplicación generando a su vez diferentes artefactos que tributarán a la documentación del proceso, además se realizarán varias pruebas a la aplicación en vistas a comprobar su calidad y eficiencia.

# CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

---

## **CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA:**

### **1.1- Introducción.**

La gestión documental abarca el ciclo de vida completo de los documentos, es decir, constituye el tratamiento secuencial y coherente que se da a los documentos desde que se producen o reciben hasta el momento en que son eliminados o conservados, en función de su valor testimonial o histórico como fuente para el conocimiento (1).

Con el auge de las TIC surgieron los sistemas de gestión documental los cuales optimizaron la gestión de los documentos como se explicará en el epígrafe siguiente.

### **1.2- Sistemas de gestión documental.**

#### **1.2.1- Definiciones.**

Según el autor Ramón J. Fernández los sistemas de gestión documental son programas de ordenador creados para gestionar gran cantidad de documentos de todo tipo, el uso de estos sistemas nos permite administrar el flujo de los mismos en una organización, recuperar información desde ellos, determinar el tiempo que los documentos deben guardarse, eliminar los que ya no tienen ninguna utilidad y asegurar la conservación indefinida de los documentos más valiosos, aplicando principios de racionalización y economía (2).

En el sitio Advice-Business se especifica que un sistema de gestión de documentos también conocido como Document Management System (DMS, en español sistema de administración de documentos) es un sistema de computadoras que se utiliza para almacenar y rastrear los documentos de naturaleza electrónica (3).

A juicio de las autoras de la presente investigación un sistema de gestión documental es un programa informático creado con el fin de optimizar los procesos de creación, modificación y eliminación de documentos procurando que esta gestión esté guiada a mantener el control de los mismos, por lo que constituye una disminución de espacio, tiempo y gastos económicos con respecto a la gestión documental tradicional, entiéndase por esto: gestión de documentos en formato duro.

Teniendo en cuenta lo planteado anteriormente se puede afirmar que los sistemas de gestión documental presentan notables e importantes ventajas las cuales serán detalladas en el subepígrafe siguiente.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.2.2- Ventajas de los sistemas de gestión documental.**

Un sistema de gestión documental se ocupa del procesado, almacenamiento, búsqueda, recuperación y distribución de documentos al conjunto de usuarios que operan en el mismo. De entre las ventajas que aportan estos sistemas a una organización destacan las siguientes (4):

- Reducción de costes de los procesos empresariales, mediante el rediseño de procesos, sustitución del trabajo administrativo no productivo y reducción del espacio físico de almacenamiento.
- Disminución de los ciclos de trabajo, aumentando la concurrencia de las distintas actividades necesarias.
- Unificación de los procesos empresariales en los distintos ámbitos departamentales y geográficos, potenciando los canales formales y los procedimientos de trabajo, lo que facilitará el cumplimiento de los requerimientos que los sistemas de calidad imponen.
- Aumento de las capacidades de comunicación en toda la organización, mejorando la integridad y seguridad de la información.

Los sistemas de gestión de documentos electrónicos tienen gran aplicación en diferentes ámbitos manteniendo características comunes así como desarrollando distintas particularidades como se evidencia en los siguientes subepígrafes.

### **1.2.3- Sistemas de gestión documental en el ámbito internacional.**

Los sistemas de gestión documental han alcanzado un auge significativo para el mejoramiento de la administración documental de las empresas, es por ello que internacionalmente se han desarrollado un sinnúmero de sistemas de este tipo que han tenido participación en este sentido, por ejemplo:

1. Uno de los sistemas de gestión documental utilizado internacionalmente es Orfeo, desarrollado por la Superintendencia de Servicios Públicos Domiciliarios (SSPD) en Colombia y licenciado como software libre bajo licencia GNU's not Unix (GNU, en español GNU no es Unix)/ General Public License (GPL, en español Licencia Publica General) para compartir el conocimiento y mantener la creación colectiva libre (5).

Principales características de Orfeo (6):

- Interfaz gráfica fácil de usar.
- Gestión sobre el documento: reasignar, modificar y archivar.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

- Agiliza las búsquedas de documentos por diversos campos de búsqueda (fechas, usuarios, tipos de documentos, etc.)
- Puede instalarse en cualquier sistema Operativo (GNU/Linux, Unix, Windows, etc).
- Soporta diferentes bases de datos.
- Maneja tipos de formatos variados.

2. Quipux también está entre los sistemas de gestión de documentos electrónicos usados en el mundo, este sistema fue modificado a partir del sistema de gestión documental Orfeo, el cual utiliza tecnologías y estándares abiertos. La subsecretaria de informática de Venezuela efectuó modificaciones a la versión original adaptándolas a las necesidades de gestión documental de las entidades de la Administración Pública Central (7).

Quipux, gestiona la documentación digital y/o impresa en el interior de una entidad, entre entidades y la enviada por la ciudadanía a las instituciones. A continuación se especifican las funcionalidades generales del sistema (7):

- Creación, envío, recepción, almacenamiento y clasificación de memorandos, oficios, circulares y anexos.
- Búsqueda, recuperación y presentación de documentos.
- Organización y clasificación de documentos digitales en carpetas o expedientes virtuales.
- Reportes estadísticos de documentos creados y enviados, tramitados, pendientes, archivados.
- Almacenamiento permanente y transferencia de archivos o expedientes.
- Creación compartida de documentos.
- Seguridad y auditoría a través de usuarios, perfiles y archivos de auditoría.

Los sistemas de gestión documental no solo han tenido auge a nivel internacional sino que también han alcanzado gran aceptación en Cuba, en ocasiones han sido creados partiendo desde cero y en otras a través de adaptaciones a los existentes en el ámbito internacional teniendo en cuenta las particularidades de las empresas cubanas, en el subepígrafe siguiente se explica con más detalles lo planteado anteriormente.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.2.3- Sistemas de gestión documental en el ámbito nacional.**

Durante la búsqueda bibliográfica realizada por las autoras para la presente investigación no se detectaron en nuestro país sistemas de gestión documental que explotaran todo lo que puede incluirse en la gestión de documentos, a pesar de esto si existen ciertas soluciones informáticas que se han dado a la tarea de implementar algunas de las facilidades que deben brindar las aplicaciones de este tipo:

1. Babel es un sistema automatizado de gestión de información para el control y gestión de las solicitudes de los servicios propios de la Unidad de Traducción del Centro de Información de la Empresa de Telecomunicaciones de Cuba S.A (ETECSA), este sistema tuvo sus inicios en el año 2003 como resultado de un proyecto de cooperación con la Facultad de Ingeniería Eléctrica de la Universidad de Las Villas (8).

Babel permite organizar, clasificar y almacenar todos los documentos traducidos, posibilita por una parte la gestión documental de los mismos, con el propósito de facilitar el acceso a la información a distintos usuarios (clientes o traductores) así como la utilización eficiente y segura de estos documentos (8).

2. El sistema de gestión de documentos electrónicos AvilaDoc fue desarrollado por la empresa Desoft de la provincia Ciego de Ávila sobre una plataforma de software libre con base de datos centralizada, destinado a la gestión, tramitación y resguardo de archivos electrónicos (9).

Es una herramienta de trabajo que gestiona el tramitado interno de los documentos a través de la intranet, sin necesidad de copias, o envíos físicos y permite el control de la entrada y salida de documentos, el manejo de anotaciones a los documentos por usuarios, el tratamiento diferencial para los documentos pendientes, así como la recuperación de la información a través del contenido de la ficha del documento. Además ofrece un historial del documento desde que fue creado hasta su eliminación. Conserva la acción que se hizo sobre el mismo, el usuario que la acometió y la fecha en que lo hizo (9).

Para la UCI también se ha hecho necesaria la utilización de soluciones informáticas que constituyan sistemas de gestión documental, debido fundamentalmente al creciente volumen de documentos que se manipulan en los proyectos, facultades y en la universidad en general, de esto se estará profundizando en el subepígrafe desarrollado a continuación.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.2.4- Sistemas de gestión documental en la UCI.**

Alfresco es uno de los sistemas de gestión documental más conocido en el mundo, es la principal alternativa de código abierto para la gestión de contenidos empresariales (Enterprises Content Management, ECM en sus siglas en inglés). Se basa en Programación Orientada a Aspectos y está escrito en el lenguaje de programación Java (10).

Este sistema posee las siguientes características (11):

- Ofrece una gestión de documentos usando una interfaz familiar, lo cual provee facilidad de interacción con el usuario.
- Posee un eficiente control de versiones de los documentos manipulados.
- Permite registrar una serie de aspectos básicos en la vida de un documento: quién ha creado, quién ha actualizado, cuándo se ha creado, cuándo se ha actualizado.
- Posibilita las notificaciones de cambios por RSS (significado de sus siglas en inglés Really Simple Syndication, Sindicación Realmente Simple en español) y email.
- Para implementar la seguridad, posee gestión de usuarios, grupos y roles.
- Posee un repositorio centralizado y personalizable, lo cual provee flexibilidad e integración a los procesos en general.

En la UCI existe un flujo bastante elevado de documentación relacionada con infinidad de temas, es por ello que se desarrolla un proyecto en la Facultad 1 el cual seleccionó al sistema de gestión documental Alfresco y se encarga de adecuar sus funcionalidades a las necesidades de la universidad. Este proyecto se encuentra en su fase inicial de desarrollo y se tiene como objetivo lograr que sea capaz de gestionar los documentos de la UCI por entidades, cada entidad<sup>1</sup> necesitará realizar una solicitud cuando desee acceder y/o modificar algún documento, entonces el nuevo sistema será responsable de buscar dicho documento dónde se tenga almacenado y proporcionárselo a la entidad solicitante, no contempla privilegios múltiples de modificación<sup>2</sup> sobre un mismo documento.

---

<sup>1</sup> Facultad, proyecto, departamento, etc.

<sup>2</sup> Que varios usuarios puedan modificar un mismo documento.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.2.5- Sistema de gestión documental a desarrollar.**

Después de haber analizado los sistemas de gestión documental anteriores se puede arribar a la conclusión de que cada uno de ellos atendiendo a sus características tiene grandes aplicaciones y por tanto son usados por diferentes empresas a nivel internacional y nacional.

Atendiendo a los sistemas en el ámbito internacional se puede concluir que no se utiliza Orfeo ni Quipux porque no presentan en sus funcionalidades el control de versiones, ni el envío de reportes de modificación a los usuarios afectados por la misma, además no administran los permisos de acceso y modificación de documentos.

Con respecto a los sistemas de gestión documental utilizados nacionalmente que han sido referenciados en el subepígrafe anterior se puede ultimar que no se puede utilizar Babel ya que este solo se encarga de la gestión de los documentos y no realiza otras funcionalidades necesarias para el área de Calidad. En cuanto a AvilaDoc este implementa algunas funcionalidades necesarias para la solución que se propone desarrollar pero carece de un motor de búsqueda avanzada.

Específicamente en la UCI se desea utilizar Alfresco el cual tiene varias funcionalidades y características útiles para la gestión de disímiles documentos, sin embargo, al constituir un proyecto en implementación, ya que se pretende adecuar a las características de la UCI no se está utilizando todavía en la universidad. Esta es una de las razones por la cual no es factible utilizarlo en la presente investigación, pues el área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4 tiene la necesidad de utilizar lo antes posible un sistema que solucione las deficiencias existentes en su gestión documental y que responda a las especificidades que deben existir al gestionar los mismos.

Además está guiado a la gestión de los documentos a nivel de entidad, debido a que para trabajar con un documento la entidad correspondiente debe solicitar el mismo y el sistema Alfresco es el encargado de proporcionárselo, característica que modificaría la lógica del negocio en el área en cuestión y traería retrasos en el cumplimiento del flujo de trabajo del control de la calidad.

El área de Calidad realiza la gestión de manera muy específica ya que cada documento posee peculiaridades por las que debe ser gestionado de manera distinta al resto y solo por algunos usuarios de la institución, esto requiere de un sistema documental estrechamente guiado a satisfacer estas necesidades y que gestione a cada documento de los ocho tipos que se manejan en dicha área de manera distinta y específica, que permita la interacción dinámica con los documentos por parte de varios usuarios lo que no contempla el sistema Alfresco a implementar en la universidad.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

Para la elaboración del sistema de gestión documental del área de Calidad de la DPA, perteneciente al Centro FORTES de la Facultad 4 se utiliza una metodología de desarrollo, así como herramientas de modelado y de implementación que serán seleccionadas y fundamentadas en los epígrafes posteriores.

### **1.3- Metodologías de desarrollo.**

Según plantea Sergi Blanco Cuaresma la metodología de desarrollo constituye una colección de documentos formales referentes a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. La finalidad de una metodología de desarrollo es garantizar la eficacia y la eficiencia en el proceso de generación de software. El núcleo de cualquier metodología de desarrollo se encuentra constituido por documentos que detallan cada una de las fases de desarrollo (12).

A consideración de las autoras una metodología de desarrollo constituye una guía para la realización de un proyecto encaminada a ayudar a los desarrolladores, ofreciendo un conjunto de procedimientos, técnicas, y documentos que guardan un historial del proceso de desarrollo del software.

#### **1.3.1- Metodologías tradicionales.**

Las metodologías tradicionales son también denominadas metodologías clásicas, las cuales están guiadas por una fuerte planificación durante todo el proceso de desarrollo basadas en normas provenientes de estándares seguidos por el entorno de desarrollo, las mismas generan un gran número de artefactos<sup>3</sup> que registran lo que se ha desarrollado en las distintas etapas del software. Además estas metodologías ofrecen una cierta resistencia a los cambios, siguen un proceso controlado por numerosas políticas y contemplan un contrato prefijado sobre las funcionalidades que deberá poseer la solución informática dada. Cuentan a su vez con la peculiaridad de que el cliente interactúa con el equipo de desarrollo mediante reuniones en las cuales se evalúan los adelantos del software (13).

