

Universidad de las Ciencias Informáticas

Facultad 1



Título: Núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.

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

Autor(es): Yinet Ríos Martínez

Alejandro Terry Fernández

Tutora: Ing. Elieyis Gómez Padrón

La Habana

2013

Declaración de autoría

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y le otorgamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Yinet Ríos Martínez
Firma del Autor

Alejandro Terry Fernández
Firma del Autor

Elieyis Gómez Padrón
Firma del Tutor



“El Secreto del éxito es la constancia en el propósito”

*Benjamín Disraeli. (Político y escritor
británico)*

Agradecimientos

A mi mami por la confianza que en mí ha depositado, por apoyarme y comprenderme en todo momento, por estar siempre a mi lado y por darme tanto amor. Te quiero del tamaño del infinito y más allá.

A mi papi por apoyarme en estos 5 años, por confiar y sentir tanto orgullo de mí. Te quiero más de lo que imaginas.

A mi hermanita que a pesar de ser tan majadera en ocasiones es lo más hermoso y grande que tengo en mi vida. Pensar en ti me da fuerzas para superar cada uno de los momentos difíciles que he vivido. Gracias por llenarme la vida de felicidad. Te quiero mucho, mucho...

A mi abuela Amparo por siempre darme esos sabios consejos que me ayudan a salir adelante, por estar siempre ahí y por quererme tanto.

A mi abuelo Berto por su amor sin medidas.

A mi tío Wilfre por estar a mi lado cuando más lo he necesitado, por su cariño incondicional y por preocuparse de cada paso que doy en la vida.

A mi tío Berti por todo el amor y la confianza depositada en mí durante toda mi carrera.

A mi tía Yelina por todo su apoyo y comprensión, por todo su cariño y todo lo buena que ha sido conmigo todos estos años.

A mi primo Victor que desde que comencé a dar mis primeros pasos en la universidad estuvo a mi lado apoyándome y ayudándome siempre que pudo.

A Julio que aunque ya no forma parte de mi vida, fue como un padre para mí durante muchos años.

A Samito por toda la ayuda que me ha brindado, por quererme y malcriarme como una hija.

A mi amiga y hermana Maité, por su cariño, por estar en las buenas y en las malas, por soportarme y aconsejarme cuando más lo necesité.

A mis amigos de toda la vida Emilito y Rainer, por aquellos momentos únicos que hemos vivido, por ser tan buenos conmigo y haberme dado fuerzas en los momentos difíciles cuando quizás pensé que no podía seguir adelante.

A mi amiguis Adry por todos aquellos días en los que íbamos a escribir un libro con todas nuestras conferencias sobre la situación que nos rodeaba, por los secretos compartidos, por estar siempre dispuesta a escucharme, aconsejarme y ayudarme en cualquier momento.

A mis amigas Marisol y Yaima por sus buenos consejos, por todos los momentos que compartimos, por ser mis compañeras de cuarto durante 3 años en los cuales pude cogerles mucho cariño.

Agradecimientos

A Javier porque sin su apoyo, amor y comprensión no hubiese terminado los primeros años de la carrera, por darme fuerzas y enseñarme a creer en mí.

A todos mis amigos de los cinco años a Yadira, Anita, Darielys, Elena, Frank, Roberto Carlos, Denis, José Manuel y Alexander, por todos aquellos momentos de mi vida que he compartido con cada uno de ustedes.

A mis amigas de cada etapa de mi vida Maricé, Anilegna y Lili que todavía las llevo conmigo, aunque estemos lejos unas de otras, las quiero mucho.

A mis amigas de la infancia Laurita y Elianys por tantos momentos lindos, por las reuniones de chicas, pero principalmente por ser mis amigas.

A mis compañeros de aula de los grupos 6107, 10207, 1304 y 1511.

A Terry por las largas horas que nos hemos pasado trabajando en la tesis, por tu paciencia y por todos los debates que ayudaron a fortalecer este trabajo.

A mi tutora y a los profesores que ayudaron a crecerme profesionalmente.

A todas las personas que de una forma u otra estuvieron presentes en todos estos largos años.

A todos Gracias.

Yinet Ríos Martínez

A Enrique Machado (Enriquito) por ser un padre para mí y enseñarme que las cosas difíciles de la vida se obtienen trabajando.

A mi madre por darme la vida y estar siempre cuando la necesité.

A mi padre por enseñarme que no importa que la vida nos esté poniendo en aprietos, lo importante es saber levantarse y tener una sonrisa siempre, para alegrar a las personas que se preocupan por nosotros.

A mi esposa que está conmigo todos los días, soportando mis cambios de carácter con paciencia y aconsejándome día a día sobre cómo debo hacer las cosas.

A mis abuelos Horten y Chuchi por acogerme en los inicios de mi vida y educarme para ser el hombre que soy hoy.

A mis abuelos Yolanda y Sergio, por apoyarme siempre y darme su cariño.

A Toti, por apreciarme como si fuera su hijo, educarme y enseñarme siempre a caminar hacia delante.

A Ana, mi suegra que aunque está distante, me ha acogido como si fuera su hijo y se ha preocupado por mi bienestar y el desarrollo de este trabajo de diploma.

Agradecimientos

A Barbarita por quererme como si fuera un hijo más de ella, por su cariño y su preocupación.

A toda mi familia por estar siempre cuando los necesité y por fortalecerme con su unidad.

A Omarito por enseñarme que ser vago no tiene límites y apoyarme cuando estamos hablando de fútbol.

A Miguel por ser mi hermano del Vedado, ayudarme cuando lo necesité y compartir conmigo las derrotas del Barcelona.

A Joaquín, por enseñarme que nadie puede extrañar más la casa que él.

A Alfredo por compartir momentos de su vida conmigo y ser un gran amigo.

A mi compañera de tesis Yinet por poner tanto empeño para que este trabajo obtuviera buenos resultados.

Al grupo 1511 por compartir casi 5 años de mi vida y tratarme tan bien como lo hicieron cuando entré nuevo.

A todas mis amistades por ayudarme cuando los necesité y aportar su granito de arena en la confección de este trabajo.

A mi tutora Elieyis, al profesor Gilberto y demás profesores que me guiaron para la confección de la tesis y me ayudaron para formarme como profesional.

A todas las personas que de una forma u otra me ayudaron para convertirme en un profesional.

A todos, Gracias.

Alejandro Terry Fernández

Dedicatoria

Dedicada a mis padres quienes a lo largo de toda mi vida han sabido formarme con buenos sentimientos, hábitos y valores, los cuales me han ayudado a salir adelante y buscar siempre el mejor camino, por permitirme venir al mundo y estar orgullosos de su niña.

Yinet Ríos Martínez

Dedicada especialmente a Enrique Machado, hombre que por caprichos de la naturaleza no se encuentra con nosotros, pero le debo gran parte de mi vida y de mi formación.

A mis padres por darme la vida y hacerme quien soy hoy.

A mi esposa por estar conmigo todos los días, animándome y guiándome para realizar este trabajo de diploma.

A mi familia por darme ánimo y la oportunidad de compartir mi vida con ellos.

Alejandro Terry Fernández

Resumen

Los medios de comunicación no cuentan con herramientas que permitan gestionar de manera eficiente los recursos fotográficos, lo que trae consigo un uso desorganizado de las imágenes, se hace compleja la integración con otras aplicaciones y existe poca disponibilidad para la publicación de contenidos. Por los motivos antes expuestos se decide crear en el proyecto Gestión Documental para la Prensa del Centro de Ideoinformática de la Universidad de las Ciencias Informáticas, un módulo que permita mejorar el trabajo con las imágenes digitales, así como los elementos que contienen cada una de ellas. El objetivo primordial que se persigue al desarrollar el presente trabajo es detallar todo el proceso para obtener el módulo Ecumene Pyxel v2.0, comenzando por un estudio de un conjunto de herramientas desarrolladas a nivel mundial, posteriormente se identifican los requisitos permitiendo el proceso de análisis e implementación, para finalmente llegar a las pruebas. El módulo desarrollado tiene como base el gestor de contenidos empresariales Alfresco 4.0. Además utiliza como entorno de desarrollo el Eclipse, el Visual Paradigm como herramienta CASE y como metodología de desarrollo OpenUp.

Palabras Claves: gestión documental, imagen digital, Alfresco, módulo, herramienta.

Índice de Contenidos

Agradecimientos.....	4
Dedicatoria	7
Resumen.....	8
Introducción.....	13
Capítulo 1: Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0	17
1.1 Imagen Digital.....	17
1.2 Gestión Documental	18
1.3 Tendencias Tecnológicas	19
1.4 Sistema Gestor de Bases de Datos	22
1.5 Entorno de Desarrollo Integrado.....	23
1.6 Herramienta CASE	24
1.7 Lenguajes seleccionados para el desarrollo.....	24
1.8 Metodología de Desarrollo de Software	26
1.9 Conclusiones del Capítulo	28
Capítulo 2: Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0.....	29
2.1 Características del Sistema.....	29
2.2 Análisis de la Estructura Organizativa	29
2.3 Especificación de los requisitos de Software	36
2.4 Definición de los Casos de Uso del Sistema.....	39
2.5 Arquitectura	45
2.6 Patrones de Diseño	47
2.7 Diagramas de clases de diseño.....	48
2.8 Modelo de Despliegue.....	49
2.9 Conclusiones del Capítulo.....	50
Capítulo 3: Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0.....	51
3.1 Diagrama de Componentes.....	51
3.2 Código Fuente	53
3.2.1 Estándares de Codificación	54
3.3 Pantallas Principales de la Aplicación	55
3.4 Validación del Sistema	56
3.4.1 Pruebas de Funcionalidad	57
3.4.2 Pruebas de Integración	60
3.4.3 Resultados de las pruebas	60

3.5 Conclusiones del Capítulo	61
Conclusiones.....	62
Recomendaciones.....	63
Bibliografías Referenciadas.....	64
Bibliografías Consultada	67
Glosario de Términos	68
Anexos	71

Índice de Figuras

Figura 1 “Diagrama general de la imagen”	32
Figura 2 “Diagrama de Proceso de Identificación de la Imagen”	33
Figura 3 “Diagrama de Proceso de Editorial de la Imagen”	34
Figura 4 “Diagrama de Proceso de Selección de la Imagen”	35
Figura 5 “Diagrama de Proceso de Documental de la Imagen”	36
Figura 6 “Diagrama de Casos de uso”	40
Figura 7 “Arquitectura del ECM Alfresco”	46
Figura 8 “Diagrama de clases”	49
Figura 9 “Modelo de despliegue”	50
Figura 10 “Diagrama de Componentes”	53
Figura 11 “Grupos de usuario del Sistema”	55
Figura 12 “Tipo de Contenido: Fotografía”	56
Figura 13 “Espacios Relacionados con los Procesos”	56

Índice de Tablas

Tabla 1“Actores del Sistema”	39
Tabla 2“Descripción textual CU Gestionar Contenido”	42
Tabla 3“Descripción textual CU Auditar Contenido”	43
Tabla 4“Descripción textual CU Gestionar grupo”	45
Tabla 5“Descripción de Componentes”	52
Tabla 6“Caso de prueba Autenticar usuario”	58
Tabla 7“Caso de prueba Adicionar usuario”	59
Tabla 8“Caso de prueba Editar usuario”	60
Tabla 9“Caso de prueba Integración”	60
Tabla 10“Resultados de las No Conformidades”	61

Introducción

Actualmente las empresas se enfrentan a una fuerte competencia condicionada por la velocidad de los cambios tecnológicos y la innovación, así como la utilización de las nuevas tecnologías de la información y las comunicaciones (TIC).