##### **1.3.1.1- Rational Unified Process (RUP en español Proceso Unificado Racional).**

RUP es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML, y trabajo de muchas metodologías utilizadas por los clientes. Fue creado por Software Rational Corporación, una división de IBM desde 2003 (14).

El ciclo de vida de RUP se caracteriza por:

---

<sup>3</sup> Documentos, diagramas, modelos, etc.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

**Dirigido por Casos de Uso (CU):** Los CU reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los CU guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los CU (15).

**Centrado en la arquitectura:** La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura. El modelo de arquitectura se representa a través de vistas en las que se incluyen los diagramas de UML (15).

**Iterativo e incremental:** RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración (15).

Es práctico dividir el trabajo en partes más pequeñas o miniproyectos. Cada miniproyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Cada iteración se realiza de forma planificada es por eso que se dice que son miniproyectos (16).

### **Ventajas de RUP (16):**

- Es sencillo, ya que sigue los pasos intuitivos necesarios a la hora de desarrollar el software.
- Seguimiento detallado en cada una de las fases.
- No se necesita interacción con el cliente en todo el ciclo de desarrollo.
- Gran número de documentación que proporciona un control estricto de todo lo que pasó a lo largo del proyecto.
- Se define el alcance del proyecto desde un principio.

### **Desventajas de RUP (16):**

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

- La evaluación de riesgos es compleja.
- El cliente deberá ser capaz de describir y entender a un gran nivel de detalle para poder acordar un alcance del proyecto con él.

### **1.3.1.2- Microsoft Solution Framework (MSF en español Marco de Soluciones Microsoft).**

Se basa principalmente en los modelos espiral<sup>4</sup> y cascada<sup>5</sup> (hitos y fases). Fue desarrollado por Microsoft, tiene como principios fundamentales la comunicación (entre cliente/usuario y entre desarrolladores), una capacitación de las personas (disciplina de disponibilidad) es decir cumple con el proceso de formación de personal al compartir los roles entre todo el equipo de trabajo. MSF es un proceso donde se debe crear versiones para el negocio de cada cliente. Otro principio es la inversión de calidad (tiempo, trabajo, dinero), ya que cada proyecto es una inversión no un gasto (17).

MSF se compone de 2 modelos y 3 disciplinas. Modelo de Equipo y de Proceso. Disciplina de Administración de Proyecto, Administración de Riesgos y Administración de la Preparación. (17)

#### **Ventajas de MSF (18):**

- Tiene facilidad de soporte y mantenimiento.
- Está basado en el trabajo en equipo.
- Se puede utilizar para proyectos de cualquier magnitud.
- Crea una disciplina de análisis de riesgos que ayuda y evoluciona con el proyecto.
- Gracias a Microsoft, este modelo cuenta con plantillas que nos ayudan en el proceso de documentación.

#### **Desventajas de MSF (18):**

- Este modelo utiliza demasiada documentación en sus fases.
- El análisis de riesgo suele llevar mucho tiempo frenando el avance del proyecto.
- Al ser un modelo de Microsoft implica que se tiene que utilizar herramientas solo de Microsoft.

---

<sup>4</sup> El modelo espiral de los procesos software es un modelo del ciclo de meta-vida. En este modelo, el esfuerzo de desarrollo es iterativo. Tan pronto como uno completa un esfuerzo de desarrollo, otro comienza.

<sup>5</sup> Contempla el desarrollo de software a través de una secuencia simple de fases, donde cada fase tiene un conjunto de metas bien definidas.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### 1.3.2- Metodologías ágiles.

Cuando el desarrollo de un software se realiza de manera incremental<sup>6</sup>, cooperativa<sup>7</sup>, sencilla y adaptable<sup>8</sup> se está en presencia de una metodología ágil. Las metodologías ágiles se encuentran preparadas para cambios durante el proyecto, siguen un proceso menos controlado, con pocos principios y sin un contrato tradicional y de existir este es bastante flexible. Requiere que el cliente sea parte activa del equipo de desarrollo. En este tipo de metodologías son generados pocos artefactos (13).

#### 1.3.2.1- Scrum.

La metodología Scrum<sup>9</sup> tiene sus orígenes en el año 1986, en la misma se asume que el proceso de desarrollo de software es impredecible, y lo trata como a una caja negra controlada, en vez de manejarlo como un proceso completamente definido (19).

Scrum no trata el proceso de desarrollo de software como un proceso lineal, en el que se sigue la secuencia de análisis, diseño, codificación y prueba. En Scrum, el proyecto puede iniciarse con cualquier actividad, y cambiar de una a otra en cualquier momento. Un proyecto administrado mediante Scrum se organiza en iteraciones, llamadas sprints<sup>10</sup>, que normalmente tienen entre dos y cuatro semanas de duración. Al principio de cada sprint se establece una lista de requerimientos llamada backlog<sup>11</sup>, que debe completarse cuando éste finalice. A diario se realizan breves reuniones del equipo de desarrollo, en las que se exponen los avances y los problemas encontrados, y se señalan posibles caminos para resolverlos (19).

El proceso de desarrollo Scrum se compone de cinco actividades principales: revisión de los planes de release<sup>12</sup>, distribución, revisión y ajuste de los estándares de producto, sprint, revisión de sprint, y cierre (19).

---

<sup>6</sup> Entregas pequeñas de software, con ciclos rápidos.

<sup>7</sup> Cliente y desarrolladores trabajan juntos constantemente con una cercana comunicación.

<sup>8</sup> Permite realizar cambios de último momento.

<sup>9</sup> Hace honor a una estrategia de juego que se utiliza en el fútbol rugby.

<sup>10</sup> Término que denomina a una iteración que está acotada en el tiempo, durante la cual el equipo trabaja para convertir elementos del backlog del producto en un incremento del producto potencialmente productivo.

<sup>11</sup> Lista de tareas que define el trabajo a realizar por los miembros del equipo de Scrum durante un sprint. En español significa acumulación de trabajo.

<sup>12</sup> Revisión o versión.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.3.2.2- Extreme Programming (XP en español Programación Extrema).**

Surge como una nueva manera de encarar proyectos de software, proponiendo una metodología basada esencialmente en la simplicidad y agilidad. Es una de las llamadas metodologías ágiles de desarrollo de software más exitosas de los tiempos recientes. La metodología propuesta en XP está diseñada para entregar el software que los clientes necesitan en el momento en que lo necesitan. Además alienta a los desarrolladores a responder a los requerimientos cambiantes de los clientes, aún en fases tardías del ciclo de vida del desarrollo. La metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance. Uno de sus requerimientos es tener al cliente disponible durante todo el proyecto, formando parte del grupo de desarrollo (20).

**Desventajas:** Una de las críticas a XP es la dificultad de estimar cuánto va a costar un proyecto. Dado que el alcance del mismo no está completamente definido al comienzo, y que esta metodología es expresamente abierta a los cambios durante todo el proceso, por lo que se torna sumamente difícil estimar un presupuesto previo (20).

Otra crítica a XP está dada por su baja documentación, pensando en el posterior mantenimiento del sistema. Si bien durante el proyecto el equipo tiene en mente todas sus particularidades, hay que prever que pasará luego de entregado, cuando el equipo se disuelva y sea necesario realizar alguna mejora (20).

### **1.3.3- Justificación de la metodología escogida.**

La metodología que se selecciona es RUP, ya que al contar con una detallada documentación se tiene un control estricto de todo lo que pasa a lo largo del proyecto. También brinda la posibilidad de definir en edades tempranas del desarrollo del software el alcance del mismo. No es necesario llevar a cabo estrictamente todas las actividades y artefactos que describe dicha metodología, puesto que se recomienda que en dependencia de las características del proyecto y de la organización, se seleccionen los artefactos y actividades que van a ser utilizados.

Con esta metodología no se necesita una interacción constante con el cliente, es decir, no se debe contar con el tiempo que disponga el mismo, esta característica es muy importante debido a que los clientes de la aplicación a desarrollar no tendrán una interacción constante con las desarrolladoras ya que su demanda de trabajo no se lo permite. Además como el sistema a implementar es para el área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4 y las autoras de la presente investigación no

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

estarán en años posteriores en la Universidad es vital la selección de RUP como metodología de desarrollo ya que brinda suficiente información al cliente o incluso a un futuro equipo de desarrollo si se desea realizar cambios posteriores a la aplicación.

Las metodologías XP y Scrum no se utilizan puesto que generan una baja documentación lo que dificulta realizar cambios futuros a la aplicación, además requieren que el cliente forme parte del equipo de desarrollo, aspecto del cual no puede depender la presente investigación.

Por otra parte no se utiliza la metodología MSF porque al ser un modelo de Microsoft requiere la utilización de herramientas solo de Microsoft, lo que resulta un impedimento para la migración a software libre en la UCI.

### **1.4- Herramientas de desarrollo.**

#### **1.4.1- Herramientas de modelado.**

Las herramientas de modelado fueron creadas con el fin de desarrollar programas, utilizando técnicas de diseño y metodologías bien definidas, soportadas por herramientas automatizadas. Suelen llamarse herramientas CASE, significado de las siglas: Computer Aided Software Engineering, en español Ingeniería de Software Asistida por Computadora (21).

Es un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación (21).

Las autoras del presente trabajo de diploma sostienen que una herramienta CASE es un instrumento que permite desarrollar soluciones informáticas abarcando todo el ciclo de vida de un software<sup>13</sup>, por lo que constituye un apoyo para los desarrolladores del mismo.

##### **1.4.1.1- Visual Paradigm.**

Visual Paradigm for UML (VP-UML) es una herramienta de diseño Unified Modeling Language (UML) cuyo significado en español es Lenguaje Unificado de Modelado y además constituye una herramienta CASE diseñada para la ayuda al desarrollo de software. Es una herramienta UML profesional que soporta el ciclo de vida completo de desarrollo de un programa: análisis y diseño orientados a objetos, construcción,

---

<sup>13</sup> Programa.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación (22).

### **Características de Visual Paradigm (22):**

- Licencia: gratuita y comercial.
- Producto de calidad.
- Soporta aplicaciones web.
- Varios idiomas.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

### **1.4.1.2- Rational Rose.**

Rational Rose es una herramienta CASE que propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del mismo. Permite crear y refinar estas vistas estableciendo de esta forma un modelo completo que representa el dominio del problema y de las características del sistema (23).

Rational Rose utiliza un proceso de desarrollo iterativo controlado. Cada iteración comienza con una primera aproximación del análisis, diseño e implementación para identificar los riesgos del diseño, los cuales se utilizan para conducir la iteración, primero se identifican los riesgos y después se prueba la aplicación para que estos se hagan mínimos. Cuando la implementación pasa todas las pruebas que se determinan en el proceso, esta se revisa y se añaden los elementos modificados al modelo de análisis y diseño (23).

### **Características (23).**

- Capacidad para proporcionar el desarrollo iterativo.
- Mantiene la consistencia de los modelos del sistema.
- Chequeo de la sintaxis UML.
- Generación de código a partir de los modelos.
- Ingeniería Inversa (crear modelo a partir código).

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

- No es gratuito, se debe hacer un previo pago para poder adquirir el producto.
- Admite la integración con otras herramientas de desarrollo.

### **1.4.2- Justificación de la herramienta de modelado escogida.**

Para el modelado del sistema se utiliza como herramienta CASE a Visual Paradigm en su versión 6.4, puesto que es una herramienta potente y distribuida bajo una licencia gratuita y comercial, es fácil de instalar y actualizar permitiendo una excelente compatibilidad entre ediciones, lo que facilita el trabajo de modelado de la aplicación. La herramienta Rational Rose no se escoge puesto que no es libre ni gratuita.

### **1.4.3- Sistemas gestores de base de datos.**

Un SGBD es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de: un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos (24).

Debe permitir (24):

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

Las autoras de la presente investigación consideran que los SGBD constituyen programas que se encargan de controlar, definir, consultar, manipular y proteger los datos que se manejan en alguna aplicación.

#### **1.4.3.1- My Structured Query Language (MySQL en sus siglas en inglés, Mi Lenguaje de Consulta Estructurado en español).**

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL (presenta restricciones) de la GNU. Su diseño multihilo<sup>14</sup> le permite soportar una gran carga de forma muy eficiente. Este gestor de bases de datos es considerado el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Existen infinidad de librerías y otras herramientas que permiten su

---

<sup>14</sup> Ejecutar varias tareas de forma simultánea.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración (25):

Las principales características de este gestor de bases de datos son las siguientes (25):

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's (Application Program Interface en sus siglas en inglés) en varios lenguajes.
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un buen nivel de seguridad en los datos.

### **1.4.3.2- PostgreSQL.**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS en sus siglas en inglés) ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. Está basado en el proyecto POSTGRES, de la universidad de Berkeley, es una derivación libre (OpenSource<sup>15</sup>) de este proyecto. PostgreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos (25).

A continuación se enumeran las principales características de este gestor de bases de datos (25):

- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.
- Está bajo la licencia BSD (Berkeley Software Distribution), esta licencia tiene menos restricciones en comparación con otras como la GPL, estando muy cercana al dominio público. Permite el uso del código fuente en software no libre.
- Ejecutable en casi todos los sistemas operativos: Linux, Unix, BSDs, Mac OS, Beos y Windows.

---

<sup>15</sup> Código abierto.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

- Documentación muy bien organizada, pública y libre, con comentarios de los propios usuarios.
- Comunidades muy activas, varias comunidades en castellano.
- Altamente adaptable a las necesidades del cliente.
- Soporte nativo para los lenguajes más populares del medio: PHP, C, C++, Perl y Python.

### **1.4.3.3- Oracle.**

Oracle es básicamente una herramienta cliente/servidor para la gestión de bases de datos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de páginas web pasa lo mismo: como es un sistema muy caro no está tan extendido como otros SGBD (26).