Las organizaciones necesitan utilizar de forma eficiente todos sus recursos y modificar sus estrategias, con el objetivo de obtener nuevas ventajas competitivas, que le permitan diferenciarse dentro de su entorno. Las estrategias deben estar encaminadas a salvaguardar el patrimonio institucional de dicha organización, por lo que surge la necesidad de almacenar grandes cúmulos de información que deben ser gestionados de manera eficaz. De esta forma nacen soluciones y servicios de gestión documental informatizados, enfocados a solventar la necesidad existente.

La gestión documental, está estrechamente relacionada con el desarrollo de los cambios tecnológicos, por lo que hoy se puede almacenar de forma digital toda aquella información que antes estaba guardada en papel perteneciente a una institución. El avance de los medios de comunicación y la necesidad de mostrarle al mundo, a través de imágenes, dígame vídeos o fotografías, cualquier acontecimiento ocurrido en un lugar determinado ha propiciado tener catalogada o clasificada toda la documentación registrada por una entidad. Esto sugiere a las organizaciones o instituciones implementar métodos organizativos que mantengan controlada la gran cantidad de información disponible.

Estos fundamentos posibilitan soluciones óptimas a los problemas de la gestión de documentos, específicamente en las organizaciones empresariales, que han necesitado buscar métodos para dejar constancia de su actividad (Campillo Torres, y otros, 2009). Varias universidades han realizado estudios para informatizar los aspectos relacionados con la gestión documental.

En la Universidad de las Ciencias Informáticas se desarrolló el sistema de gestión documental de imágenes digitales Ecumene Pyxel v1.0. Esta aplicación cuenta con un núcleo que permite la gestión, identificación y documentación de los recursos fotográficos, integrado al gestor de contenidos empresariales (ECM, por sus siglas en inglés) Alfresco en su versión 3.3g; sin embargo, presenta los siguientes inconvenientes:

- No existe una capa de servicios que facilite la integración con otros sistemas que requieran consultar las fotografías almacenadas en el núcleo de Ecumene Pyxel v1.0.
- Desarrollado en una versión desactualizada del ECM Alfresco.
- Tiempo de búsqueda y recuperación de la información relativamente alto reduciendo la accesibilidad al sistema.

Introducción

- Poca disponibilidad para compartir contenidos, no soporta el acceso a los recursos empleando los protocolos de comunicación WebDAV y SAMBA, lo que limita la publicación de los recursos fotográficos.
- La estructura organizacional concebida no se ajusta a los nuevos esquemas de las instituciones que hacen uso de la gestión documental de imágenes digitales, no se contemplan todos los roles identificados en estas entidades.

Por lo antes planteado se identifica como **problema de la investigación**: ¿Cómo mejorar la gestión documental de los recursos fotográficos en el sistema de gestión documental de imágenes digitales Ecumene Pyxel v1.0?

El **objeto de estudio** se enfoca en los sistemas de gestión documental, incidiendo en el **campo de acción** el núcleo del sistema de gestión documental de imágenes digitales Ecumene Pyxel.

Se propone como **objetivo general** desarrollar el núcleo del sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0 para mejorar la gestión documental de los recursos fotográficos. Para alcanzar el objetivo propuesto se han definido los siguientes **objetivos específicos**:

- Caracterizar las tendencias actuales de los sistemas de gestión documental.
- Diseñar las funcionalidades del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.
- Implementar las funcionalidades del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.
- Validar las funcionalidades del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.

En la realización del presente trabajo se toma como **idea a defender** que con el desarrollo del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0 se mejorará la gestión documental de los recursos fotográficos.

Para dar cumplimiento a los objetivos definidos, se identificaron las siguientes **tareas de investigación**:

- Caracterización de los ECM que permiten la gestión documental de contenidos.
- Selección de las herramientas y tecnologías para el desarrollo del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.
- Identificación de los requerimientos funcionales y no funcionales del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.
- Diseño de las funcionalidades del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.

- Implementación de las funcionalidades del núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.
- Aplicación de las pruebas de funcionalidad, integración y seguridad al núcleo para el sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0.

Para dar cumplimiento al objetivo planteado se emplearon los siguientes métodos científicos:

Métodos teóricos:

- **Analítico-Sintético:** Se utilizó en la revisión de documentos y artículos, donde se extrajeron las ideas centrales relacionadas con el funcionamiento de los sistemas de gestión documental; permitió obtener el conocimiento esencial para generar una propuesta de solución adecuada.
- **Histórico-lógico:** Mediante esta práctica se analizó la versión anterior del núcleo del Sistema de Gestión Documental de imágenes digitales, además se realizó un análisis de un conjunto de sistemas de gestión documental a diferentes escalas permitiendo documentar los elementos necesarios para garantizar la comprensión de estos sistemas.

Métodos empíricos:

- **Entrevista:** Permitted identificar las propuestas a implementar en la nueva versión del Núcleo, así como obtener toda la información referente a la descripción de los flujos de trabajo, los nuevos roles que intervienen en el sistema y sus funciones, para el diseño e implementación de la solución enunciada.

El presente trabajo está estructurado por 3 capítulos, resumen, introducción, conclusiones, recomendaciones, referencias bibliográficas, anexos y glosario de términos.

Capítulo 1. “Fundamentación teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0”: Contiene las principales definiciones acerca de la gestión documental. Se hace un estudio de las diferentes tecnologías y metodologías utilizadas en el desarrollo del presente trabajo de diploma.

Capítulo 2. “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0”: Se describe el problema para su futura implementación. Se realiza un estudio de las posibles modificaciones en el ciclo de vida de la imagen y sus procesos. Contiene además la captura de los requisitos funcionales y no funcionales. Se elabora el modelado del diagrama de clases, diagrama de casos de uso del sistema, modelo de despliegue, especificación de los casos de uso, así como una breve descripción de los actores que intervienen en el sistema.

Capítulo 3: “Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0”: Se presenta una descripción de las principales funcionalidades desarrolladas para el sistema, se muestra el modelo de implementación, diagramas de despliegue y casos de prueba realizados.

Capítulo 1: Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0

En el capítulo que se expone a continuación se tratan una serie de conceptos que ayudan a consolidar las bases de la presente investigación, así como el conjunto de herramientas, lenguajes y metodologías que son empleadas para lograr la personalización del sistema.

1.1 Imagen Digital

La imagen digital es “*la representación bidimensional de una imagen empleando bits¹, que se emplea a instancias de la informática y cualquier dispositivo de tipo digital*” (Gómez, 2012).

La imagen digital se puede obtener a partir de dispositivos de conversión analógica a digital, como es el caso de los escáneres, las cámaras fotográficas digitales o a través de programas informáticos pertinentes; un ejemplo del mismo, son los dibujos que se realizan con el *mouse*.

Los autores del presente trabajo consideran que una imagen digital es una fotografía, un dibujo o una representación gráfica de un determinado objeto, que puede ser utilizado para el análisis posterior de un acontecimiento.

La fotografía tiene un papel importante en los medios actuales de comunicación, gracias a su versatilidad y manipulación, la hace un documento indispensable para la creación de noticias y la representación de un hecho determinado.

Según la Real Academia de la Lengua Española la fotografía es el “*arte de fijar y reproducir por medio de reacciones químicas, en superficies convenientemente preparadas, las imágenes recogidas en el fondo de una cámara oscura*”. Actualmente la fotografía es un recurso informativo que contextualiza, consolida y apoya la información textual. La fotografía ha dejado de ser una simple representación gráfica de un hecho determinado para convertirse en un importante documento que engloba relevantes características que ayudan a su clasificación. Además, encierra en sí misma, cualidades propias que son distintivas al resto de las fuentes de información (Díaz, 2006).

Al igual que el resto de las fuentes documentales, la fotografía está conformada por tres partes esenciales (Díaz, 2006):

- Mensaje: Constituye el núcleo informacional básico, compuesto en este caso por una imagen.
- Formato: Describe cómo esa imagen se encuentra representada, incluyendo las formas extensivas o comprimidas que adquiere con la tecnología digital.

¹ En informática y otras disciplinas es la unidad mínima de información, que puede tener solamente dos valores (cero o uno).

- Soporte: Permite la legibilidad de la imagen fotográfica y su conservación en un fondo documental para formar la memoria.

La fotografía constituye un material permanentemente utilizado por los profesionales del sector periodístico, investigadores e instituciones. Se enmarca en un sistema que encadena los procesos de generación, selección, representación, almacenaje, recuperación, distribución y uso; teniendo en cuenta las particularidades que presenta el documento (Díaz, 2006).

Por lo antes expuesto y dada la proliferación de este recurso en los medios informativos, se hace necesario encontrar soluciones que permitan el manejo eficiente de las imágenes.

1.2 Gestión Documental

La gestión documental es el almacenamiento, recuperación y captura de documentos para facilitar la búsqueda y manejo de información. Mediante la gestión documental se pueden archivar los documentos en carpetas físicas o digitales, ofreciendo a cada persona que opera con ellos una manipulación rápida y eficiente (Prieto, 2009).

Para evitar la pérdida de la información lo más idóneo es emplear sistemas de gestión documental porque en un gran por ciento permiten asegurar, recuperar y personalizar de manera eficaz los documentos de entrada-salida de la empresa (Telecon, 2007).

En general la importancia del empleo de sistemas de gestión documental para el desarrollo del presente trabajo está basada en (Notte, 2009):

- Evitar la pérdida de documentos, ya que están todos seguros en un servidor central.
- Evitar la pérdida de tiempo debido a que varias personas tengan que recuperar los documentos desde diferentes lugares y departamentos. Los documentos son inmediatamente recuperados por la persona que está buscando la información.
- El acceso a la información se realiza de forma segura y estructurada al poder definir diferentes permisos de acceso a los datos y documentos dentro del sistema de gestión documental.
- Captura y extracción de la información de los documentos en papel y la integración en el sistema de gestión documental como metadatos.
- Indexación de los metadatos para su fácil recuperación mediante búsquedas personalizadas.
- Recuperación de los documentos en formato electrónico y archivado de las copias físicas.

Una eficaz gestión documental es fundamental para conservar la historia de cualquier empresa que se rige por la actividad archivística del documento hasta su destino final, transformándose además en una herramienta óptima para conservar y aprovechar información. Gracias al desarrollo tecnológico de la información existen herramientas, capaces de facilitar la gestión documental en una entidad o centro

donde sea necesario utilizarla dado el volumen de información manejados, estas herramientas son llamadas Gestores de Contenidos Empresariales.

1.3 Tendencias Tecnológicas

A continuación se caracterizan un conjunto de herramientas para la gestión documental, con el fin de seleccionar la tecnología más apropiada basada en la filosofía de software libre y su capacidad para ser personalizadas con la mayor cantidad de funcionalidades para la gestión de estos recursos. Los sistemas que serán estudiados son el gestor de contenidos empresariales Alfresco y el Knowledge Tree a nivel internacional, el ÁvilaDoc como sistema a nivel nacional y a nivel institucional el Exscriba.

ECM Alfresco

Los ECM son herramientas usadas para capturar, manejar, salvaguardar, preservar y entregar contenido y documentos relacionados con procesos organizacionales. Es decir, un ECM tiene como finalidad el manejo eficiente de la información (Borrego, 2009).

Esta tecnología permite reducir los tiempos para la creación de políticas de venta, compra, manuales de políticas y procedimientos. Además, para centralizar la información por parte de las empresas en un solo lugar, y sobre todo mejorar la toma de decisiones (Borrego, 2009).

Alfresco es una alternativa de código abierto que permite desarrollar proyectos de contenido empresarial. Cumple con la filosofía de software libre y está desarrollado en Java. Ofrece una arquitectura flexible para proveer la gestión de documentos (Leyva, 2011).

Las aplicaciones construidas usando Alfresco pueden ser desplegadas en cualquier ambiente. Permite usar los siguientes gestores de bases de datos, PostgreSQL, MySQL, Oracle, entre otros; e incluso puede ejecutarse sobre cualquier servidor de aplicaciones que soporte Java y trabajar con el navegador web que desee.

A continuación se exponen otras características que presenta el ECM Alfresco (Pyme, 2012):

- Acceso al repositorio vía CIFS/SMB, FTP y WebDAV.
- Gestión de documentos e imágenes.
- Aplicación de permisos en un espacio (carpeta) o se puede configurar para cada elemento de contenido.
- Soporte multiplataforma (oficialmente Windows, GNU/Linux y Solaris).
- Interfaz gráfica basada en navegadores de Internet (oficialmente Internet Explorer y Mozilla Firefox).
- Soporte de varios idiomas.

Capítulo 1: “Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0”

- Facilita la configuración de flujos de trabajo.

A pesar de ser un sistema completo y robusto acorde a sus características y funcionalidades posee como uno de sus inconvenientes que su interfaz de usuario es de difícil manejo y requiere de un conocimiento básico o avanzado del usuario final.

Knowledge Tree

Knowledge Tree es una herramienta basada en software libre, destinada a la gestión documental y tiene como objetivo el control de todo el ciclo de vida de los documentos de la empresa. Es un completo gestor documental que permite guardar, ordenar y recuperar documentos de forma ágil y segura, controlando accesos, versiones y flujos de trabajo (Selection, 2009).

Las principales funcionalidades que presenta el software Knowledge Tree son (Maggi, 2011):

- Definir una estructura de carpetas y controlar el acceso de los usuarios a cada una de ellas según un perfil de autorización que distingue entre las diferentes acciones ejecutables para cada documento.
- Crear grupos de usuarios para facilitar la asignación de permisos. Mantener el historial de acciones ejecutadas sobre un documento, permitiendo la trazabilidad de los cambios mediante un sistema de control de versiones.
- Definir flujos de trabajo controlando, mediante roles, las transiciones de un documento durante su ciclo de vida.
- Generar avisos a los usuarios interesados, sobre el cambio de estado de los documentos o ejecutando acciones automáticas sobre ellos.
- Definir tipos de documentos, asignando a cada uno de ellos, campos variables que facilitarán su búsqueda.
- Realizar búsquedas multi-criterio y almacenarlas para futuras ejecuciones.
- Tener accesible un tablón de anuncios con el resumen de sus actividades.

KnowledgeTree provee todas las funcionalidades de revisión, edición y colaboración que cualquier empresa pequeña y mediana puede necesitar (Maggi, 2011).

Su principal desventaja se centra en que presenta inconvenientes con la escalabilidad y las traducciones adicionales proporcionadas por la comunidad.

Capítulo 1: “Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0”

ÁvilaDoc

ÁvilaDOC es una aplicación web, desarrollada sobre la plataforma de software libre con una base de datos centralizada, destinada a la gestión, tramitación y resguardo de archivos electrónicos y/o digitales; facilitando la búsqueda o recuperación de información de forma rápida y sencilla (Badillo Goy, y otros, 2007).

La aplicación permite la organización y distribución de la información mediante el uso de carpetas con estructura de árbol, facilitando así el acceso personalizado a parte o a todo el árbol de información.

Permite acceder directamente al escáner y el tramitado interno de los documentos se realiza sin necesidad de reproducirlos o distribuirlos físicamente a las oficinas involucradas. Esto se hace a través del sistema en la red y los usuarios no necesitan moverse de sus puestos de trabajo (Badillo Goy, y otros, 2007).

ÁvilaDOC también cuenta con otras funcionalidades entre las que se pueden mencionar: cierre de periodo, recuperación de información y traslado de información al archivo pasivo (Badillo Goy, y otros, 2007).

Objetivos del sistema.

- Lograr un archivo digital centralizado.
- Organizar el flujo documental.
- Definir estructura archivística.
- Implementar archivos portables.

Entre sus ventajas más notables se encuentran (Badillo Goy, y otros, 2007):

- El sistema constituye un archivo digitalizado, que agiliza las tareas de registro, búsqueda, reproducción y distribución de los documentos.
- 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.
- Facilita el conocimiento del nivel de percepción de los servicios que brindan, Aunque proporciona una visión relativa, ya que sólo reflejan la opinión de las personas que han optado por expresar su incidencia.

El sistema posee como inconveniente la no existencia de una capa de servicios web para lograr una interoperabilidad con sistemas externos.

eXcriba

eXcriba es un sistema creado para la gestión de grandes cantidades de documentos. Es desarrollado por

ingenieros del Centro de Informatización Universitaria de la Facultad 1 de la Universidad de las Ciencias Informáticas. Es un sistema amigable, de fácil uso, orientado a usuarios de poco conocimiento de la informática. El sistema gestor de documentos administrativos está compuesto por un núcleo o motor que es el ECM Alfresco y una aplicación Web la cual se integra mediante los *Webservices API* y los *External WebScripts* que brinda el ECM Alfresco.

eXcriba, permite identificar un mercado creciente de necesidades y adaptar el software a las características propias y específicas de los interesados. Su principal inconveniente es que no permite el tratamiento documental de la fotografía en todo su ciclo de vida (Leyva, 2011).

Gestor de Contenidos Empresariales seleccionado

Las tecnologías de gestión documental estudiadas permiten eficiencia y seguridad en la gestión de contenidos, cada una de ellas presenta ventajas y limitaciones. Debido a los requisitos y características del proyecto Gestión Documental para la Prensa, las necesidades, restricciones y conocimiento del cliente, se decide seleccionar como tecnología para dar solución a la problemática planteada, el ECM Alfresco. Este sistema cumple con las políticas de software establecidas por el país y es una solución de software libre que extiende la Gestión Documental para dar respuesta al ciclo de vida de forma íntegra, no solo de los documentos, sino de todos los contenidos de una organización. Además la versión anterior del núcleo Ecumene Pyxel fue desarrollada en el ECM Alfresco.

1.4 Sistema Gestor de Bases de Datos

Un Sistema Gestor de Bases de Datos (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 consulta².

PostgreSQL

Es un potente Sistema Gestor de Bases de Datos relacional libre. Funciona en todos los sistemas operativos importantes, Linux, Unix, BSDs, Mac OS, Beos, Windows, entre otros. Es considerado como el SGBD de código abierto más avanzado del mundo. Permite incrementar las prestaciones y el rendimiento de las aplicaciones (Martínez, 2009).

Entre las ventajas del potente gestor de bases de datos son destacadas las siguientes (Martínez, 2011):

- Instalación ilimitada, multiplataforma.
- Posee una gran escalabilidad. Es capaz de ajustarse al número de Unidades Centrales de

² http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos

Procesamiento (CPUs por sus siglas en inglés) y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta.

- Ahorros considerables en costos de operación.
- Implementa el uso de subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz y ofreciendo soluciones en campos en los que MySQL no podría.
- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos.
- Permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.
- Tiene una documentación muy bien organizada, pública, libre, con comentarios de los propios usuarios y comunidades muy activas, varias en castellano.
- Herramientas gráficas de diseño y administración de bases de datos.

De acuerdo con las características del sistema y el volumen de información existente en los medios de comunicación o entidades que realizan la gestión de imágenes; PostgreSQL es el Sistema Gestor de Bases de Datos más indicado para cumplir con el objetivo propuesto; permite la instalación ilimitada y es un sistema multiplataforma. Asegura la integridad y confiabilidad de los datos. Es uno de los sistemas más importantes a nivel mundial, ya que permite la manipulación y almacenamiento de grandes volúmenes de información, manteniendo su estabilidad.

1.5 Entorno de Desarrollo Integrado

Un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) es un programa informático compuesto por un conjunto de herramientas de programación y consiste en un compilador o depurador de interfaz gráfica. Algunos entornos son compatibles con múltiples lenguajes de programación como el Eclipse el cual está basado en Java³.

Eclipse

Eclipse es un IDE que facilita las tareas de edición, compilación y ejecución de programas durante su fase de desarrollo (Martínez, 2007).

Eclipse es una aplicación de código abierto, que proporciona el entorno de desarrollo solamente, siendo necesario disponer del Paquete de Desarrollo de JAVA (JDK, por sus siglas en inglés) para poder compilar y ejecutar las aplicaciones desarrolladas en el lenguaje de programación Java (Martínez, 2007).

³ <http://programaciondesarrollo.es/que-es-un-entorno-de-desarrollo-integrado-ide/>

1.6 Herramienta CASE

Una herramienta CASE más conocida como Ingeniería Asistida por Ordenador es una aplicación informática dedicada a aumentar la productividad en el desarrollo de software ayudando a que se reduzca el coste en cuanto al tiempo y dinero (Gómez, 2011).

Esta herramienta es un apoyo para realizar el diseño en todo el ciclo de vida de desarrollo del software, además permite la compilación automática y la detección de errores así como la documentación de información (Gómez, 2011).

Visual Paradigm

Es una herramienta CASE que propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Ha sido concebido para soportar el ciclo de vida completo del software mediante la representación de todo tipo de diagramas. Es una herramienta privada, disponible en varias ediciones, fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque orientado a objetos; se caracteriza particularmente por (Pressman, 2002):

- Multiplataforma.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Licencia gratuita y comercial.
- Fácil de instalar y actualizar.

1.7 Lenguajes seleccionados para el desarrollo

Un lenguaje, en términos informáticos, es una herramienta diseñada para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo.

En la actualidad existen varios lenguajes, tanto de programación, como de marcado los que han ido evolucionando según las características y necesidades de las plataformas para las cuales han sido desarrollados.

Capítulo 1: “Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales *fcumene Pyxel v2.0*”

XML

XML es la abreviatura de *Extensible Markup Language*, lo que se traduce como lenguaje de marcas generalizado. Se utiliza para definir la estructura de los documentos, es un estándar para el intercambio de información entre diferentes instancias, logrando ser empleado para estructurar bases de datos, editores de textos u hojas de cálculo (Lapuente, 2011).