Principales características (27):

- Es una herramienta de administración gráfica que es mucho más intuitiva y cómoda de utilizar.
- Ayuda a analizar datos y efectuar recomendaciones concernientes a mejorar el rendimiento y la eficiencia en el manejo de aquellos datos que se encuentran almacenados.
- Apoya en el diseño y optimización de modelos de datos.
- Apoya en la definición de estándares de diseño y nomenclatura de objetos.
- Permite documentar y mantener un registro periódico de las mantenciones, actualizaciones de hardware y software, cambios en las aplicaciones y, en general, todos aquellos eventos relacionados con cambios en el entorno de utilización de una base de datos.

### **1.4.4- Justificación de la herramienta gestora de base de datos escogida.**

Para el desarrollo del sistema no se utiliza el SGBD Oracle debido a que es completamente privativo con precios muy elevados reconocidos a nivel mundial. Tampoco se utiliza MySQL pues está bajo la licencia GPL la cual presenta restricciones ya que al modificarse y distribuirse debe mantener el mismo tipo de licencia así como el nombre y los créditos de los autores originales.

Se escoge la herramienta gestora de base de datos PostgreSQL ya que brinda ventajas sustanciales en el trabajo con la información pues ofrece una garantía en la integridad de los datos y una mejor escalabilidad y rendimiento bajo grandes cargas de trabajo, además porque es una herramienta libre (licencia BSD) y

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

atendiendo a la migración a software libre en la UCI es necesario que la solución sea implementada utilizando herramientas de este tipo, puesto que es de vital importancia para el país la completa y adecuada migración hacia estos ámbitos.

### **1.4.5-Entornos de desarrollo integrado (IDEs).**

#### **1.4.5.1- Zend Studio.**

Zend Studio es un editor de texto orientado a desarrollar aplicaciones web en lenguaje PHP. Este IDE consta de dos partes en las que se dividen las funcionalidades del lado del cliente y las del servidor. Zend Studio es un completo entorno de desarrollo integrado para el lenguaje de programación PHP. Está escrito en Java, y se encuentra disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux. Se puede decir que además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código (28).

#### **1.4.5.2- Eclipse.**

Eclipse es un IDE para Java muy potente. Es libre y fue creado originalmente por IBM<sup>16</sup>. No es tan sólo un IDE, se trata de un marco de trabajo modular ampliable mediante complementos (plugins). Existen complementos que nos permiten usar Eclipse para programar en PHP, Perl, Python, C/C++, entre otros lenguajes. Sin embargo presenta una interfaz y entorno gráfico complicado para los usuarios que vayan a interactuar con él (29).

#### **1.4.5.3- NetBeans.**

NetBeans es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. La plataforma NetBeans es un IDE de código abierto escrito completamente en Java. Tiene soporte para varios lenguajes como PHP, JavaScript, HTML, CSS. Es un producto libre y gratuito sin restricciones de uso. Esta plataforma permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Además, es una base modular y extensible

---

<sup>16</sup> El mayor fabricante de ordenadores del mundo. IBM es el inventor del PC y con sus sistemas medios (AS/400) y sus grandes ordenadores (mainframes) ha revolucionado el mundo de la empresa. Es también uno de los mayores vendedores de software, servicios y equipos de comunicaciones.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

usada como una estructura de integración para crear aplicaciones de escritorio grandes, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación (30).

### **1.4.6-Justificación del IDE escogido.**

Fue escogido el IDE NetBeans debido a que es un producto libre y gratuito sin restricciones de uso, que presenta soporte para varios lenguajes y específicamente en la solución que se desarrolla es importante escribir, compilar, depurar y ejecutar el CMS escogido, el cual integra varios lenguajes tanto del lado del cliente como del lado del servidor, esta integración es realizada a través de módulos preinstalados que hacen mucho más fácil su utilización.

Eclipse no se utiliza porque a pesar que tiene características que pueden ser útiles fue hecho fundamentalmente para Java, la utilización de otros lenguajes es a través de la instalación de plugins lo que puede resultar complicado a la hora de entender el trabajo con este IDE, además presenta un interfaz y entorno gráfico complicado.

Zend Studio se descarta debido a que esta hecho solo para PHP, no integra otros lenguajes que deben utilizarse en el desarrollo de la aplicación.

### **1.5 - Tecnologías de desarrollo.**

#### **1.5.1- Sistemas gestores de contenidos (CMS).**

Los CMS, son sistemas que permiten administrar contenidos en un medio digital. Una herramienta CMS generalmente contiene una interfaz basada en formularios, a los que habitualmente se accede con el navegador, donde se pueden publicar los contenidos fácilmente. Esos contenidos luego aparecen en la página en los lugares donde se ha indicado al publicarlos. Por lo tanto, un CMS está compuesto de dos partes, un back<sup>17</sup> y un front<sup>18</sup>, siendo el back la parte donde los administradores publican las informaciones y el front la parte donde los visitantes visualizan las mismas (31).

Dicho de otra forma, un CMS es una herramienta que permite a un editor crear, clasificar y publicar cualquier tipo de información en una página web. Generalmente los CMS trabajan contra una base de datos, de modo que el editor simplemente actualiza una base de datos, incluyendo nueva información o editando la existente (31).

---

<sup>17</sup> Atrás.

<sup>18</sup> Frente.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

A consideración de las investigadoras los CMS constituyen sistemas encargados de gestionar los distintos contenidos existentes en una página web, permitiendo la manipulación de la información persistente en la base de datos asociada a la misma.

Existen diferentes CMS que han sido desarrollados internacionalmente, de ellos se muestran algunos con sus principales características en los subepígrafos siguientes.

### **1.5.1.1- CMS Drupal.**

Drupal es un sistema de gestión de contenido modular multipropósito y muy configurable que permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. Es un sistema dinámico: en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno web (32).

Es libre, con licencia GNU/GPL, escrito en PHP Hypertext Processor (PHP), desarrollado y mantenido por una activa comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web, y un énfasis especial en la usabilidad y consistencia de todo el sistema (32).

El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitios web (33).

Tiene una capa básica que soporta los módulos que permiten comportamientos adicionales. Los módulos disponibles para Drupal proveen un amplio surtido de características, incluyendo sistemas de e-commerce<sup>19</sup>, galerías de fotos, administración de listas de correo electrónico, etc (33).

---

<sup>19</sup> Comercio electrónico.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### 1.5.1.2- CMS Joomla.

Con Joomla sólo se debe publicar, ya que el sistema gestiona todos los demás detalles técnicos y administrativos, a continuación se muestran sus principales características (34):

- **Organización del sitio web:** está preparado para organizar eficientemente los contenidos de su sitio en secciones y categorías, lo que facilita la navegabilidad para los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores.
- **Publicación de Contenidos:** se puede crear y editar las páginas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados.
- **Escalabilidad e implementación de nuevas funcionalidades:** ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregan servicios de valor a los visitantes de un sitio web, entiéndase: galerías de imágenes, foros, clasificados, etc.
- **Administración de usuarios:** permite almacenar datos de usuarios registrados y también la posibilidad de enviar emails masivos a todos los usuarios. La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de facultades/permisos dentro de la gestión y administración del sitio.
- **Diseño y aspecto estético del sitio:** es posible cambiar todo el aspecto del sitio web tan solo con un par de clicks, gracias al sistema de plantillas que utiliza.
- **Navegación y menú:** totalmente editables desde el panel administrador.

### Desventajas de Joomla (35):

- No brinda flexibilidad al construir el sitio.
- Lleva mucho más tiempo para su configuración y puesta en marcha.
- No garantiza alto rendimiento.

### 1.5.2- Lenguajes de programación a utilizar.

#### 1.5.2.1- Tecnologías del lado del servidor.

##### 1.5.2.1.1- PHP.

PHP es un lenguaje de script interpretado en el lado del servidor, embebido en páginas HyperText Markup Language (HTML) y ejecutado en el servidor, utilizado para la generación de páginas web dinámicas (36).

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas (36).

Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas (36):

- Soporte para una gran cantidad de gestores de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, Sybase mySQL, Informix, entre otras.
- Ofrece una solución simple y universal para las paginaciones dinámicas del servidor web de fácil programación.
- Perceptiblemente más fácil de mantener y poner al día que el código desarrollado en otros lenguajes.
- Soportado por una gran comunidad de desarrolladores, como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.
- PHP permite el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas.

### **1.5.2.2- Tecnologías del lado del cliente.**

#### **1.5.2.2.1- JavaScript.**

JavaScript es un lenguaje de scripts (guiones) desarrollado por Netscape para incrementar las funcionalidades del lenguaje HTML. Sus características más importantes son (37):

- Es un lenguaje interpretado, es decir, no requiere compilación. El navegador del usuario se encarga de interpretar las sentencias JavaScript contenidas en una página HTML y ejecutarlas adecuadamente.
- Constituye un lenguaje orientado a eventos. Cuando un usuario pincha sobre un enlace o mueve el puntero sobre una imagen se produce un evento. Mediante JavaScript se pueden desarrollar scripts que ejecuten acciones en respuesta a estos eventos.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

- Es considerado un lenguaje basado en objetos. El modelo de objetos de JavaScript está reducido y simplificado, pero incluye los elementos necesarios para que los scripts puedan acceder a la información de una página y puedan actuar sobre la interfaz del navegador.

El código JavaScript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido; un script ejecutado en el servidor, sin embargo, sometería a éste a una dura prueba y los servidores de capacidades más limitadas podrían resentir de una continua solicitud por un mayor número de usuarios (38).

### **1.5.2.2.2- CCS (Cascading Style Sheets en sus siglas en inglés, Hojas de Estilos en Cascada en español).**

Las CSS consisten en la creación de estilos propios dentro de las páginas web, o en un archivo separado que es invocado por el sistema al momento de presentar los documentos. El uso de las hojas de estilo data de hace algunos años atrás, aunque sólo han comenzado a popularizarse últimamente. Sin el uso de hojas de estilo, los documentos web son excesivamente largos, ya que cada vez que se necesite cambiar el tipo, el color o el tamaño, se deben redefinir mediante etiquetas en algunas circunstancias, prácticamente párrafo por párrafo y en varios archivos. CSS ofrece más opciones y es más preciso y sofisticado. Además, está soportado en la actualidad por todos los navegadores (39).

### **1.5.2.2.3-XHTML**

XHTML (Extensible Hypertext Markup Language) es un conjunto de documentos basados en XML que nace como extensión del HTML 4.0 y que permite la interoperabilidad entre aplicaciones basadas en XML. Se trata de un paso adelante en el campo del desarrollo web, al aprovecharse las ventajas del XML, manteniendo la compatibilidad de las aplicaciones de usuario realizadas con versiones distintas (40). Su empleo proporciona una serie de ventajas (41):

- Permite un sencillo manejo mediante herramientas XML estándar.
- El código fuente que se obtiene es muy claro y muy fácil de modificar.
- Las páginas hechas con XHTML cargan más rápido y son mejor rastreadas por los buscadores.
- Es muy fácil de aprender.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### 1.5.3- Servidores web.

#### 1.5.3.1- Apache.

El servidor Apache es un servidor web de tecnología Open Source sólido y para uso comercial desarrollado por la Apache Software Foundation. Está estructurado en módulos, los cuales se pueden clasificar en tres categorías (42):

- **Módulos Base:** módulo con las funciones básicas del Apache.
- **Módulos Multiproceso:** son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender dichas peticiones.
- **Módulos Adicionales:** cualquier otro módulo que le añada una funcionalidad al servidor.

En su versión más actual fueron incluidas nuevas características que mejoran su funcionalidad (42):

- Módulos Apache API: donde se utiliza un nuevo conjunto de interfaces de programación de aplicaciones (APIs).
- Filtrado: los módulos pueden actuar como filtros de contenido.
- Directrices simplificadas: se han eliminado una serie de directrices complicadas y otras se han simplificado.

#### 1.5.3.2- Microsoft IIS.

El servidor IIS (Internet Information Server, en español Servidor de Información de Internet) fue desarrollado por Microsoft desde sus inicios se han desarrollado diferentes versiones que constituyen mejoras al mismo y presenta las siguientes características (43):

- Asegura los datos mediante SSL (Secure Sockets Layer, en español Capa de Enchufe Seguro), esto proporciona un método para transferir datos entre el cliente y el servidor de forma segura, permitiendo también que el servidor pueda comprobar al cliente antes de que inicie una sesión de usuario.
- Presenta autenticación implícita que permite a los administradores autenticar a los usuarios de forma segura a través de servidores de seguridad y proxy.
- Es capaz de impedir que aquellos usuarios con direcciones IP desconocidas obtengan acceso no autorizado al servidor, permitiendo especificar la información apropiada en una lista de restricciones.

## CAPÍTULO 1- FUNDAMENTACIÓN TEÓRICA

### **1.5.4- Justificación de las tecnologías escogidas.**

Atendiendo a los CMS analizados anteriormente se puede decir que Joomla no se utiliza para el desarrollo de la aplicación debido a que el mismo a pesar de ser libre requiere de una base de datos MySQL y esta no es libre en toda su totalidad, además con Joomla no se logra flexibilidad, ni rendimiento al diseñar el sitio.

En cambio CMS Drupal posee una elevada flexibilidad, escalabilidad y rendimiento ofreciendo a los que lo utilicen un gestor de contenido altamente configurable con excelentes resultados, además puede tener base de datos PostgreSQL dicho gestor es totalmente libre, característica fundamental debido a la migración de la UCI a herramientas y tecnologías libres que no presenten ningún tipo de restricciones en sus licencias.

Drupal está escrito en el lenguaje PHP utilizando XHTML para el maquetado de las páginas así como CSS para agregar estilos nuevos a las mismas, además del lenguaje JavaScript para la validación de los datos en el lado del cliente garantizando aspectos de seguridad.