XML tiene punteros a la estructura de los datos, lo que ahorra tiempo y simplifica el software de aplicación. Posee independencia de los navegadores y del sistema de objetos. Es un lenguaje que permite jerarquizar y estructurar la información así como describir los contenidos dentro del propio documento (Lapuente, 2011).

Se selecciona el lenguaje, porque facilita la creación de los tipos de contenido, es utilizado por el ECM Alfresco para la configuración de la mayoría de sus ficheros. Tiene formatos abiertos y flexibles que permiten ser más ágiles en el acceso a las bases de datos y en la entrega de estos datos a los clientes de la aplicación web.

JavaScript

JavaScript es un lenguaje que no requiere compilación, es utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página Web. Se trata de un lenguaje de programación del lado del cliente compatible con la mayoría de los navegadores modernos, se interpreta en el cliente y en su código usa objetos incorporados del sistema pero no provee creación de clases o herencia, puede ser embebido en un documento HTML y por último no se hace necesaria la declaración de variables y tipos de datos (Navarrete, 2007).

JavaScript nació con la necesidad de permitir a los autores de sitios web crear páginas que permitan intercambiar con los usuarios, se necesitaba crear webs de mayor complejidad y el HTML solo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios.

Es un lenguaje basado en acciones que poseen menos restricciones, gran parte de la programación está centrada en describir objetos, escribir funciones que respondan a movimientos del *mouse*, aperturas, utilización de teclas así como cargas de páginas. Es independiente de la plataforma, además se puede ejecutar en cualquier ordenador del mercado: Linux, Windows, Apple entre otros y debido a sus características es muy empleado para internet (Valdés, 2007).

Ventajas: Es un lenguaje seguro y fiable, los *scripts* tienen capacidad limitada por razones de seguridad y el código se ejecuta en el cliente.

Capítulo 1: “Fundamentación Teórica del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0”

Desventajas: Presenta un código visible para cualquier usuario y el código debe descargarse completamente.

Su selección para el desarrollo del sistema está dada por su compatibilidad con los navegadores modernos y es factible para implementar los servicios que serán utilizados para establecer una conexión con sistemas externos.

UML

UML es el Lenguaje de Modelado Unificado seguido por una serie de métodos de análisis y diseño orientados a objetos y una de sus metas principales es avanzar en el estado de la integración institucional proporcionando herramientas de interoperabilidad para el modelado visual de los objetos sentando las bases para que surjan una serie de beneficios que aparecen a continuación: (Weitzenfeld, 2005)

- Mejoras en los tiempos totales de desarrollo.
- Permite modelar sistemas utilizando conceptos orientados a objetos.
- Establecer conceptos y artefactos ejecutables.
- Encaminar el desarrollo del escalamiento en sistemas complejos de misión crítica.
- Crear un lenguaje de modelado utilizado tanto por humanos como por máquinas.
- Mejor soporte para la planeación y el control de los proyectos.
- Alta reutilización y minimización de costos.

El lenguaje de modelado UML se selecciona para definir los procesos que intervienen en el sistema, por su representación gráfica.

1.8 Metodología de Desarrollo de Software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas, herramientas así como un soporte documental que ayuda a los desarrolladores a realizar nuevo software, se centran en la definición detallada de los procesos (Commons, 2006).

RUP

Es una metodología de desarrollo de software que está basada en componentes e interfaces bien definidas, y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP (Proceso Unificado de Desarrollo) es un proceso que puede especializarse para una gran variedad de sistemas de software, en diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto (Peña, 2011).

Tiene como inconveniente que genera complejidad en los controles de administración, debido al volumen de documentación que presenta.

OpenUP

Está definida como un proceso iterativo para el desarrollo de software que permite la construcción de un sistema extensible que puede ser empleada para agregar o adaptar procesos. Es una metodología centrada en la arquitectura que permite minimizar riesgos así como organizar el desarrollo evolutivo de forma que se pueda obtener retroalimentación y mejoramiento continuo. Disminuye las probabilidades de fracaso e incrementa las de éxito. Evita la elaboración de documentos, diagramas e iteraciones innecesarios que son imprescindibles en RUP⁴ y por ser una metodología ágil tiene un enfoque centrado al cliente y con iteraciones cortas (Karen, 2008).

Entre los aspectos más importantes de OpenUP se destacan los siguientes (Hernández, 2012):

- Es una metodología de desarrollo de software de código abierto diseñado para pequeños equipos organizados, quienes quieren tomar una aproximación ágil del desarrollo.
- Proceso iterativo e incremental que es Mínimo, Completo y Extensible.
- Se valora la colaboración y el aporte de los *stakeholders*⁵ sobre los entregables y las formalidades innecesarias.
- Practicantes de desarrollo de software (desarrolladores, administradores de proyectos, analistas y probadores) trabajan juntos como un equipo de proyecto. No define un modelo de negocio ni de dominio necesario.
- Permite detectar errores tempranos a través de un ciclo iterativo.

Metodología seleccionada

Después del análisis realizado a las metodologías mencionadas anteriormente se selecciona como metodología de desarrollo de software OpenUP porque es apropiada para proyectos pequeños y de bajos recursos. Evita la elaboración de documentación, diagramas e iteraciones innecesarios requeridos en la metodología RUP. Además es la metodología definida por el centro IdeoInformática, para guiar el proceso de desarrollo de software de sus productos

⁴ El Proceso Unificado Racional (Rational Unified Process en inglés, habitualmente resumido como RUP).

⁵ Cualquier persona o entidad que es afectada o concernida por las actividades o la marcha de una organización.

1.9 Conclusiones del Capítulo

En el capítulo se realizó un estudio detallado de los principales conceptos que están asociados al objeto de estudio con el objetivo de ubicar al lector en el dominio del problema. Se desarrolló una amplia investigación sobre algunos de los sistemas para la gestión documental incidiendo en las características y funcionalidades que proveen y determinándose que es necesario el desarrollo de un sistema que permita la gestión de los recursos fotográficos de manera eficiente. Además se definen la metodología de desarrollo, herramientas, así como los lenguajes y tecnologías que sirven de guía para el proceso de desarrollo del software, teniendo como resultado:

- **Metodología de Desarrollo:** OpenUP.
- **Lenguaje de Programación del lado del cliente:** JavaScript 1.6.
- **Lenguaje de Modelado:** UML 6.4.
- **Lenguaje de Marcado:** XML 1.0.
- **Herramienta CASE:** Visual Paradigm 8.0.
- **Sistema Gestor de Bases de Datos:** PostgreSQL 9.1.
- **Sistema Gestor de Contenidos Empresariales:** Alfresco 4.0.
- **Entorno de Desarrollo Integrado:** Eclipse 3.4.2 Ganymede.

Quedando sentadas las bases para dar comienzo a la construcción de la propuesta de solución.

Capítulo 2: Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0.

En el presente capítulo se describe el flujo actual de los procesos involucrados en el sistema. Se abordan las características que deberá tener el núcleo. Además se enumeran los requisitos funcionales y no funcionales que el producto propuesto necesita implementar, permitiendo hacer una concepción general del mismo e identificar mediante diagramas de casos de uso, las relaciones de los actores que se involucran con el sistema y el diseño que se propone como parte de la solución, donde se representa a partir de la descripción detallada de los casos de uso el diagrama de clases del diseño. Se explica conjuntamente el modelo de despliegue y los patrones de diseño utilizados.

2.1 Características del Sistema

El núcleo del sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0 estará montado sobre la versión 4.0 del ECM Alfresco, la cual, provee características que lo hacen más rápido y confiable. Permite compartir recursos a través de los protocolos de Comunicación FTP (en español, Protocolo de transferencia de archivos), CIFS/SAMBA y WebDaV (en español, Edición y versionado distribuidos sobre la web), ya que son tecnologías que han tomado gran desarrollo en la comunicación actual.

El sistema a desarrollar servirá para gestionar los recursos fotográficos de las entidades que como parte de su rutina productiva generan grandes volúmenes de imágenes digitales. Ecumene Pyxel v2.0 debe ser capaz de seleccionar, identificar, describir, representar, recuperar, publicar y compartir los recursos fotográficos, además de generar automáticamente diferentes perfiles a partir de la imagen digital adicionada originalmente al sistema. El sistema a implementar es un módulo que se incorpora al Alfresco, gracias a la estructura modular que este posee. Así mismo el sistema deberá integrarse con un grupo de herramientas externas mediante la extensión de su capa de servicio.

2.2 Análisis de la Estructura Organizativa

Se describen los procesos que se realizan en los medios de comunicación masiva y los roles que intervienen en cada uno de ellos, los cuales serán representados seguidamente a través de los diagramas de proceso, para una mejor personalización del sistema y desarrollo del módulo Ecumene Pyxel v2.0.

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0”

Procesos:

- **Proceso de Identificación:** Tiene como objetivo obtener de una fuente fiable (el fotógrafo) la información relativa al acontecimiento representado, la fecha, la localización geográfica, la descripción de los personajes que aparecen y el material utilizado. Constituyen los primeros elementos que servirán para la descripción posterior de las imágenes (Díaz, 2006).
- **Proceso de Selección:** Delimita el inicio del ciclo de vida. Se selecciona atendiendo a los intereses del medio y a las características, funciones, atributos y valores del material fotográfico. Desde la selección, se recogen las fotografías de las distintas procedencias, para que posteriormente se realice una primera valoración, se describan, y se les pueda almacenar, teniendo en cuenta el tipo de soporte. En procesos cíclicos, la valoración se realiza con el fin de formular acciones en cuanto a la conservación o descarte definitivo del material (Díaz, 2006).
- **Proceso Editorial:** Es el proceso donde se selecciona la imagen que va a ser publicada.
- **Proceso Documental:** Es el proceso que permite la gestión y recuperación de los recursos fotográficos. Realiza una serie de subprocesos cuyo objetivo es lograr su representación formal y almacenamiento.
- **Proceso de Almacenamiento:** El almacenamiento es la forma en que se disponen los materiales fotográficos, cuyo objetivo es organizarlos físicamente para su recuperación efectiva, una vez que hayan sido localizados (Díaz, 2006).

El almacenamiento estará dividido en 3 colecciones (Díaz, 2006):

- ✓ Colección para difusión: Constituida por aquellas fotografías que están disponibles para el uso y publicación. Las imágenes dispuestas aquí son digitales y en formato para ser usadas, disponibles mediante almacenamiento en línea en servidores locales.
- ✓ Colección para conservación: Constituida por los originales analógicos y formatos digitales que indican su autenticidad, que tienen un valor permanente o a largo plazo. Los negativos se consideran originales, así como las fotografías en papel que no tienen respaldo en dicho material. Las imágenes digitales deben contar con los datos de identificación de fotógrafos. Disponible mediante almacenamiento *off-line* (si es digital). Si es analógica, en estantes independientes.
- ✓ Colección congelada: Constituida por aquellas imágenes que cumplen con los requerimientos de calidad física, alta novedad y actualidad permanente para ser conservadas, pero por política del medio o estrategia del país, no es recomendable su uso para la difusión. Está compuesta por imágenes analógicas y digitales. Disponible mediante almacenamiento *off-line* (si es digital). Si es analógica, en estantes independientes.

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Fcumene Pyxel v2.0”

Roles:

- **Fotógrafo:** Es el encargado de tomar las fotos donde ocurren los hechos, debe ser capaz de seleccionar las mejores fotografías y editarlas para ponerlas a disposición de todo el personal que la necesite.
- **Editor:** Es el encargado de editar, publicar y enviar la fotografía a los diferentes espacios según el proceso.
- **Gestor de Información:** Es el encargado de realizar los procesos de selección, procesamiento y almacenamiento.
- **Colaborador:** Tiene la posibilidad de subir imágenes para después ser analizadas en el proceso editorial.
- **Socio:** Es la persona que tiene un contrato con la entidad y colabora en la adición de imágenes al sistema.
- **Administrador:** Es el encargado de darle mantenimiento al sistema, gestiona y controla toda la información referente a los usuarios, imágenes y sistema en general.

Procesos generales por los que transita la Imagen

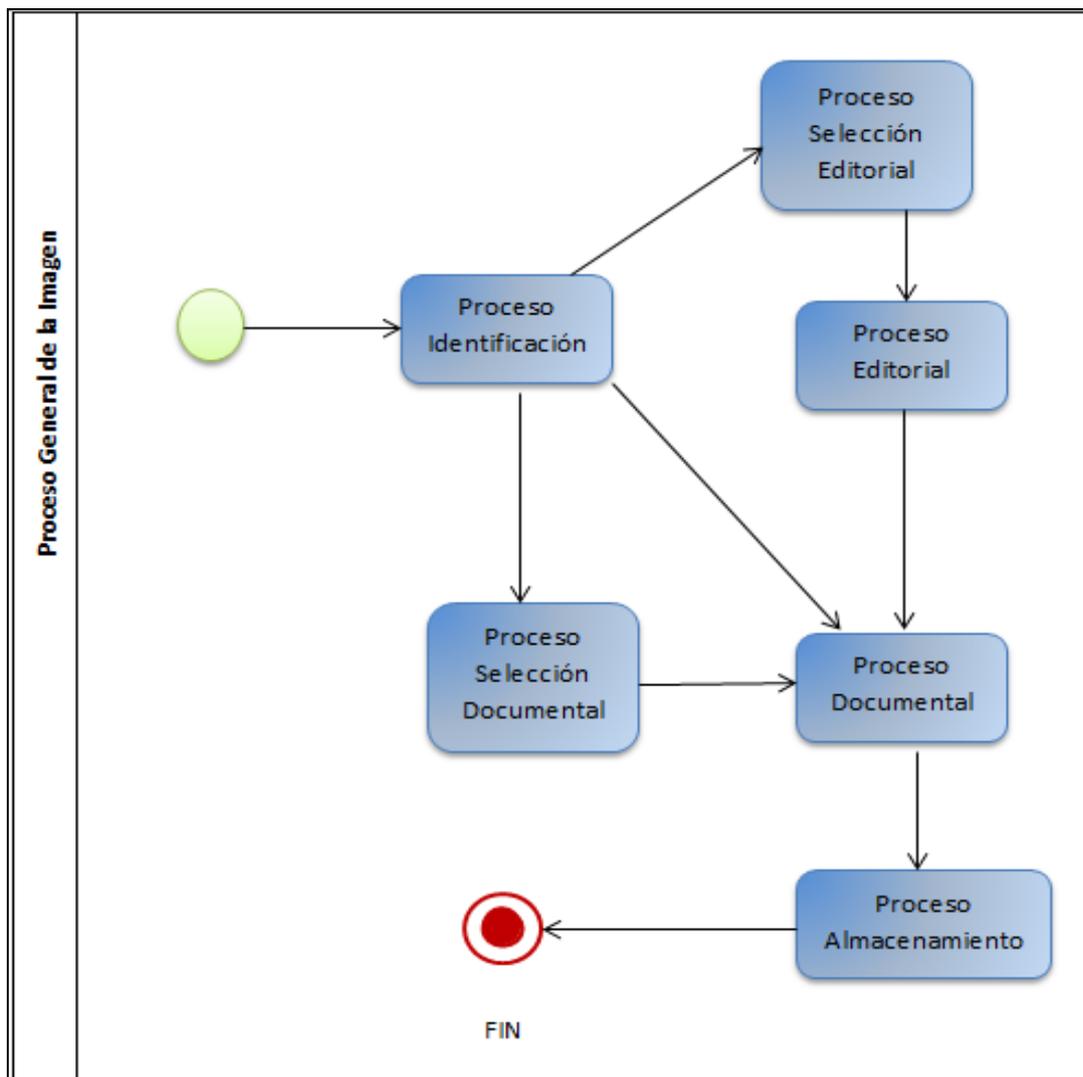


Figura 1 "Diagrama general de la imagen"

Proceso de Identificación de la Imagen

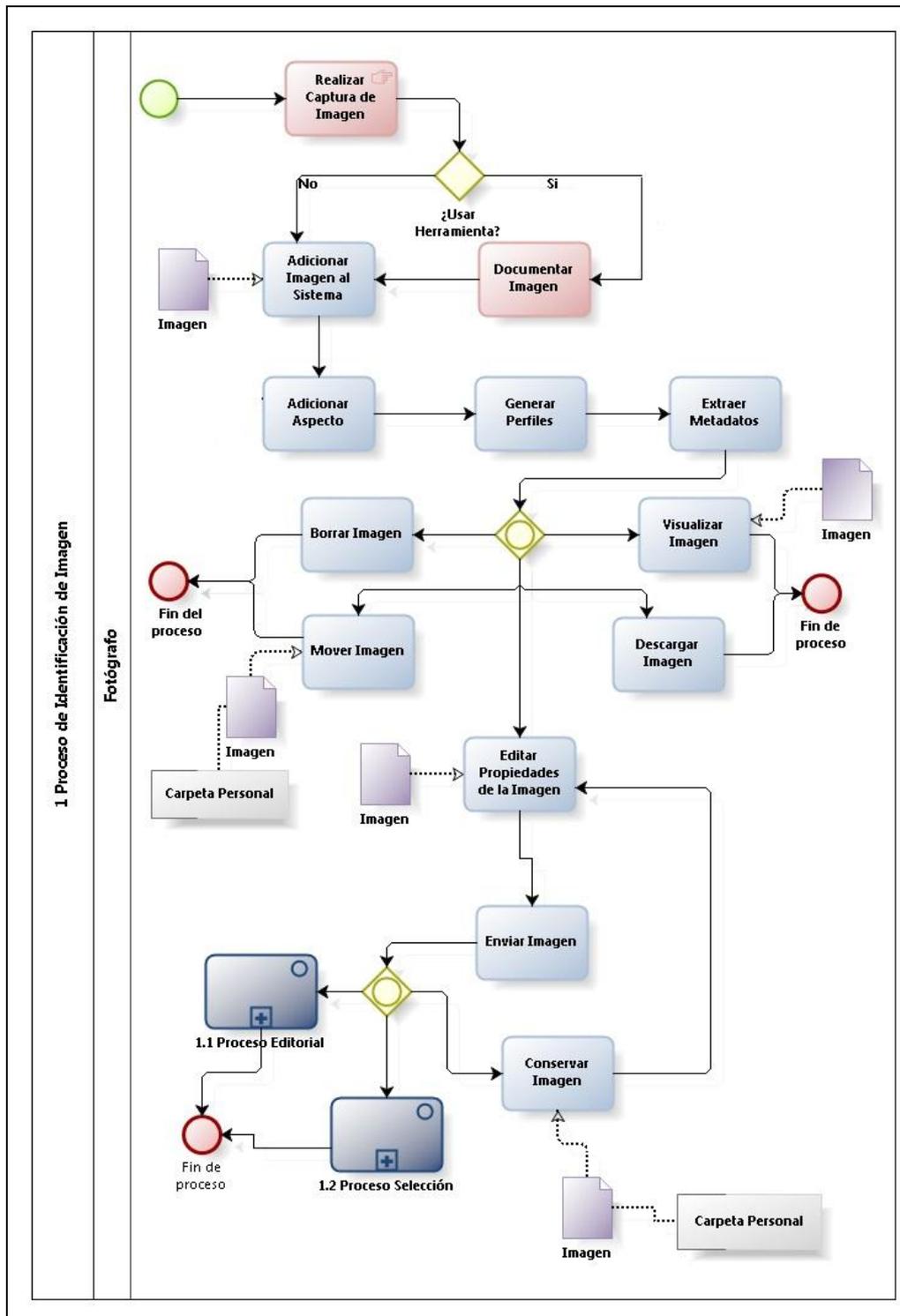


Figura 2 "Diagrama de Proceso de Identificación de la Imagen"