Como servidor se utiliza Apache debido a que es libre mientras que Microsoft IIS es totalmente privativo.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

### **CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA:**

#### **2.1- Introducción.**

En el presente capítulo se realiza un estudio profundo del entorno donde se desplegará la aplicación, así como sus procesos y procedimientos de trabajo, para de esta forma establecer mejoras que faciliten la labor de las personas implicadas y que permitan mejorar la eficiencia en la producción, estas mejoras están dadas por los requerimientos funcionales y no funcionales que se identifican, los cuales encaminan todo el proceso de desarrollo del software.

Los conceptos y artefactos que se muestran en los siguientes epígrafes constituyen el resultado de los conocimientos obtenidos producto a la investigación y puesta en práctica de la metodología de desarrollo escogida.

#### **2.2- Modelo del negocio.**

El modelo del negocio captura la lógica del negocio a través de diagramas que evidencian la secuencia de pasos referentes a cómo se realiza el flujo de trabajo en una empresa, dicho modelo brinda las bases para definir las mejoras a realizar para dicho flujo (13).

En los siguientes subepígrafes se muestran los artefactos que surgen a raíz del estudio del negocio en el área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4.

##### **2.2.1- Actores del negocio.**

Un actor del negocio es cualquier persona o entidad con los que el negocio puede interactuar en algún momento, incluso un sistema de información externo puede ser considerado como un actor del negocio (13).

A continuación se muestra el actor involucrado con el negocio que se analiza.

<b>Actor</b>	<b>Descripción</b>
<b>Empleado DPA</b>	Individuo que trabaja con la información recogida en los documentos generados en el área de Calidad.

Tabla 1: Actor del negocio

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

### 2.2.2- Casos de uso del negocio (CUN).

Un proceso de negocio constituye la realización de varias actividades que se relacionan entre sí mediante una secuencia lógica. Un CUN simboliza un proceso de negocio, por lo que se corresponde con una secuencia de acciones que arrojan un resultado significativo para ciertos actores del negocio (13).

Luego de estudiada la lógica del negocio se identifica el CUN:

- Solicitar documento.

### 2.2.3- Trabajadores del negocio.

Representan a personas o sistemas dentro del negocio, son los responsables de realizar las actividades que tributan al cumplimiento de las funcionalidades establecidas en cada flujo de trabajo de una empresa (13).

Seguidamente se evidencia el trabajador implicado en el negocio en cuestión:

Trabajador	Descripción
<b>Empleado de calidad</b>	Persona que trabaja en la gestión de los documentos del área de Calidad.

Tabla 2: Trabajador del negocio

### 2.2.4- Entidades del negocio.

Las entidades del negocio constituyen aquellos objetos que los trabajadores del negocio inspeccionan, manipulan, producen o utilizan durante el cumplimiento de las actividades definidas en los CUN (13).

La entidad involucrada en el negocio que se analiza se define a continuación:

Entidad	Descripción
<b>Documento</b>	Se maneja en el negocio, pueden ser de diferentes tipos: procedimiento, instrucción, mapa de procesos, matriz de procedimientos, registro de calidad, norma,

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

	política de calidad, y evaluación del personal.
--	---

Tabla 3: Entidades del negocio

### 2.2.5- Descripción del CUN.

CUN	Solicitar documento
Actores	Empleado DPA (no pertenece al área de calidad.)
Resumen	Se inicia cuando el actor desea obtener un documento correspondiente a los generados en el área de Calidad para su posterior consulta. Para esto se dirige al Jefe del grupo de calidad y le solicita el mismo. El Jefe del grupo de calidad accede a su máquina de trabajo donde se encuentran almacenados los documentos en formato digital, verifica la existencia del mismo, en caso de existir realiza una copia del mismo y se lo entrega, en caso contrario, informa su inexistencia.
Acción del actor	Respuesta del proceso de negocio
1- Solicitar documento: en esta actividad el actor realiza la solicitud al Jefe del grupo de calidad del documento deseado.  4- Consulta el documento.	2- El Jefe del grupo de calidad verifica la existencia del mismo.  3- Si el documento se encuentra:  Realiza una copia del mismo y lo entrega.  Si no se encuentra:  Informa inexistencia del documento.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

Otras secciones	
Mejoras propuestas	A la vez que se produzca el software se evitará la demora de la búsqueda del documento a través del proceso de solicitud, el sistema permitirá el acceso a los documentos por la red atendiendo al cargo y/o responsabilidades de cada empleado.

Tabla 4: Descripción del CUN Solicitar documento

### 2.2.6- Diagrama de CUN.

Los diagramas de CUN constituyen una representación gráfica de un conjunto de elementos tales como actores y casos de uso, así como las relaciones y dependencias que se establecen entre ellos (13).

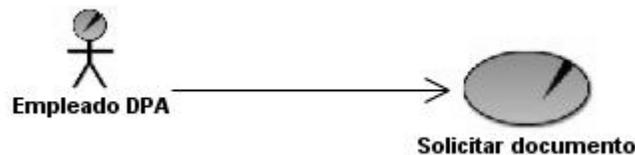


Figura 1: Diagrama de CUN

### 2.2.7- Diagrama de clases del modelo de objetos del negocio.

El diagrama de clases, representa un artefacto que surge producto de la necesidad de describir el modelo de objetos del negocio, evidencia la participación de los trabajadores y entidades del negocio y la relación entre ellos (13).



Figura 2: Modelo de objetos del negocio

### 2.3 - Propuesta de solución.

Teniendo en cuenta las deficiencias encontradas en la gestión documental del área de Calidad de la DPA perteneciente al Centro FORTES de la Facultad 4, y al carecer de un sistema que centralice toda la

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

información generada por el flujo de trabajo de dicha entidad se propone el desarrollo de un sistema que gestione los documentos que generan los procesos de dicha área para de esta forma lograr un control absoluto de toda la documentación.

La solución propuesta constituye una aplicación web administrada por el CMS Drupal al que se le agregaron nuevas funcionalidades y módulos; tiene como nombre Sistema Documental para la Calidad-ZYAN asociado a su principal objetivo y a las iniciales de los nombres de las desarrolladoras.

Los módulos adicionados fueron:

**Avisos:** Muestra avisos a los usuarios atendiendo al rol que desempeñan en el sistema y a las acciones de creación, modificación y eliminación de los documentos. Estos poseen un tiempo de visualización de 48 horas.

**Historial:** Brinda la posibilidad de ver historiales permitiendo visualizar el historial correspondiente a un usuario determinado atendiendo a las acciones de creación, modificación y eliminación a los documentos por parte de este, y “Eliminar historiales por fecha” mediante la cual se pueden eliminar los historiales que se encuentren dentro de un rango seleccionado.

**Buscar:** Posibilita la búsqueda por varios criterios entrados por el usuario.

**Procedimiento, registro, evaluación, mapa, matriz, política, instrucción y norma:** Estos módulos permiten la gestión diferenciada de los distintos tipos de documentos.

**Consultar:** Permite consultar los documentos atendiendo a un tipo seleccionado, solo se podrán visualizar los que se encuentren en la categoría aprobada.

### **2.4- Requerimientos del sistema.**

Un requerimiento se traduce en una condición o capacidad que debe estar garantizada por un sistema o componente de un sistema para satisfacer las necesidades del usuario plasmadas en cualquier tipo de documento impuesto formalmente. Aquellas ideas surgidas por parte de clientes, usuarios y miembros del equipo de proyecto acerca de lo que debe hacer el sistema, deben ser contempladas como candidatas a requisitos (44).

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

En los siguientes subepígrafes se evidencian los requerimientos funcionales y no funcionales con los que cuenta la aplicación.

### **2.4.1- Requerimientos funcionales (RF).**

RF1: Gestionar usuario.

- RF1.1 Crear usuario.
- RF 1.2 Modificar usuario.
- RF1.3 Eliminar usuario.
- RF1.4 Consultar usuario.

RF2: Autenticar usuario.

RF3: Gestionar historial de usuario.

- RF3.1 Ver historiales.
- RF3.2 Eliminar historiales por fecha.

RF 4: Gestionar procedimiento.

- RF4.1 Insertar procedimiento.
- RF4.2 Modificar procedimiento.
- RF4.3 Eliminar procedimiento.

RF5: Gestionar instrucción.

- RF5.1 Insertar instrucción.
- RF5.2 Modificar instrucción.
- RF5.3 Eliminar instrucción.

RF6: Gestionar política de calidad.

- RF6.1 Insertar política de calidad.
- RF6.2 Modificar política de calidad.
- RF6.3 Eliminar política de calidad.

RF7: Gestionar mapa de procesos.

- RF7.1 Insertar mapa de procesos.
- RF7.2 Modificar mapa de procesos.
- RF7.3 Eliminar mapa de procesos.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

RF 8: Gestionar norma.

- RF8.1 Insertar norma.
- RF8.2 Eliminar norma.

RF 9: Gestionar evaluación.

- RF9.1 Insertar evaluación.
- RF9.2 Modificar evaluación.
- RF9.3 Eliminar evaluación.

RF 10: Gestionar matriz de procedimientos.

- RF10.1 Insertar matriz de procedimientos.
- RF10.2 Modificar matriz de procedimientos.
- RF10.3 Eliminar matriz de procedimientos.

RF 11: Gestionar registro de calidad.

- RF11.1 Insertar registro de calidad.
- RF11.2 Modificar registro de calidad.
- RF11.3 Eliminar registro de calidad.

RF12: Consultar documento.

RF13: Crear aviso documento.

RF14: Categorizar documento.

RF15: Buscar documento por atributo.

RF16: Administrar foro.

- RF16.1 Crear foro.
- RF16.2 Eliminar foro.
- RF16.3 Eliminar comentario del foro.

RF17: Participar foro.

- RF17.1 Consultar foro.
- RF17.2 Emitir comentario en foro.

### **2.4.2- Requerimientos no funcionales (RNF).**

RNF 1: Software.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

- Tanto las estaciones de trabajo como el servidor pueden tener instalados los sistemas operativos Windows o Linux, pues el sistema puede ser ejecutado en cualquiera de los dos, se recomienda Windows XP o superior y Ubuntu 10.4 o superior.
- En el servidor se requieren los programas Servidor Web Apache 2.2.6, Drupal 6.9, el SGBD PostgreSQL 8.2 y PHP 5.2.5.
- En las máquinas cliente se accede a la dirección de la aplicación a través de cualquier navegador, se recomienda Mozilla Firefox debido a que Internet Explorer sobre todo en su versión anterior a la 8 no ha asumido totalmente los estándares de codificación HTML, sino que se rige por los propios y por su parte el código HTML que genera Drupal cumple con los estándares de la Consorcio World Wide Web (W3C).

### RNF 2: Hardware.

#### Servidor:

- Para el correcto funcionamiento de la aplicación se debe disponer de un procesador Pentium II o superior y una memoria RAM de 256 MB o superior.
- Capacidad de disco duro en Gigabyte, preferiblemente mayor a los 10 GB.
- Tarjeta de red.

#### Cliente:

- Procesador Pentium II o superior y una memoria RAM de 256 MB o superior.
- Tarjeta de red.

### RNF 3: Requerimientos de apariencia o interfaz externa.

- El sistema debe contar con una interfaz externa amigable que sea fácil de usar y que brinde una correcta navegación para el usuario. Los colores de la interfaz externa deben ser claros procurando de esta forma que no se dificulte la visualización de la misma. Se evita el uso excesivo de imágenes que pueden dificultar el rendimiento de la aplicación.

### RNF 4: Requerimientos de seguridad.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

- Se establece que las funcionalidades que dispone y/o administra el sistema deben ser accedidas por los usuarios autorizados atendiendo al rol asignado siguiendo el principio de mínimo privilegio y haciendo uso de la autenticación.
- Las contraseñas se encriptan mediante la función resumen md5.
- En las consultas se utilizan marcadores de posición y luego las variables a ser incluidas. Esto es un mecanismo de desinfección para evitar ataques masivos de inyección SQL. Se emplea el marcador de posición %d para enteros y %s para las cadenas.
- En caso de usar el sistema operativo Windows, en cada estación de trabajo debe estar instalado el programa Kaspersky Antivirus for Windows Workstations 6.0 o superior, el mismo debe ser actualizado cada dos días.
- Se utiliza un cortafuego en cada estación de trabajo para evitar accesos no autorizados a la información del sistema.

### RNF 6: Requerimientos de usabilidad.

- Los usuarios deben poseer conocimientos básicos correspondientes al manejo de computadoras.
- Se muestran mensajes de error correspondientes al llenado de los datos para garantizar que el usuario se sienta guiado en todo momento en su navegación por la aplicación.

### RNF 7: Requerimientos de soporte.

- Luego de terminado el software se realizan pruebas de rendimiento al mismo para garantizar la eficacia y eficiencia en su funcionamiento, además se brindan servicios de instalación, configuración y mantenimiento del producto a los usuarios finales durante el mes de junio del presente año.
- Se debe adiestrar al personal correspondiente en el trabajo con el mismo.

### RNF 8: Rendimiento.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

- La velocidad de respuesta del servidor ante una solicitud debe ser rápida para evitar retrasos en el flujo de trabajo de la institución.

RNF 9: Portabilidad.

- Multiplataforma. El sistema se podrá instalar en los sistemas operativos Linux y Windows.

A partir de los requisitos definidos anteriormente se procede a modelar el sistema.

### **2.5- Modelo del sistema.**

Para el modelado del sistema se definen actores y casos de uso así como la relación entre ellos, permitiendo una mejor comprensión de las funcionalidades a automatizar en el mismo.

#### **2.5.1- Actores del sistema.**

Se considera actor del sistema a aquellos trabajadores del negocio que realizan actividades que son automatizadas en el sistema, así como a los actores del negocio que deban interactuar con el propio sistema. Los actores del sistema no constituyen parte de él, sin embargo pueden intercambiar información con él, ser un recipiente pasivo de información e incluso representan el rol que desempeña una o varias personas, un equipo o un sistema automatizado (44).