Proceso de Editorial de la Imagen

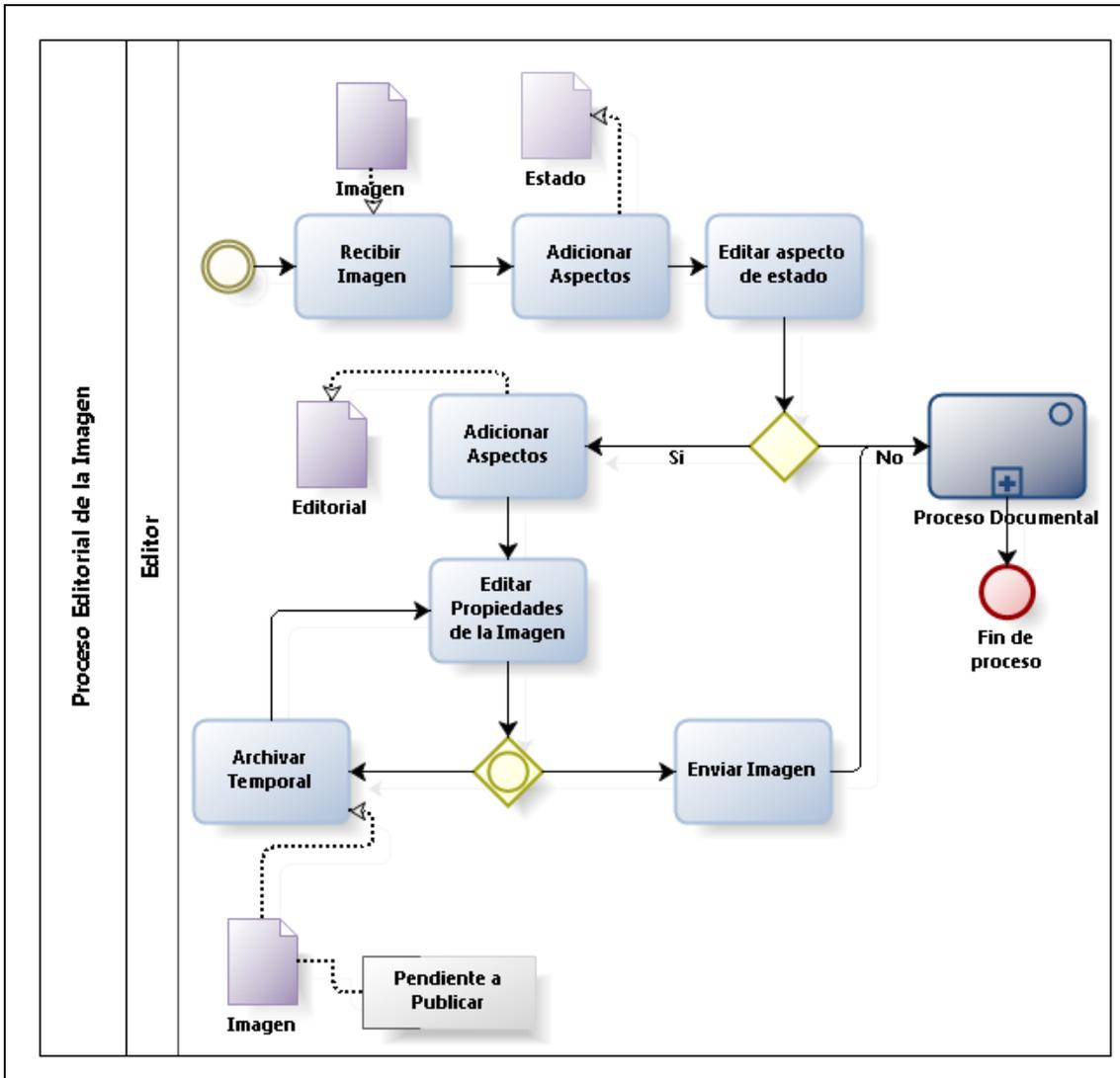


Figura 3 "Diagrama de Proceso de Editorial de la Imagen"

Proceso de Selección de la Imagen

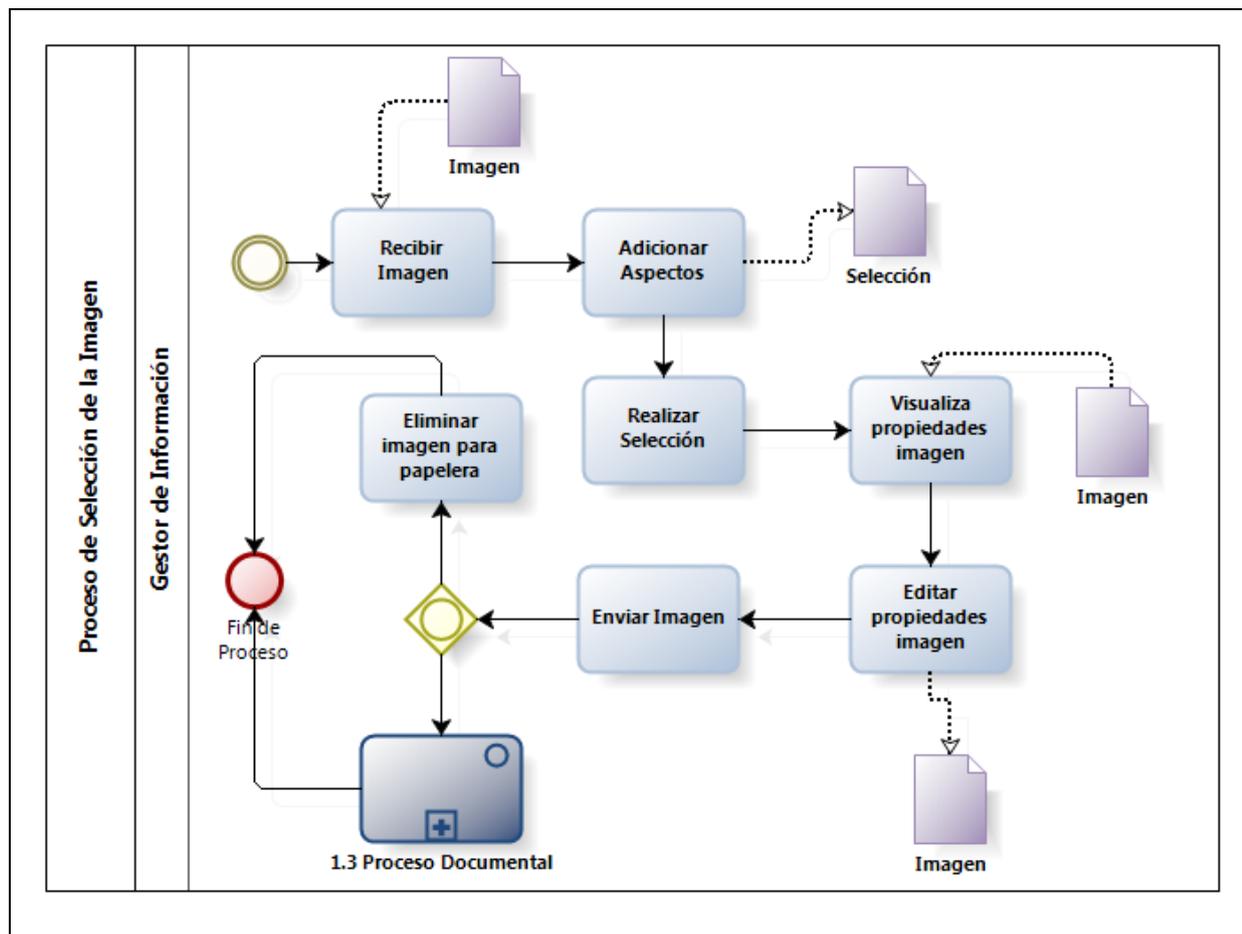


Figura 4 "Diagrama de Proceso de Selección de la Imagen"

Proceso Documental de la Imagen

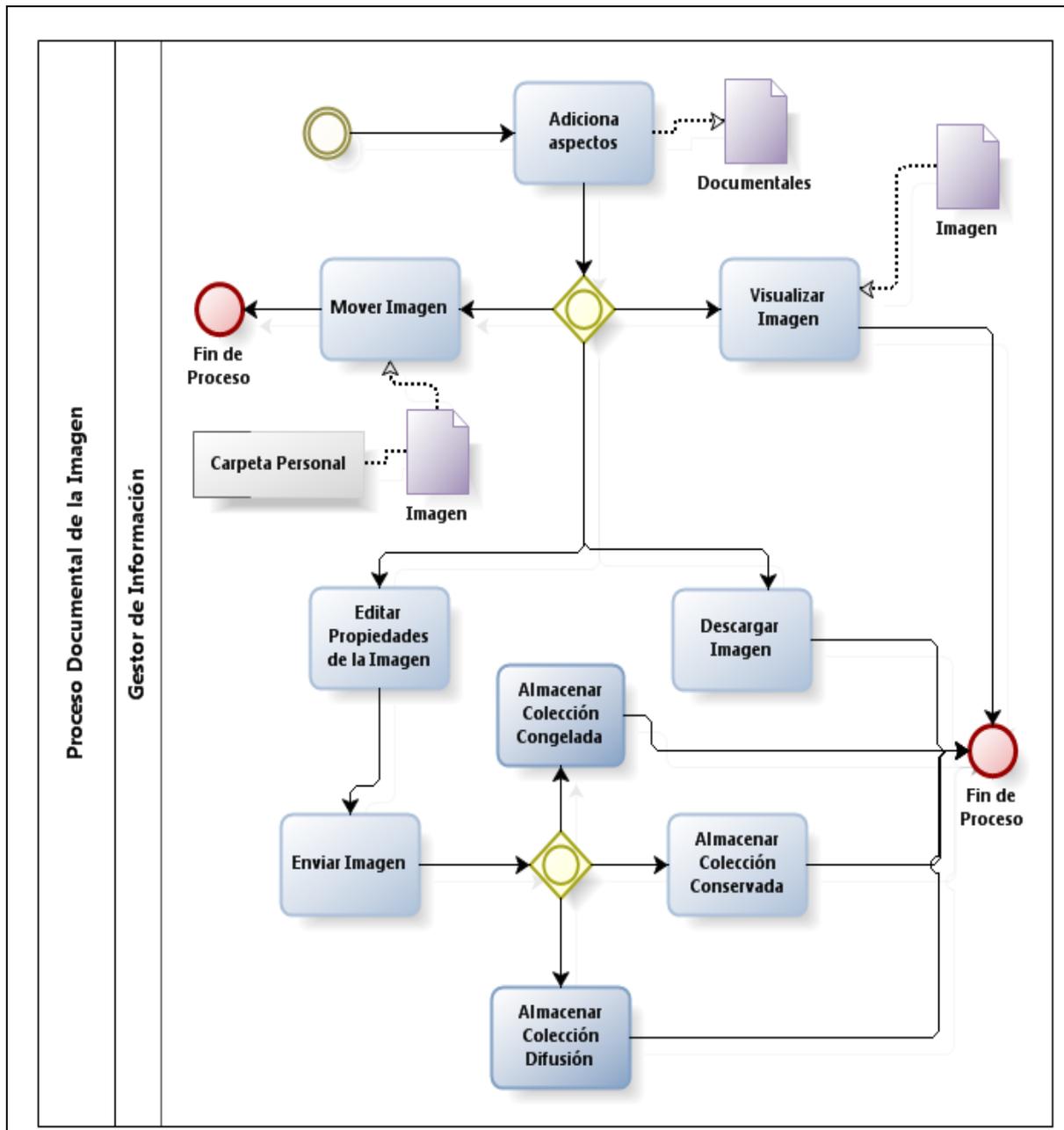


Figura 5 “Diagrama de Proceso de Documental de la Imagen”

2.3 Especificación de los requisitos de Software

Los requisitos de software son condiciones que tienen que ser alcanzadas o poseídas por un sistema o componente de un sistema para satisfacer un contrato, estándar u otro documento impuesto formalmente. Definen lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen. La especificación de requisitos de software establece la base para el acuerdo entre los clientes y los desarrolladores, lo que debe hacer el producto de

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Fcumene Pyxel v2.0”

software, así como lo que no se espera que este haga. Se puede concluir que los requisitos de software son características y funcionalidades que debe cumplir un sistema. Están enfocados hacia todo lo que debe hacer el sistema, el usuario y los miembros del equipo de proyecto. Son clasificados en requisitos funcionales y requisitos no funcionales (Olivera, 2010).

Requisitos Funcionales:

Los requisitos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer (Olivera, 2010).

Se desea que el sistema permita:

- Autenticar usuario
- Adicionar usuario
- Editar usuario
- Eliminar usuario
- Filtrar cuentas de usuario
- Listar cuentas de usuario
- Cambiar contraseña del usuario
- Visualizar cuota del usuario
- Visualizar perfil del usuario
- Buscar grupo
- Buscar usuario en todos los grupos
- Visualizar grupos y usuarios
- Adicionar grupos
- Eliminar grupos
- Asignar usuario a grupos
- Eliminar usuario en todos los grupos
- Visualizar rol
- Seleccionar rol
- Crear estructura organizacional
- Adicionar regla del negocio
- Editar regla del negocio
- Eliminar regla del negocio
- Deshabilitar regla del negocio
- Adicionar contenido
- Editar contenido
- Eliminar contenido
- Visualizar contenido
- Auditar contenidos
- Descargar contenidos
- Copiar contenidos
- Mover contenidos

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales *fcumene Pyxel v2.0*”

- Cortar contenidos
- Exportar/importar contenidos
- Actualizar contenidos
- Configurar envío de imágenes por FTP, SMB/CIFS, WEBDAV

Requisitos no Funcionales:

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener para que el mismo sea usable, rápido, fiable, entre otras características. Los requisitos no funcionales forman una parte significativa de la especificación. Permiten la interacción entre clientes y usuarios para que puedan valorar las características no funcionales del producto (Olivera, 2010). Los requerimientos no funcionales son clasificados en diferentes categorías. A continuación se muestran los que se definieron para la presente solución.

Requerimientos de Software

- Sistema Operativo GNU-Linux Distribución Ubuntu 12.04 LTS, Máquina Virtual de Java openjdk6.
- La estación de trabajo cliente debe tener instalado el navegador Mozilla Firefox 18.x o superior.

Requerimientos de Hardware Mínimo

- Microprocesador Core 2Duo 2.20GHZ HDD 160GB 1GB RAM

Restricciones en el diseño y la implementación

- Gestor de contenidos empresariales Alfresco versión 4.0.
- Lenguaje de programación Java.

Requerimientos de apariencia o interfaz externa

- Usar la interfaz de Alfresco Explorer

Requerimientos de Seguridad

- Se controlaran los permisos de accesos, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema.
- Se utilizaran mecanismos de autenticación segura mediante el empleo del SSL.

Requerimientos de Usabilidad

- Se utilizarán mecanismos de internacionalización de los textos en español.

Requerimientos de Soporte

- Realizar Pruebas, Mantenimiento e Instalaciones necesarias para lograr el mejoramiento.
- Establecer procedimiento para garantizar redundancia en las bases de datos y asegurar la recuperación ante fallos.

Requerimientos Legales

- Las herramientas seleccionadas para el desarrollo del producto están respaldadas por licencias libres, bajo las condiciones de software libre.

Requerimientos de Documentación

- La documentación resulta requisito indispensable como evidencia de la evolución del proyecto.

2.4 Definición de los Casos de Uso del Sistema

Un caso de uso ilustra el ciclo de transacciones desarrolladas por un sistema en respuesta a una acción que realiza el actor, de esta forma ayudan a describir lo que el sistema debe hacer (Pressman, 2002). A continuación se muestran los artefactos que se han creado durante esta fase, para la correcta obtención de requisitos:

Definición de los actores

Actores	Justificación
Usuario	Visualiza la información contenida en cada uno de sus espacios, además para poder acceder al sistema debe autenticarse mediante el LDAP.
Administrador	Es el encargado de darle mantenimiento al sistema, gestiona y controla toda la información referente a los usuarios, imágenes y sistema en general.
Fotógrafo	Encargado de tomar las fotografías y subirlas a su espacio para un posterior análisis en el proceso editorial.
Gestor de la Información	Encargado de aprobar y clasificar las fotografías.
Editor	Es la persona que decide sobre la publicación o no de las fotografías y las condiciones de esa aparición pública de la imagen captada.
Proveedor de imágenes	Tiene la tarea de subir fotos y artículos para su posterior análisis.

Tabla 1 “Actores del Sistema”

Diagrama de casos de uso del Sistema

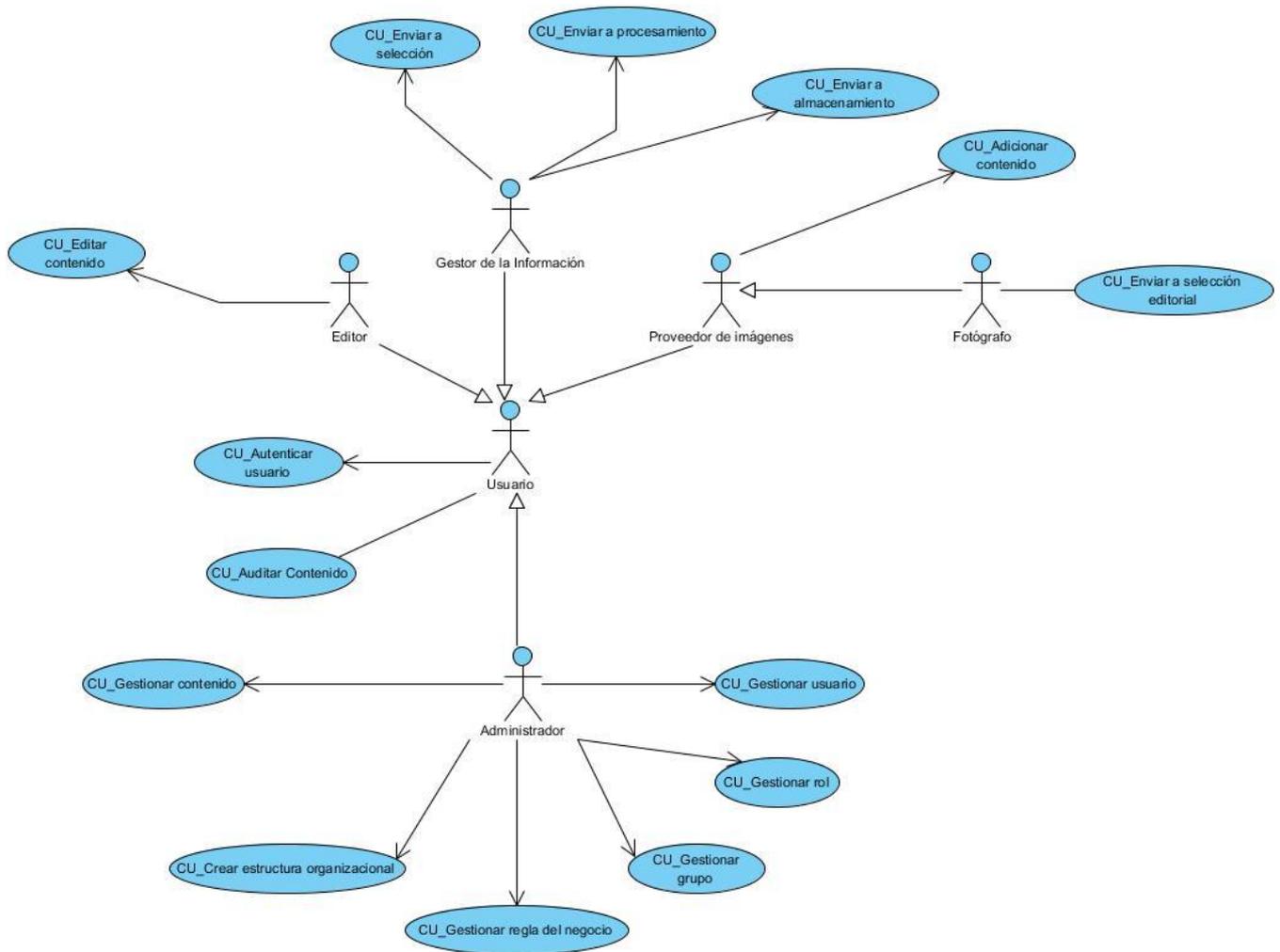


Figura 6 “Diagrama de Casos de uso”

Descripción textual de los casos de uso

A continuación serán descritos los casos de uso del sistema que fueron modelados en el diagrama. En cada descripción se especifica el propósito general de cada uno de ellos, el resumen de su funcionamiento, así como las condiciones que deben existir para que estos ocurran.

Gestionar Contenido

Objetivo	Subir, editar, eliminar y mostrar contenidos.
Actores	Administrador (Inicia)
Resumen	El CU se inicia cuando un usuario con permisos de administración ya autenticado en el sistema es capaz de gestionar contenidos en el sistema.
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El actor tiene que estar autenticado.

Capítulo 2: "Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcmene Pyxel v2.0"

Postcondiciones	Nuevo contenido registrado en el sistema. Contenido eliminado. Contenido editado.	
Flujo de eventos		
Flujo básico Subir contenido		
	Actor	Sistema
1	Escribe en su navegador la siguiente dirección: https://ip_servidor:puerto/alfresco/service .	Muestra una ventana donde el usuario debe llenar los campos usuario y contraseña.
2	El usuario introduce su nombre de usuario y la contraseña.	Muestra la familia de servicios del Alfresco.
3	Busca la familia de servicios nombrada Epyxel2 Services.y accede a ella	Muestra una serie de servicios.
4	Busca el servicio upload y presiona el vínculo para acceder.	Muestra una nueva interfaz con un formulario.
5	Busca el contenido a añadir, rellena los otros campos y presiona el botón Subir archivo.	Muestra el siguiente mensaje: "El archivo se ha subido correctamente"
6		Termina el CU.
Sección 2: "Editar contenido"		
Flujo básico Editar contenido		
	Actor	Sistema
1	Escribe en su navegador la siguiente dirección: https://ip_servidor:puerto/alfresco/service .	Muestra una ventana donde el usuario debe llenar los campos usuario y contraseña.
2	El usuario introduce su nombre de usuario y la contraseña.	Muestra la familia de servicios del Alfresco.
3	Busca la familia de servicios nombrada Epyxel2 Services y presione el vínculo para acceder.	Muestra una serie de servicios.
4	Busca el servicio modify y presiona el vínculo para acceder.	Muestra un mensaje de error.
5	En la barra de direcciones del navegador debe llenar los campos {uid},{props} (ejemplo en props {"cm:name": "Nombre", "cm:title": "Titulo" }) y {lastName}.	Muestra los detalles del contenido modificado.
6		Termina el CU.
Sección 3: "Eliminar contenido"		
Flujo básico Eliminar contenido		
	Actor	Sistema
1.	Escribe en su navegador la siguiente dirección: https://ip_servidor:puerto/alfresco/service .	Muestra una ventana donde el usuario debe llenar los campos usuario y contraseña.
2.	El usuario introduce su nombre de usuario y la contraseña.	Muestra la familia de servicios del Alfresco.

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcmene Pyxel v2.0”

3.	Busca la familia de servicios nombrada Epyxel2 Services y presione el vínculo para acceder.	Muestra una serie de servicios.
4.	Busca el servicio delete y presiona el vínculo para acceder.	Muestra el siguiente mensaje de error. "No se ha realizado la operación"
5.	En la barra de direcciones del navegador debe llenar el campo {uuid} o {node}.	Muestra el mensaje “El elemento ha sido eliminado”.
6.		Termina el CU.
Sección 2: “Mostrar contenido”		
Flujo básico Mostrar contenido		
	Actor	Sistema
1.	Escribe en su navegador la siguiente dirección: https://ip_servidor:puerto/alfresco/service .	Muestra una ventana donde el usuario debe llenar los campos usuario y contraseña.
2.	El usuario introduce su nombre de usuario y la contraseña.	Muestra la familia de servicios del Alfresco.
3.	Busca la familia de servicios nombrada Epyxel2 Services y presione el vínculo para acceder.	Muestra una serie de servicios.
4.	Busca el servicio detail y presiona el vínculo para acceder.	Muestra un mensaje de error
5.	En la barra de direcciones del navegador debe llenar el campo {uuid} o {node}.	Muestra los detalles del contenido adicionado.
6.		Termina el CU.