<b>Actor</b>	<b>Descripción</b>
<b>Usuario simple</b>	Actor común entre todos los actores existentes en el sistema constituye una generalización, realiza funciones básicas del sistema: consulta documentos, busca documentos, se autentica y participa en el foro.
<b>Administrador del sistema</b>	Realiza funciones administrativas lleva un control estricto de todos los usuarios y sus permisos así como del foro de la aplicación. Gestiona todo tipo de

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

	documentos, los historiales de usuario así como realiza la consulta y búsqueda de documentos. Especialización de Usuario simple y GestorAdminDoc,
<b>Responsable de procedimientos</b>	Especialización de Gestor PIR y Usuario simple, gestiona documentos de tipo procedimiento o instrucciones.
<b>Jefe del grupo de calidad</b>	Especialización de Gestor admindoc y Usuario simple, gestiona todo tipo de documentos.
<b>Gestor PIR (Procedimientos, instrucciones y algunos tipos de registros.)</b>	Generalización entre Responsable de procedimientos y Gestor admindoc para gestionar procedimientos, instrucciones y algunos tipos de registros.
<b>Asistente de control del director</b>	Gestiona los documentos de tipo evaluación especialización de Gestor evaluaciones.
<b>Gestor evaluaciones</b>	Generalización entre Asistente de control del director y Gestor admindoc para gestionar las evaluaciones.
<b>Gestor admindoc</b>	Generalización entre Administrador del sistema y Jefe del grupo de calidad; especialización de Gestor PIR y Gestor evaluaciones para la gestión de todo tipo de documento.

Tabla 5: Actores del sistema

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

### **2.5.2- Casos de uso del sistema (CUS).**

Un CUS proporciona uno o más escenarios que indican la manera en la que debería interactuar el sistema con el usuario o con otro sistema, para conseguir un objetivo específico y darle cumplimiento a uno o varios requisitos funcionales. Están constituidos por una secuencia de interacciones entre el sistema y sus actores como respuesta a algún evento iniciado por el actor y que se desarrolla sobre el propio sistema (44).

En la presente investigación se identificaron los siguientes CUS:

- Gestionar usuario.
- Autenticar usuario.
- Buscar usuario.
- Gestionar historial de usuario.
- Buscar historial de usuario.
- Llenar historial de usuario
- Gestionar procedimiento.
- Gestionar instrucción.
- Gestionar política de calidad.
- Gestionar mapa de procesos.
- Gestionar norma.
- Gestionar evaluación.
- Gestionar matriz de procedimientos.
- Gestionar registro de calidad.
- Consultar documento.
- Buscar documento.
- Crear aviso documento.

## CAPÍTULO 2- CARACTERÍSTICAS DEL SISTEMA

- Categorizar documento.
- Buscar documento por atributo.
- Administrar foro.
- Participar foro.

Las descripciones correspondientes a estos casos de uso pueden consultarse en el anexo 1.

### **2.5.3- Diagrama de CUS.**

El diagrama de CUS es una representación de los casos de uso que se generan a raíz de los requisitos funcionales y la relación que tienen estos con los distintos actores que interactúan con el sistema. Los casos de uso pueden presentar varios tipos de relaciones: generalización/especialización<sup>20</sup>, include<sup>21</sup> y extend<sup>22</sup>; los actores solo pueden presentar la relación de generalización/especialización; es importante resaltar que un caso de uso puede ser inicializado por un solo actor (44).

---

<sup>20</sup> Se comparten estructuras, propósitos y comportamientos.

<sup>21</sup> El caso de uso de inclusión se inserta explícitamente dentro del comportamiento definido para el caso de uso base.

<sup>22</sup> El caso de uso de extensión tiene un comportamiento opcional u optativo, puede ser usado o no por caso de uso base.



### **CAPÍTULO 3- ANÁLISIS Y DISEÑO:**

#### **3.1- Introducción.**

La realización del análisis y diseño de un software en desarrollo constituye la base para alcanzar un producto final exitoso, durante el proceso de análisis se obtienen artefactos como las clases del análisis y los diagramas de clases correspondientes, los cuales permiten estructurar el sistema y comprender cómo debe ser diseñado el mismo.

A partir de los resultados arrojados por el análisis se procede al diseño, donde se establece con más detalle las características del software final mediante el esbozo de la base de datos, las interfaces de usuario así como la estructura de clases de la aplicación y la relación entre ellas.

En el presente capítulo se muestran los principales elementos pertenecientes al análisis y diseño que arrojan artefactos importantes, los cuales determinan y facilitan en gran medida la implementación y las pruebas de la aplicación que se desea obtener, dichos elementos surgen como resultado de la apropiación del conocimiento por parte de las autoras de la presente investigación.

#### **3.2- Modelo de análisis.**

El modelo de análisis comprende varias características que describen sus principales funciones, las cuales se evidencian a continuación (44):

- Es un modelo conceptual y genérico, constituye una abstracción del sistema.
- Es un esbozo del diseño del sistema.
- Puede sufrir cambios durante el ciclo de vida del software.
- Define una estructura para modelar el sistema.

El siguiente subepígrafe muestra los principales diagramas que se generaron durante el análisis de la aplicación.

##### **3.2.1- Diagramas de clases del análisis.**

Un diagrama de clases del análisis puede representarse por cada CUS o por cada escenario presente en estos casos de uso, este tipo de diagrama representa las clases de tipo interfaz presentadas por cada caso de uso para los actores que interactúan, así como la secuencia de presentación de las mismas.

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

También contiene una clase controladora mediante la cual se accede a la clase entidad para la obtención de los datos de la misma (44):

A continuación se muestra el diagrama de clases del análisis correspondiente al CUS Gestionar procedimiento, los otros diagramas se pueden encontrar en el anexo 2.

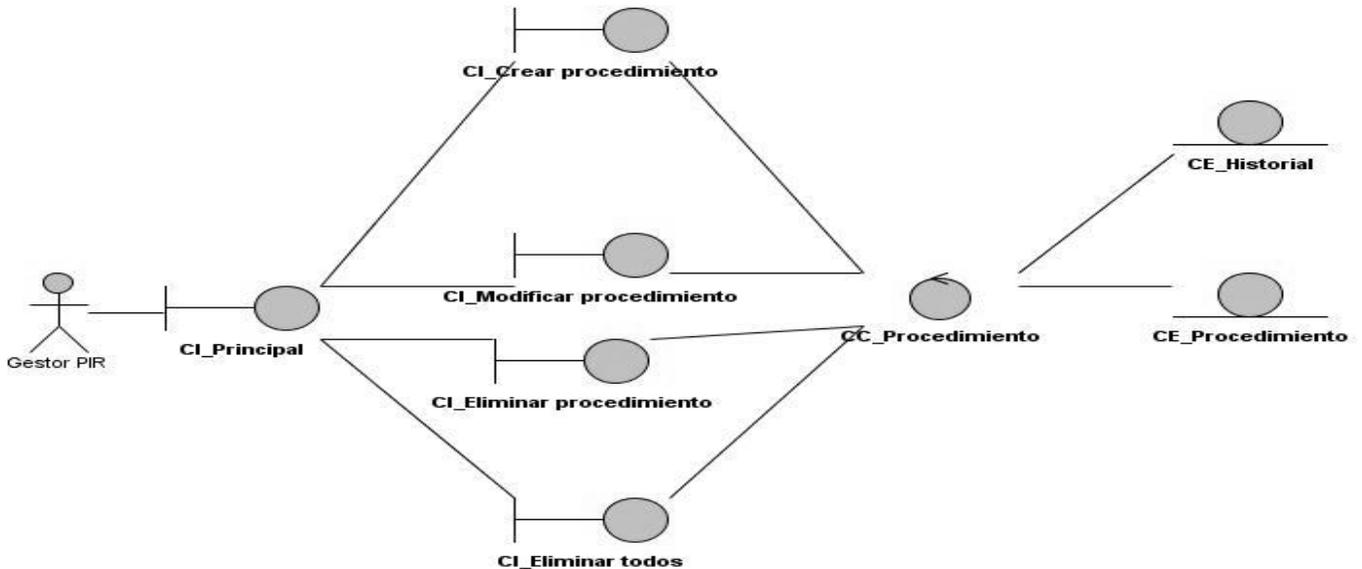


Figura 4: Diagrama de clases del análisis CUS Gestionar procedimiento

### 3.3- Arquitectura del CMS Drupal.

Cuando se habla de la arquitectura de Drupal se puede hablar en términos de "modelo-vista-controlador" ya que Drupal hace una excelente separación entre la persistencia de los datos y los formularios, la lógica, y la vista que emerge como la respuesta HTML al explorador del internet (45).

El secreto de Drupal para conseguir su reconocida flexibilidad y brindar una gran facilidad en la creación de sitios web es la abstracción y organización en capas que utiliza en el tratamiento de los contenidos. A diferencia de otros CMS que consideran a un sitio web como un conjunto de páginas interrelacionadas, Drupal estructura los contenidos en varios elementos básicos: nodos (nodes), módulos (modules), bloques y menús (blocks & menús), permisos de usuario y plantillas (templates) que están representados en diferentes capas; esta separación entre la información que se brinda, las funcionalidades que se ofrecen y el aspecto gráfico posibilita cambiar el diseño y/o la apariencia del sitio web sin necesidad de modificar los

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

contenidos o de cambiar el código de los módulos, incluso permite añadirle funcionalidades sin cambiar su diseño visual lo que resulta muy práctico, pues garantiza la disminución de la complejidad de estas acciones así como el ahorro de tiempo (45).

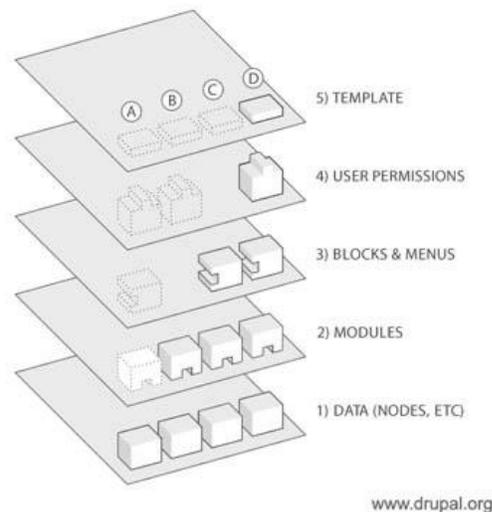


Figura 5: Arquitectura de Drupal

### 3.4- Modelo de diseño.

Por su parte el modelo de diseño presenta varias características que se exponen a continuación (46):

- Es un modelo físico y concreto, muy cercano a la implementación.
- Es una realización del diseño del sistema.
- Debe ser mantenido durante todo el ciclo de vida del software.
- Estructura al sistema.

#### 3.4.1- Patrones de diseño.

Drupal utiliza varios paradigmas orientados a objetos en su diseño poniendo en práctica conceptos tales como objetos, abstracción, encapsulamiento, polimorfismo y herencia. Este CMS recurre al empleo de soluciones simples y elegantes para problemas específicos y comunes del diseño orientado a objetos (47).

**Singleton:** Los módulos y los temas de Drupal son considerados objetos, y siguen el patrón singleton. En general, estos objetos no encapsulan datos, lo que separa a un módulo de otro es el conjunto de funciones que contiene, por lo que debe ser pensado como una clase aislada (47).

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

---

**Decorator:** Drupal hace amplio uso del patrón decorator. El uso de `hook_nodeapi()`, que permite a los módulos extender arbitrariamente el comportamiento de todos los nodos, permite una amplia variedad de comportamientos que se añade a los nodos (47).

**Observer:** El patrón observer es generalizado en Drupal. Cuando una modificación es hecha a un vocabulario en el sistema de taxonomía de Drupal, el `hook_taxonomy`<sup>23</sup> es llamado en todos los módulos que lo implementan. Mediante la aplicación del hook, se han registrado como observadores del objeto vocabulario; cualquier cambio a esto puede entonces actuar como es apropiado (47).

**Bridge:** La capa de abstracción de bases de datos de Drupal se aplica de una forma similar al patrón de diseño bridge. Los módulos necesitan ser escritos en una forma que es independiente del sistema que se está utilizando en la base de datos, y proporciona la capa de abstracción para ello. La nueva capa de base de datos se puede escribir conforme a la API definida por el puente, añadiendo soporte para más sistemas de bases de datos sin la necesidad de modificar el código del módulo (47).

**Chain of responsibility:** El sistema de menús de Drupal sigue el patrón chain of responsibility. En cada solicitud de la página, el menú del sistema determina si hay un módulo para gestionar la solicitud, si el usuario tiene acceso a los recursos solicitados, y que la función se llama para hacer el trabajo. Para ello, el mensaje se pasa a la opción del menú correspondiente a la vía de la solicitud. Si el elemento de menú no puede manejar la petición, se pasa de la cadena. Esto continúa hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario, o la cadena se ha agotado (47).

**Command:** Muchos de los hooks de Drupal utilizan el patrón command para reducir el número de funciones que son necesarias para la aplicación, pasando por la operación como un parámetro, junto con los argumentos. De hecho, el propio sistema de gancho utiliza este modelo, a fin de que los módulos no tengan que definir cada hook, sino sólo los que vayan a aplicar (47).

### 3.4.2- Diagramas de clases del diseño con estereotipos web.

Debido a la utilización de Drupal como CMS para desarrollar el sistema, en el presente subepígrafe se explica brevemente el funcionamiento del mismo y posteriormente se muestra el diagrama de clases.

El contenido en Drupal se almacena en nodos, los cuales constituyen objetos de contenido genérico, que en ocasiones corresponden con una página, un artículo, una entrada de blog, entre otros.

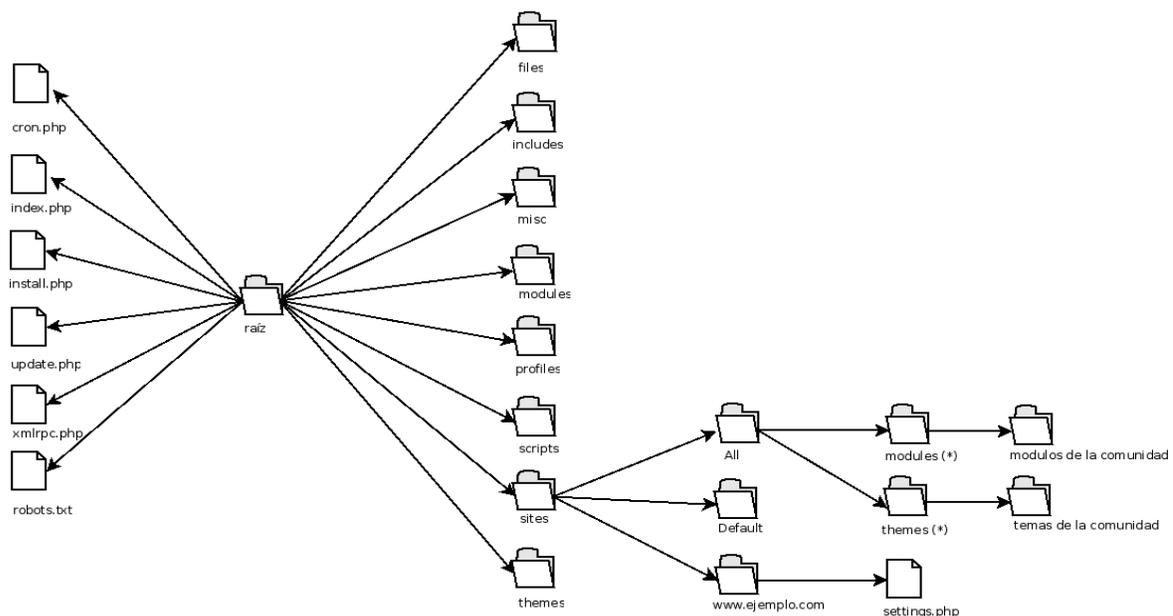
---

<sup>23</sup> Los hooks son funciones que permiten la comunicación con todo el núcleo de Drupal, pueden ser eventos u operaciones con campos, usuarios, etc.

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

Drupal presenta una estructura de carpetas y archivos mediante los cuales se pueden realizar diferentes tareas, a continuación se evidencian cada una de ellas con una breve descripción de sus características (48):

- **Includes:** contiene archivos que poseen código php y que serán llamados por el core de Drupal para la realización de tareas diversas.
- **Misc:** contiene archivos diversos como imágenes, JavaScript, CSS.
- **Modules:** es donde residen todos los módulos del core de Drupal.
- **Profiles:** este directorio contiene los perfiles de instalación.
- **Scripts:** son archivos que contienen instrucciones que pueden ser ejecutados por el Shell<sup>24</sup>.
- **Sites:** en esta carpeta es donde residen todos los archivos de la instalación de Drupal. Es el lugar donde se recomienda colocar todos los temas o módulos adicionales que se instalen. Constituye el único directorio sobre el que se deben realizar cambios. Contiene el directorio files de forma predeterminada.
- **Themes:** en este directorio residen todos los temas que vienen con el core de Drupal.



<sup>24</sup> Dispositivo de software que acepta una entrada que contiene una orden y la ejecuta; la envía a la utilidad que debe ejecutarla, o controla su ejecución.

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

Figura 6: Carpetas y archivos de Drupal

Los diagramas de clases del diseño están compuestos por clases que en ocasiones son representadas con estereotipos web. Estas clases pueden representar diferentes elementos tales como: páginas servidoras, páginas clientes, formularios, clases controladoras, clases de acceso a datos así como las clases de tipo entidad, en este tipo de diagrama también pueden representarse paquetes o carpetas que son controlados por las páginas servidoras.

A continuación se evidencia el diagrama de clases del diseño correspondiente a la presente investigación; el acceso a los datos, el control y los propios datos están representados por los paquetes mostrados, los cuales controlan toda la información de la aplicación:

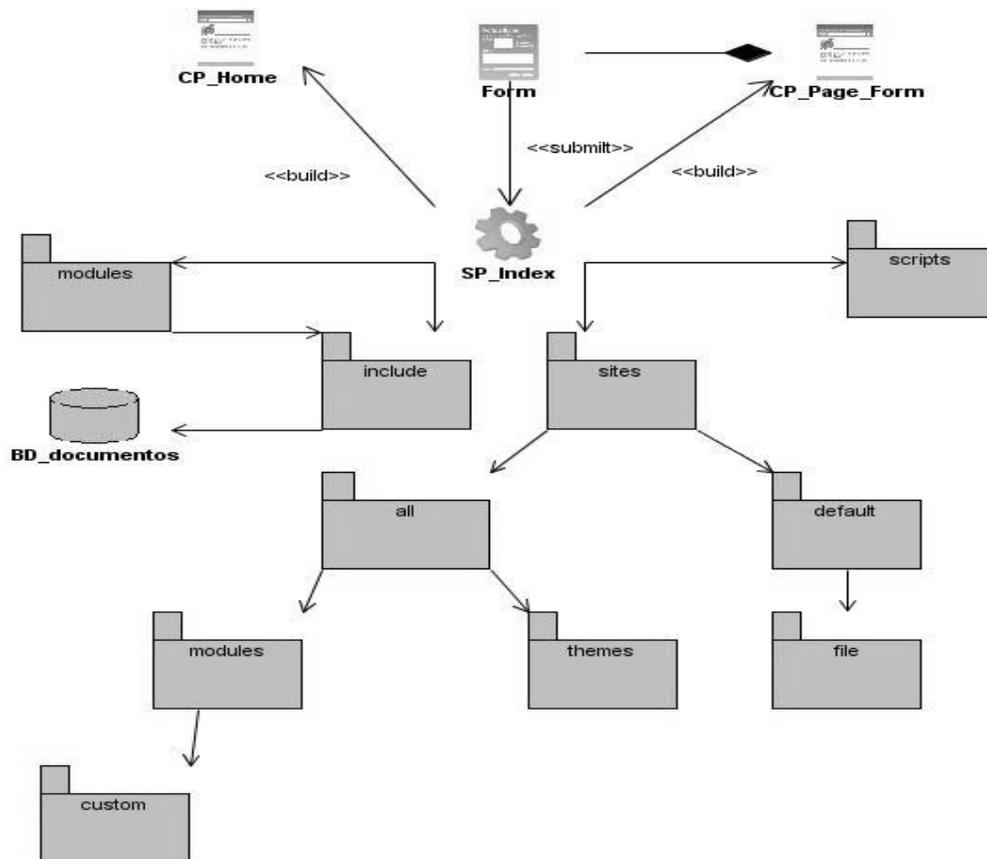


Figura 7: Diagrama de clases del diseño

### **3.4.3- Diseño de la base de datos.**

La base de datos contiene información específica de las entidades que se manejan en el sistema, así como de las relaciones existentes entre ellas, de la calidad de su diseño depende el éxito de la obtención de los datos que residen en la misma.

En los siguientes subepígrafes se evidencian aspectos fundamentales referentes a la base de datos utilizada por el sistema, se muestran las tablas correspondientes a la información persistente en la misma, así como sus atributos, operaciones y la relación existente entre cada tabla.

### **3.4.4- Diagrama de clases persistentes.**

El diagrama de clases persistentes modela las clases persistentes de un sistema, entiéndase por esto aquellas clases que poseen la capacidad de mantener su valor en el espacio y en el tiempo. Es responsabilidad del diseñador definir cuáles clases son las que deben ser persistentes y plasmarlo en el diagrama de clases persistentes junto a las relaciones que se manifiestan entre ellas (44).

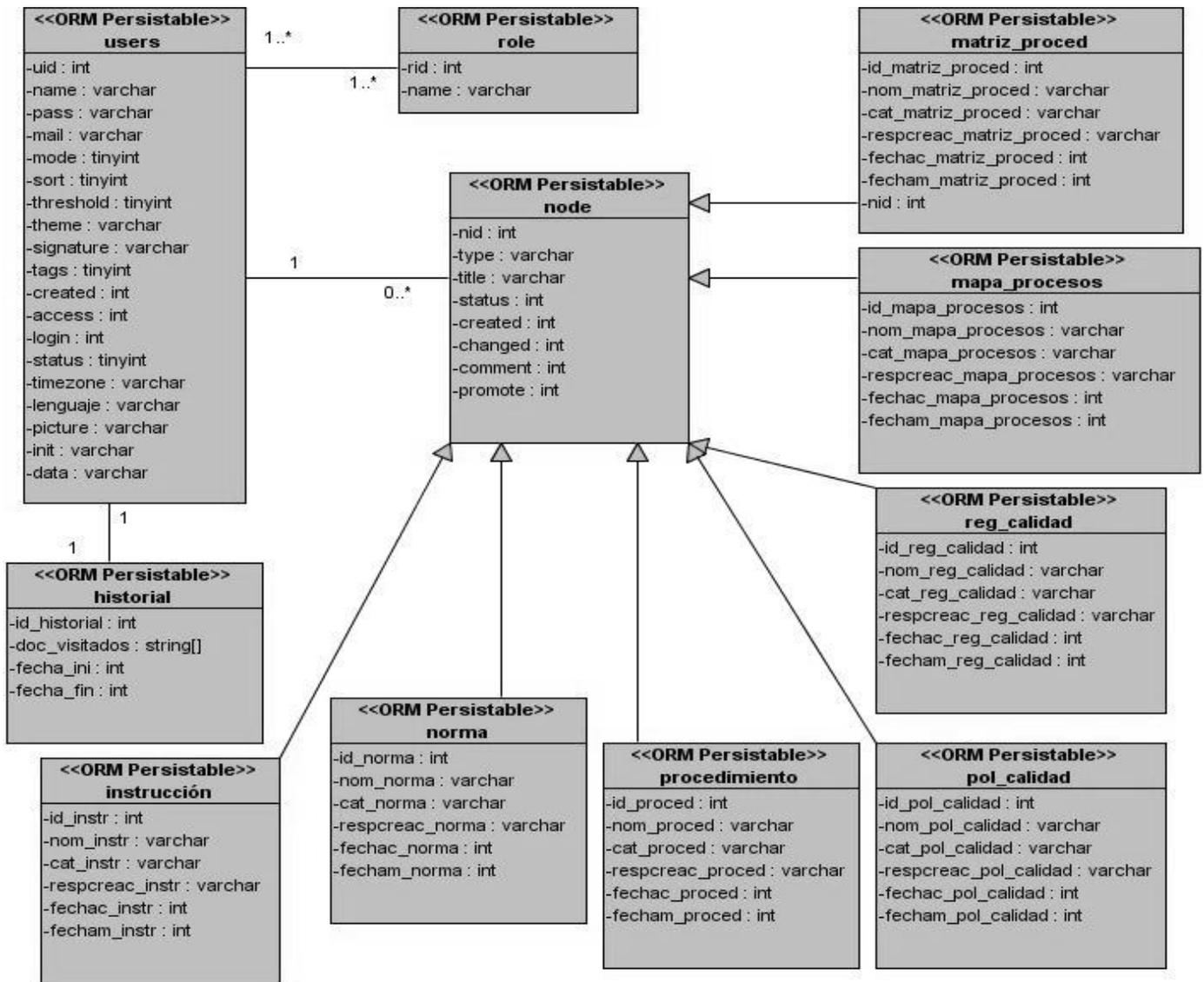


Figura 8: Diagrama de clases persistentes

### 3.4.5- Modelo de datos.

Un modelo de datos constituye un conjunto de reglas o restricciones que permiten describir y manipular los datos de cierta situación de la realidad que se desea modelar y almacenar en la base de datos. Con la realización de un modelo de datos queda descrita la estructura de los datos, las restricciones de integridad

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

que deben cumplir los datos para garantizar que se refleje correctamente la realidad que se está modelando, y por último las operaciones de manipulación de los mismos (44).

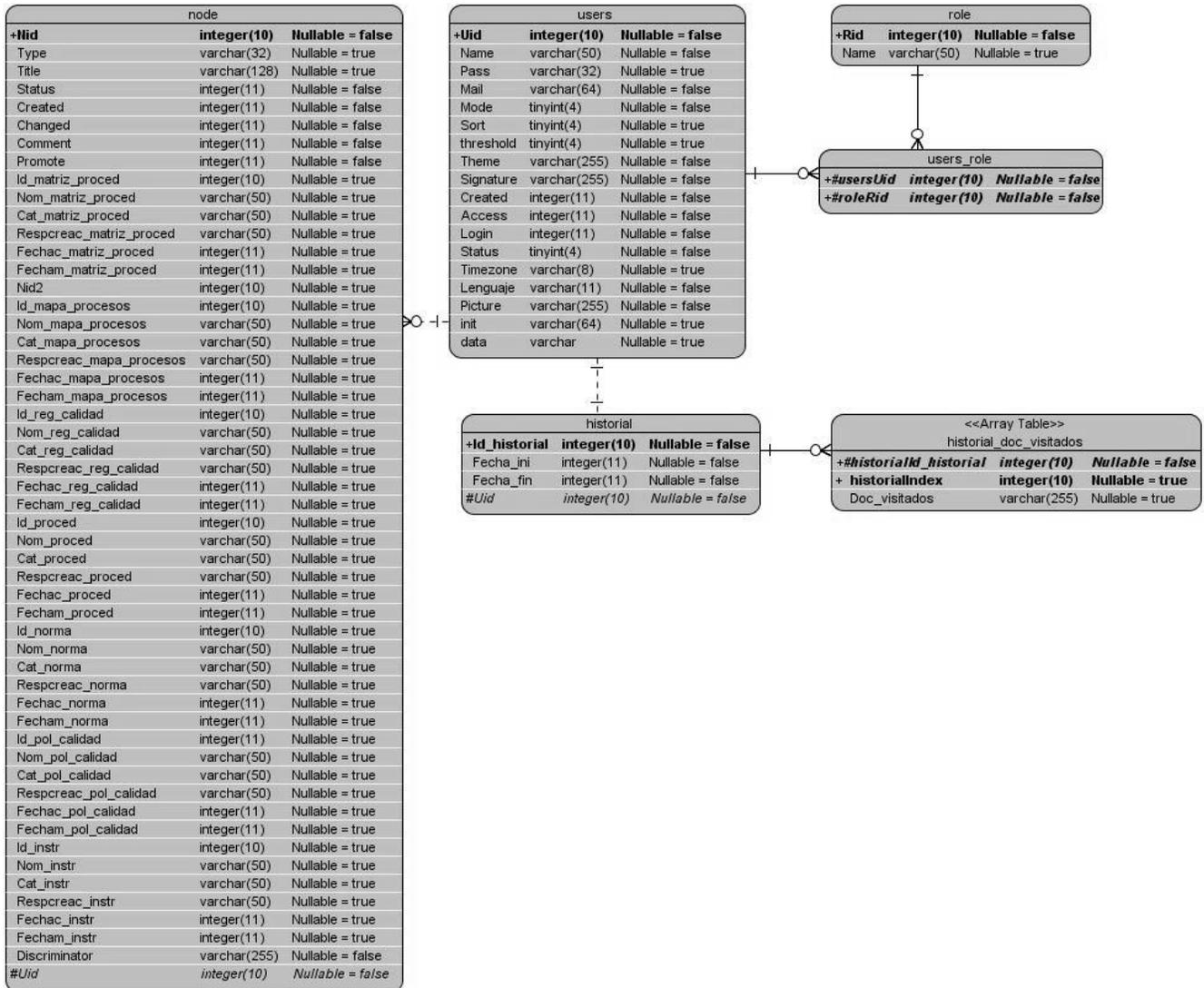


Figura 9: Modelo de datos

### 3.5- Prototipos de interfaz de usuario.

Los prototipos de interfaz de usuario constituyen una representación visual de cómo deben mostrarse las vistas para los usuarios, atendiendo a los permisos definidos para estos, así como a las funcionalidades que les corresponden según las definiciones de los casos de uso.

## CAPÍTULO 3 - ANÁLISIS Y DISEÑO

A continuación se muestra el prototipo no funcional correspondiente a la sesión del rol Gestor admindoc. Los prototipos restantes se encuentran en el anexo 3.



Figura 10: Prototipo de interfaz de usuario sesión para rol Gestor admindoc

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

---

### **CAPÍTULO 4- IMPLEMENTACIÓN Y PRUEBA:**

#### **4.1- Introducción.**

Luego de obtener las bases del software a través del análisis y diseño correspondiente, se realiza el flujo de trabajo de implementación el cual se rige estrictamente por los resultados y artefactos obtenidos en los primeros flujos. A este sistema se le aplican varias pruebas para así ratificar y comprobar la calidad del mismo, en cuanto al código generado y su funcionamiento.

En los siguientes subepígrafes se evidencian los principales resultados arrojados por los flujos de trabajo de Implementación y Prueba; como resultado se obtuvo un sistema que cumple con todos los requisitos tanto funcionales, como no funcionales identificados en los inicios de la presente investigación.

#### **4.2- Modelo de implementación.**

La implementación es la puesta en funcionamiento y/o aplicación de los métodos y medidas necesarios para llegar a la solución final, parte del resultado de las etapas anteriores y comienza a pensar en el sistema en términos de componentes. Persigue desarrollar la arquitectura y el sistema como un todo.

En los siguientes subepígrafes se evidencian los diagramas principales que constituyen la base para una implementación exitosa.

##### **4.2.1- Diagrama de despliegue.**

El siguiente diagrama esboza la arquitectura en tiempo de ejecución de procesadores, dispositivos y los componentes de software que se ejecutan en cada unidad, describe la tipología del sistema, estructura de hardware y los protocolos de comunicación entre ellos.

La aplicación necesita para su ejecución un servidor web Apache, el SGBD PostgreSQL y las estaciones de trabajo, donde cada usuario tendrá acceso a la aplicación.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA



Figura 11: Diagrama de despliegue

### 4.2.2- Diagrama de componentes.

Los siguientes diagramas representan los componentes de software, sus dependencias y la estructura del código. Estos componentes pueden ser de varios tipos tales como fuentes, binarios y ejecutables.

A continuación se presenta el diagrama de componentes correspondiente al CUS Gestionar procedimiento, el resto se encuentra en el anexo 4.

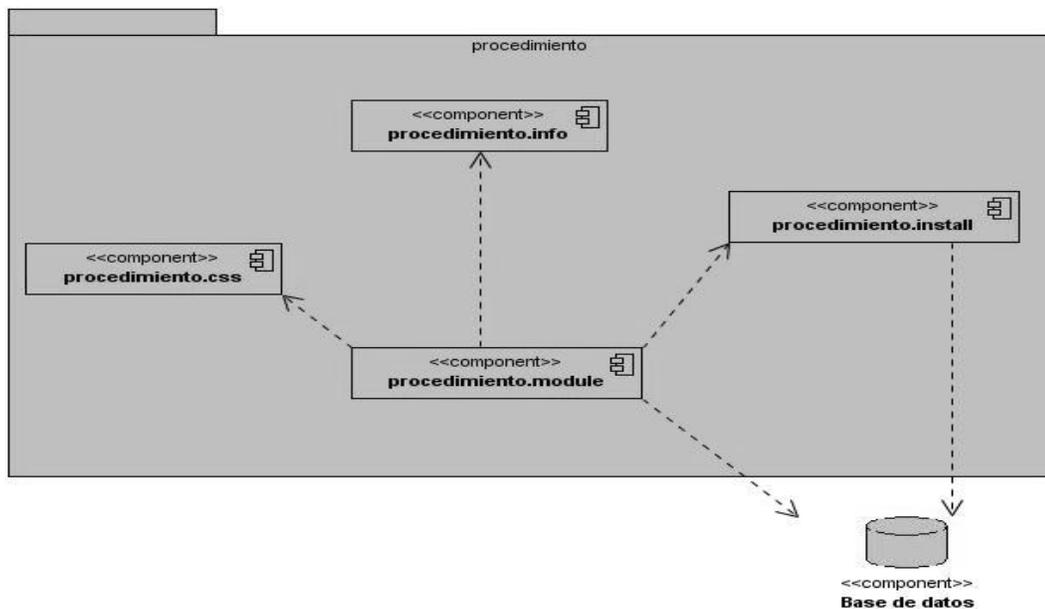


Figura 12: Diagrama de componentes CUS Gestionar procedimiento

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

### 4.3- Estándares de codificación.

Un estándar de codificación constituye un modelo a seguir donde se le da una estructura específica al código, es además una guía probada y efectiva para los programadores. Para que el estándar se considere completo debe comprender todos los aspectos de la generación de código y su puesta en práctica por todos los programadores de un software garantiza una correcta legibilidad y mantenimiento del código (49).

Para la presente investigación son de vital interés los estándares de codificación para base de datos y para el lenguaje de desarrollo PHP.

#### 4.3.1- Estándar de codificación para la base de datos.

- Si el nombre de las tablas se compone de varias palabras deben separarse por un guión bajo y escribirse en minúscula.  
Ejemplo: access\_permission.
- Los atributos se escriben con minúscula, de forma sugerente procurando que transmitan una idea lo más cercana a la descripción de la característica real del objeto.  
Ejemplo: idu, fechac, fecham

#### 4.3.2- Estándar de codificación para PHP.

##### ➤ Comentarios:

El estilo de comentarios que se aplicará debe ser (*/\* \*/* ó *//*). Deben estar elaborados con un lenguaje formal sin faltas de ortografía y con carácter explicativo.

##### ➤ Estructuras de control:

El uso de las llaves es de estricto cumplimiento “{ }”, incluso en situaciones donde son técnicamente opcionales, debido a que con su uso se incrementa la legibilidad del código y reduce la probabilidad de introducir errores lógicos con la agregación de nuevas líneas.

##### ➤ Definiciones de funciones:

En la declaración de una función, los argumentos con valores por defecto van al final de la lista de argumentos. La apertura de la llave debe encontrarse luego del cierre del paréntesis, el cuerpo de la función se encuentra en una línea posterior.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

### ➤ **Llamadas de funciones:**

Las funciones se llaman sin espacios entre el nombre de la función, la apertura del paréntesis y el primer parámetro y sin espacios entre el último parámetro y el cierre del paréntesis, seguido de un punto y coma. Los parámetros se separan por coma y sin espacios entre ellos.

### ➤ **Convención de nombres de variables, funciones y clases:**

Tanto las variables como las funciones deben nombrarse de forma sencilla y descriptiva.

### **4.4-Modelo de prueba.**

El proceso de realización de pruebas al software constituye la etapa donde se evalúa si el producto cumple con las especificidades y requisitos definidos en sus inicios, si es de gran aceptación para el cliente y si puede optimizarse la solución propuesta. Se debe recalcar que esta validación enmarcada en la calidad del software debe realizarse durante todo el ciclo de desarrollo de la aplicación si se desea lograr excelentes resultados (44).

Existen diferentes niveles de pruebas, los cuales enmarcan un objetivo específico a alcanzar con su aplicación, estos niveles están dados por (44):

- Prueba de desarrollador.
- Prueba independiente.
- Prueba de unidad.
- Prueba de integración.
- Prueba de sistema.
- Prueba de aceptación.

En la presente investigación se aplica el nivel correspondiente a las pruebas de sistema con el objetivo de comprobar la disponibilidad y eficiencia del software y que pueda ser usado por los usuarios finales para ejecutar aquellas funciones y tareas para las cuales fue concebido en sus inicios.

Las pruebas realizadas corresponden al tipo funcionalidad, las cuales se ejecutan con el objetivo de validar las funciones, los servicios y los casos de uso presentes en el sistema. Para la puesta en práctica de dichas pruebas se empleó el método de caja negra el cual contempla las pruebas que se aplican a la interfaz del software. Se ejecutaron tres iteraciones de prueba en las cuales se identificaron algunas no conformidades que fueron solucionadas en el transcurso de las mismas.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

En el anexo 5 se evidencian dichas pruebas, representadas por los distintos casos de prueba, los cuales están basados en los CUS identificados en el capítulo anterior. Las no conformidades generadas por estos casos de prueba se muestran en el subepígrafe siguiente.

### 4.4.1- Registro de defectos y dificultades detectadas.

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa	No Sig.	Recomend.	Estado NC	Resp. equipo de desarrollo
Macroproyecto	1	Permitió la modificación dejando el campo macroproyecto vacío, campo que es obligatorio. No mostró el mensaje de error definido en la ECU del Gestionar registro de calidad.	No mostró el mensaje de error correspondiente.	1ra iteración	2: Media	-	-	PD(pendiente) Fecha: 10/05/2011  RA(resuelta y aprobada) Fecha: 12/05/2011	Conforme con la identificación de la NC. Procede.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

Año	2	Permitió la entrada de un año fuera del rango comprendido. No mostró el mensaje de error definido en la ECU del Gestionar registro de calidad.	No mostró el mensaje de error correspondiente.	1ra iteración	2: Media	-	-	PD(pendiente) Fecha: 9/05/2011  RA(resuelta y aprobada) Fecha: 12/05/2011	Conforme con la identificación de la NC. Procede.
Origen de reclamaciones	3	El mensaje mostrado no coincide con la ECU del Gestionar historial de usuario.	El mensaje de error mostrado fue: "Debe introducir correctamente el origen de reclamaciones, existe un exceso de caracteres en el mismo".	1ra iteración	1: Baja	-	-	PD(pendiente) Fecha: 11/05/2011  RA(resuelta y aprobada) Fecha: 13/05/2011	Conforme con la identificación de la NC. Procede.
Nombre	4	El mensaje mostrado no	El mensaje de error mostrado fue:	1ra iteración	1: Baja	-	-	PD(pendiente) Fecha:	Conforme con la identificación de la NC.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

		coincide con el descrito en la ECU Gestionar procedimiento.	“El nombre del procedimiento no debe exceder los 50 caracteres”.					11/05/2011 RA(resuelta y aprobada) Fecha: 12/05/2011	Procede.
El procedimiento será borrado	5	Fue eliminado el procedimiento de la base de datos pero no el archivo de la carpeta de almacenamiento situada en D:\www\ZYAN\sites\default\files\procedimiento.	No eliminó el archivo correspondiente.	1ra iteración	2: Media	-	-	PD(pendiente) Fecha: 11/05/2011 RA(resuelta y aprobada) Fecha: 13/05/2011	Conforme con la identificación de la NC. Procede.
Todos los procedimientos serán borrados	6	Fueron eliminados todos los procedimientos de la base de datos	No eliminó los archivos correspondientes.	1ra iteración	2: Media	-	-	PD(pendiente) Fecha: 11/05/2011 RA(resuelta y	Conforme con la identificación de la NC. Procede.

## CAPÍTULO 4 - IMPLEMENTACIÓN Y PRUEBA

		pero no los archivos de la carpeta de almacenamiento situada en D:\www\ZYAN\sites\default\files\procedimiento.						aprobada) Fecha: 13/05/2011	
Subir archivo	7	No muestra el mensaje correspondiente descrito en la ECU Gestionar mapa de procesos.	El mensaje de error mostrado fue: "Debe adjuntar un archivo".	1ra iteración	1: Baja	-	-	PD(pendiente) Fecha: 11/05/2011 RA(resuelta y aprobada) Fecha: 12/05/2011	Conforme con la identificación de la NC. Procede.

*Tabla 6: No conformidades resultantes al aplicar las pruebas*

## CONCLUSIONES GENERALES

---

### CONCLUSIONES GENERALES:

1. Luego de haber realizado un estudio de diferentes sistemas de gestión documental en los ámbitos internacional, nacional y universitario se concluye que los mismos no cumplen con las demandas y especificidades que requiere el sistema a utilizar en el área de Calidad de la DPA del Centro FORTES de la Facultad 4 ya que presentan limitaciones en sus funcionalidades y en los servicios que brindan, por lo que se decidió desarrollar y aplicar el Sistema Documental para la Calidad-ZYAN para solventar estas deficiencias.
2. Se efectuó un análisis de distintas metodologías con el objetivo de seleccionar la que guiaría el proceso de desarrollo del sistema arribando a la conclusión de que RUP por su alta documentación y grado de especificidad en la descripción de las distintas fases que define, brinda una información detallada de cómo se desarrolló la aplicación a los clientes y futuros desarrolladores interesados en agregarle nuevas funcionalidades.
3. Con respecto a las herramientas de desarrollo se concluye que fueron escogidas atendiendo tanto a la licencia de uso y distribución que poseen como a sus prestaciones, por lo que como herramienta de modelado se escogió Visual Paradigm for UML, como SGBD PostgreSQL, como IDE NetBeans, como CMS Drupal el cual utiliza tecnologías tales como PHP, XHTML, CCS y JavaScript.
4. Atendiendo a la metodología y a la herramienta de modelado escogida se realizó el estudio de las características del sistema lo que permitió comprender la lógica del negocio y definir las deficiencias existentes en la misma para puntualizar las funcionalidades con las que cumple la aplicación.
5. A través de los flujos de trabajo Análisis y Diseño se precisó la estructura que debe estar presente en el sistema y se obtuvo los diagramas de clases del análisis y del diseño los cuales permitieron sentar las bases para una implementación exitosa donde se vincularon los componentes de la aplicación con la arquitectura definida.
6. Con la implementación del sistema ZYAN se obtuvo una aplicación para el área de Calidad de la DPA capaz de gestionar los documentos que se manejan en la misma, de manera controlada y centralizada.

## **CONCLUSIONES GENERALES**

---

7. Las pruebas de funcionalidad realizadas a la aplicación arrojaron algunas no conformidades que fueron solucionadas en sucesivas iteraciones propiciando que el producto final resultara con la calidad y eficacia requerida.

### **RECOMENDACIONES:**

Una vez concluido el desarrollo del presente trabajo de diploma se exhorta a:

- Seguir perfeccionando el sistema desarrollado de acuerdo con las necesidades futuras del cliente.
- La utilización de esta experiencia como material de estudio para el desarrollo de aplicaciones similares por parte de otros desarrolladores.
- Confeccionar el Manual de usuario del sistema.

### REFERENCIAS BIBLIOGRÁFICAS:

1. **Valderrama, Luis David Fernández.** Guell Consulting. [En línea] [Citado el: 10 de diciembre de 2010.] <http://www.sociedadelainformacion.com/12/Gestion%20Documental.pdf>.
2. **Fernández, J. Ramón.** Tecnología aplicada a la administración de fincas. [En línea] [Citado el: 10 de diciembre de 2010.] <http://taaf.blogdiario.com/>.
3. Advice-Business.com. Definición de Gestión Documental. [En línea] [Citado el: 10 de diciembre de 2010.] <http://www.advice-business.com/es/consejos-1538192.htm>.
4. LA GESTIÓN DEL CONOCIMIENTO. GESTIÓN DOCUMENTAL. [En línea] [Citado el: 10 de diciembre de 2010.] <http://www.gestion-conocimiento.com/contenido/gestiondoc.asp>.
5. OrfeoGpl.org. Bienvenido a Nuestra Comunidad OrfeoGpl. [En línea] [Citado el: 13 de diciembre de 2010.] <http://orfeogpl.org/ata/?q=node/1>.
6. Slideshare.Present yourself. Sistema de Gestión documental ORFEO. [En línea] [Citado el: 10 de diciembre de 2010.] <http://www.slideshare.net/radarsas/sistema-de-gestin-documental-orfeo>.
7. Secretaría General de la Administración Pública. Subsecretaría de Informática. [En línea] [Citado el: 15 de diciembre de 2010.] <http://www.informatica.gov.ec/index.php/quipux-desc/quipux-funcionalidad>.
8. **Elda Jeny Báez Álvarez, Jovial Rodríguez Cepero y Boris Luis Orduñez.** Bibliociencias. [En línea] <http://www.bibliociencias.cu/gsd/collect/eventos/index/assoc/HASH25c5.dir/doc.pdf>.
9. Desoft. FichaTecnicaAvilaDoc. [En línea] [Citado el: 13 de diciembre de 2010.] <http://www.desoft.cu/Portals/0/FichaTecnicaAvilaDoc.pdf>.
10. Alfresco. Sobre Alfresco. [En línea] [Citado el: 13 de diciembre de 2010.] <http://www.alfresco.com/es/about/>.
11. Servicios de Internet Cyberpymes. Características de Alfresco. [En línea] [Citado el: 15 de diciembre de 2010.] <http://www.cyberpymes.com/v3/aplicaciones-a-medida/alfresco/alfresco-caracteristicas.html>.
12. **Cuaresma, Sergi Blanco.** Metodologías de desarrollo. [En línea] [Citado el: 10 de noviembre de 2010.] <http://www.marblestation.com/?p=644> .
13. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** El proceso Unificado de Desarrollo de Software. Madrid : Addison Wesley, 2000. 84-7829-036-2.
14. Worldlino. Proceso unificado racional de IBM. [En línea] [Citado el: 15 de diciembre de 2010.] [http://www.worldlino.com/ma/enwiki/es/IBM\\_Rational\\_Unified\\_Process](http://www.worldlino.com/ma/enwiki/es/IBM_Rational_Unified_Process).

## REFERENCIAS BIBLIOGRÁFICAS

---

15. EcuRed. Proceso Unificado de Desarrollo. [En línea] [Citado el: 15 de diciembre de 2010.] [http://www.ecured.cu/index.php/Proceso\\_Unificado\\_de\\_Desarrollo#Fases](http://www.ecured.cu/index.php/Proceso_Unificado_de_Desarrollo#Fases).
16. **Roberth G. Figueroa, Camilo J. Solís, Armando A. Cabrera.** Metodologías tradicionales vs. metodologías ágiles. [En línea] [Citado el: 15 de noviembre de 2010.] <http://adonisnet.files.wordpress.com> .
17. Intro Ingenieria Software . Microsoft Solution Framework 3.0 (MSF). [En línea] [Citado el: 15 de noviembre de 2010.] <http://scruz334.blogspot.es/i2007-11/>.
18. **Pérez, Segundo.** Scribd. Comparación de los modelos MSF Y Cascada. [En línea] [Citado el: 10 de enero de 2011.] <http://www.scribd.com/doc/35014145/Comparacion-Metodo-Msf-y-Metodologia-Cascada>.
19. **Mortier, Gustavo du.** El método Scrum. [En línea] [Citado el: 16 de noviembre de 2010.] <http://www.mastersoft.com.ar/MsWeb/otros> .
20. **Joskowicz, Ing. José.** Reglas y Prácticas en eXtreme Programming. [En línea] [Citado el: 16 de noviembre de 2010.] <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf> .
21. **Luis Giraldo, Yuliana Zapata.** HERRAMIENTAS DE DESARROLLO DE INGENIERIA DE SW PARA LINUX. [En línea] [Citado el: 24 de noviembre de 2010.] [http://hugolopez.phi.com.co/.../file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,\\_id=17](http://hugolopez.phi.com.co/.../file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17).
22. Free Download Manager- Sitio de descargas de software. Paradigma visual para UML (Plataforma Java) (Visual Paradigm for UML [Java Platform]) 6.0. [En línea] [Citado el: 24 de noviembre de 2010.] [http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_%5Bcuenta\\_de\\_Plataforma\\_de\\_Java\\_14715\\_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/).
23. **Medina, Yennifer.** Ingeniería de Software. [En línea] [Citado el: 25 de noviembre de 2010.] <http://yennifermedina.blogspot.es/>.
24. Computer Audio Video System Integrator(CAVSI). ¿Qué es un Sistema Gestor de Bases de Datos o SGBD? . [En línea] [Citado el: 18 de noviembre de 2010.] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
25. **Pecos, Daniel.** PostgreSQL vs. MySQL. [En línea] [Citado el: 18 de diciembre de 2010.] [http://danielpecos.com/docs/mysql\\_postgres/x57.html](http://danielpecos.com/docs/mysql_postgres/x57.html).
26. Descargas Gratuitas. Oracle xe 10g mas conectores para el visual stud. [En línea] [Citado el: 19 de noviembre de 2010.] <http://www.el-hacker.net/oracle-xe-10g-mas-conectores-para-el-visual-stud/>.

## REFERENCIAS BIBLIOGRÁFICAS

---

27. ORACLE. Curso Oracle. [En línea] [Citado el: 20 de noviembre de 2010.] <http://www.uaem.mx/posgrado/mcruz/cursos/miic/oracle3.ppt>.
28. Taringa.Inteligencia Colectiva. PHP 5 todo lo que necesitas (1 solo link) y el Zend Studio v. [En línea] [Citado el: 21 de noviembre de 2010.] [http://www.taringa.net/posts/downloads/894620/PHP-5-todo-lo-que-necesitas-\\_1-solo-link\\_-y-el-Zend-Studio-v.html](http://www.taringa.net/posts/downloads/894620/PHP-5-todo-lo-que-necesitas-_1-solo-link_-y-el-Zend-Studio-v.html).
29. Ubuntu Manitas.Todo lo Relacionado con Software Libre. Java en Ubuntu con Eclipse. [En línea] [Citado el: 22 de noviembre de 2010.] <http://ubuntumanitas.wordpress.com/page/7/>.
30. Taringa. 110 programas gratuitos en un solo link. [En línea] [Citado el: 21 de noviembre de 2010.] <http://www.taringa.net/posts/downloads/5084555/110-programas-gratuitos-en-un-solo-link.html>.
31. Creador Gráfico. Archivo etiqueta CMS. [En línea] [Citado el: 30 de noviembre de 2010.] <http://creadorgrafico.com/?tag=cms>.
32. VersioNet.com. Drupal. [En línea] [Citado el: 1 de diciembre de 2010.] <http://www.versionet.com/drupal>.
33. **Segan**. drupaleros.com.ar. ¿Qué es Drupal? [En línea] [Citado el: 30 de noviembre de 2010.] <http://drupaleros.com.ar/node/28>.
34. **Ingeniería, CeronJC**. Joomla CMS- Sistema de Gestión de Contenidos. [En línea] [Citado el: 2 de diciembre de 2010.] [http://www.ceronjc.com/index.php?option=com\\_content&view=article&id=60&Itemid=69](http://www.ceronjc.com/index.php?option=com_content&view=article&id=60&Itemid=69).
35. **Rincón, Carlos**. Blog. [En línea] [Citado el: 16 de diciembre de 2010.] <http://carlos.rinconsanchez.com/drupal-vs-joomla-una-comparativa-sincera-de-un-consultor-de-ibm>.
36. **Hinostroza, Raul Rodas**. LinuxCentro.net. [En línea] [Citado el: 2 de diciembre de 2010.] <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>.
37. Estréllate y arde. Goodbye Microsoft, Hello Linux. [En línea] [Citado el: 5 de diciembre de 2010.] <http://www.estrelateyarde.org/discover/manual-javascript-lo-mas-basico>.
38. Desarrollo Web. Diseño, desarrollo y ejemplos de páginas web. [En línea] [Citado el: 5 de diciembre de 2010.] <http://www.vadewebs.es/programacion>.
39. Páginas Web sobre HTML y CSS. [En línea] [Citado el: 8 de diciembre de 2010.] <https://belenus.unirioja.es/~guprado/pagweb/caraccss.html>.
40. **Pérez, Javier Eguíluz**. librosweb.es. [En línea] [Citado el: 23 de noviembre de 2010.] <http://www.librosweb.es/xhtml/>.

## REFERENCIAS BIBLIOGRÁFICAS

---

41. **Barea, Julian Vida.** GyC2001 Desarrollo Web. [En línea] [Citado el: 23 de noviembre de 2010.] <http://www.gyc2001.com/ventajas-xhtml.php>.
42. **María José Pérez Fernández, Julio Villena Román.** Trabajo de Diploma. [En línea] septiembre de 2009. [Citado el: 8 de diciembre de 2010.] [http://e-archivo.uc3m.es/bitstream/10016/7562/1/PFC\\_MJose\\_Perez\\_Fernandez.pdf](http://e-archivo.uc3m.es/bitstream/10016/7562/1/PFC_MJose_Perez_Fernandez.pdf).
43. **Ferrer, Fernando.** Al final de la Rambla. [En línea] [Citado el: 16 de diciembre de 2010.] <http://fferrer.dsic.upv.es/cursos/Windows/Avanzado/ch08s02.html>.
44. **S Pressman, Roger.** Ingeniería del Software. Un enfoque Práctico. España : McGraw.Hill/Interamericana, 2002.
45. Tutorial Drupal. Arquitectura. [En línea] [Citado el: 7 de febrero de 2011.] <http://www.cursosdrupal.com/content/arquitectura>.
46. Scribd. Rup etapa de diseño. [En línea] [Citado el: 7 de febrero de 2011.] <http://es.scribd.com/doc/395783/RUP-etapa-diseno>.
47. **Martín, Liber Matos.** Se soluciona programando. [En línea] [Citado el: 24 de enero de 2011.] <http://jitime.blogspot.com/2010/06/poo-y-patrones-de-diseno-en-drupal.html>.
48. Ecosistema Drupal. Estructura de directorios y archivos. [En línea] [Citado el: 25 de enero de 2011.] <http://www.ecosistema-drupal.org/libros/estructura-de-directorios-y-archivos>.
49. Opentia. Estudio sobre Estándares Informáticos tipos y caracterizaciones. [En línea] [Citado el: 20 de febrero de 2011.] <http://people.ffii.org/~abarrio/estandares/OPENTIA-estudio-TiposDeEstandares-20070129.pdf>.