Tabla 2“Descripción textual CU Gestionar Contenido”

Auditar Contenido

Objetivo	Auditar contenidos del sistema.	
Actores	Administrador:(Inicia)	
Resumen	El CU permite auditar un contenido del sistema.	
Complejidad	Media	
Prioridad	Secundario	
Precondiciones	Usuario del sistema autenticado.	
Postcondiciones	Contenido auditado.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
1.	Al entrar al espacio de empresa, se dirige hacia los elementos de contenidos y da clic	Despliega las acciones.

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcmene Pyxel v2.0”

	en el vínculo más acciones.	
2.	Accede a la acción pre-visualizar en plantilla.	Muestra una nueva vista.
3.	Selecciona la opción show_audit.ftl.	Muestra una tabla con la información de auditoría del contenido.
4.		Termina CU.

Tabla 3 “Descripción textual CU Auditar Contenido”

Gestionar Grupo

Objetivo	Debe ser capaz de crear y eliminar grupo, adicionar usuario al grupo y eliminarlo.	
Actores	Usuario. (Inicia)	
Resumen	El CU se inicia cuando un usuario ya autenticado en el sistema da clic en el vínculo Gestionar Grupos de usuarios.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	Usuario del sistema autenticado.	
Postcondiciones	Grupos de usuarios creados, editados o eliminados.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
1.	Al estar autenticado en el sistema, se dirige a la consola de Administración y da clic en el vínculo Gestionar Grupos de usuarios.	Muestra la interfaz con las demás funcionalidades.
2.		Termina el CU.
Sección 1: “Crear grupo”		
Flujo básico		
	Actor	Sistema
1.	Una vez posicionado en la página de Gestionar Grupos de usuarios, debe presionar el vínculo Crear Grupo.	Responde con un formulario donde se tiene que especificar el campo identificador.
2.	Pulsa el botón Crear grupo para guardar los cambios	Se crea el grupo en el sistema.
3.		Termina el CU.
Flujos alternos		
	Actor	Sistema
2.1	Cancelar en caso de querer detener la acción.	
Sección 2: “Adicionar usuario al grupo”		

Capítulo 2: “Análisis y Diseño del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcmene Pyxel v2.0”

Flujo básico		
	Actor	Sistema
1.	Para adicionar usuario(s) al grupo o a los grupos, debe estar posicionado en la página de Gestionar Grupo de usuarios.	Responde con un formulario donde se tiene que especificar el campo identificador.
2.	Si conoce el identificador del grupo procede a su escritura en la caja de diálogo.	Muestra una caja de diálogo para escribir el identificador y realiza la búsqueda.
3.	Pulsa el vínculo Añadir Usuario, localizado en la parte central e inferior del grupo seleccionado	Responde con otra caja de diálogo para que usted introduzca el nombre del usuario.
4.	Introduce el nombre.	Realiza la búsqueda en el sistema.
5.	Se adiciona dicho usuario y luego se accede al botón Aceptar	El sistema guarda los cambios.
6.		Termina el CU.

Flujos alternos		
	Actor	Sistema
3.1	Cancelar en caso de querer detener la acción.	.
5.1	Cancelar en caso de querer detener la acción.	

Sección 3: “Eliminar usuario del grupo”

Flujo básico		
	Actor	Sistema
1.	Debe estar posicionado en la página de Gestionar Grupo de usuarios.	El sistema responderá con un formulario donde se tiene que especificar el campo identificador.
2.	Si conoce el identificador del grupo procede a su escritura.	Muestra una caja de diálogo y realiza la búsqueda.
3.	Si es recuperado satisfactoriamente entonces accede al grupo.	Mostrará el o los usuario(s) que pertenecen a dicho grupo.
4.	Pulsa el vínculo Eliminar para completar la acción.	
5.		Termina el CU.

Sección 4: “Eliminar grupos”

Flujo básico		
	Actor	Sistema
1.	Para eliminar un grupo, debe estar posicionado en la página de Gestionar Grupo de usuarios.	Responde con un formulario donde se tiene que especificar el campo identificador.
2.	Si conoce el identificador del grupo procede a su escritura.	Muestra una caja de diálogo y realiza la búsqueda.
3.	Si es recuperado satisfactoriamente entonces procede a pulsar el botón Borrar grupo.	Muestra un mensaje de confirmación de eliminación.

4.	Pulsa el botón Borrar.	Elimina el grupo seleccionado
5.		Termina el CU.
Flujos alternos		
	Actor	Sistema
4.1	En caso contrario pulsar el botón Cancelar para detener la acción.	

Tabla 4 “Descripción textual CU Gestionar grupo”

2.5 Arquitectura

La arquitectura de software es la organización fundamental de un sistema encarnado en sus componentes, las relaciones de los componentes con cada uno de los otros y con el entorno y los principios que orientan su diseño y evolución.

Luego de realizar un análisis de la solución propuesta y considerando que su finalidad es extender el modelo de contenidos del ECM Alfresco, se utiliza como arquitectura para este módulo la misma filosofía que implementa el Alfresco en su arquitectura mediante el estilo arquitectónico en capas.

Alfresco divide su arquitectura en tres niveles o capas: la capa de presentación, mediante la cual se le proporciona al usuario acceso visual al sistema a través de una interfaz web a la que acceden mediante los navegadores web; la capa de aplicación, donde se separa a nivel lógico el repositorio⁶ (Alfresco *Repository*) de las aplicaciones (Alfresco *Applications*) que acceden al sistema. Finalmente se encuentra la capa de almacenamiento, aquí es donde se asegura la persistencia de los datos subidos al repositorio, el contenido se almacena en el sistema de ficheros (*File System*) mientras que los metadatos asociados son almacenados en la base de datos (*Database*) (Raul, 2011).

⁶ Lugar donde se almacenan los documentos.

A continuación se muestra la arquitectura utilizada:

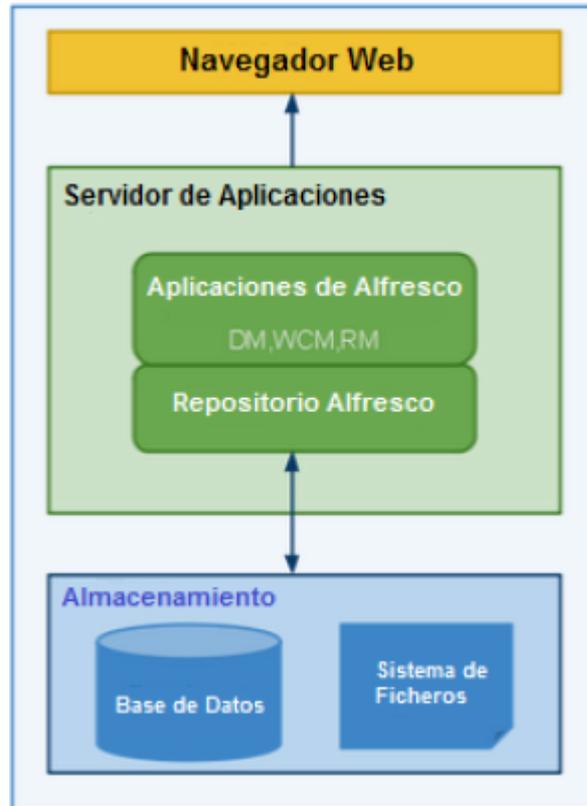


Figura 7 “Arquitectura del ECM Alfresco”

El modelo de arquitectura posibilita que las capas puedan estar distribuidas en el mismo sistema físico o no, proporcionando un modelo perfectamente adaptable a las necesidades y posibilidades tecnológicas del entorno donde se implemente; aunque se puede incrementar la escalabilidad, la tolerancia a fallos y el rendimiento si se distribuyen las capas entre múltiples sistemas físicos. Ofrece una mayor sencillez a los clientes, la centralización de los aspectos de seguridad. No se replica la lógica del negocio en los clientes, lo que significa que las modificaciones y mejoras sean aprovechadas automáticamente por el conjunto de usuarios reduciendo los costes de mantenimiento, permite aislar los cambios en tecnologías a ciertas capas para reducir el impacto en el sistema (Rojas, 2008).

El núcleo del sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0 utiliza el estilo de arquitectura Modular que define Alfresco, pues está diseñado como un sistema compuesto por elementos separados que se conectan preservando sus relaciones. Esta arquitectura ofrece la posibilidad de instalar solo aquellas capacidades que se necesiten y de extender el sistema incorporando nuevas funcionalidades requeridas.

2.6 Patrones de Diseño

El ECM Alfresco gestiona sus procesos y teniendo en cuenta que el módulo Ecumene Pyxel es una extensión del modelo de contenidos del Alfresco, los patrones de diseño son heredados del ECM Alfresco. A continuación se exponen los utilizados en el desarrollo de la solución (Figuroa, 2012):

- *Session Facade*
Define un componente de negocio de alto nivel que contiene y centraliza las interacciones complejas entre los componentes de negocio de nivel inferior. Proporciona a los clientes una única interfaz para la funcionalidad de una aplicación o un subconjunto de la aplicación. Separa los componentes de nivel inferior de negocio, uno del otro, proporcionando un diseño más flexible y comprensible.
- *Data Access Object (DAO)*
Se encarga de administrar la conexión con la base de datos tanto para obtener datos como para almacenarlos. Proporciona una técnica para la separación de la persistencia de objetos y la lógica de acceso a datos de cualquier mecanismo de persistencia en particular o de la API. Así como flexibilidad para cambiar de un mecanismo de aplicación de la persistencia en el tiempo sin la necesidad de lógica de la aplicación de reingeniería, que interactúa con el nivel de DAO.
- *State*
Permite que el comportamiento de un objeto sea una función de su estado. Esto significa que al modificar el estado del objeto, este cambia su comportamiento.

Luego de realizar una breve descripción sobre cada uno de los patrones utilizados, se muestra el modo en que se emplea cada uno de ellos en el diseño de la propuesta de solución. Tanto las peticiones de los usuarios como las respuestas del sistema son gestionadas a través de una única interfaz, cumpliendo con lo que plantea el patrón *Session Facade*. Una vez creados o adicionados los contenidos a la aplicación, su comportamiento es controlado a través de estados que se le adicionan como aspectos a los diferentes tipos de contenidos, lo cual se encuentra en correspondencia con el principio del patrón *State*. Finalmente el patrón DAO se emplea para gestionar la conexión a la base de datos, ya que proporciona un mecanismo para la separación de la lógica de acceso a datos y los objetos persistentes.

2.7 Diagramas de clases de diseño

Un diagrama de clases representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas. Nos sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de convencimiento. Un diagrama de clases está compuesto por los siguientes elementos: Clase: atributos, métodos y visibilidad. Relaciones: Herencia, Composición, Agregación, Asociación y Uso. Son la base para los posteriores diagramas de componentes y de despliegue (Olivares, 2009).

A continuación se muestra el diagrama de clases (Ver Figura 8) correspondiente al núcleo Ecumene Pyxel v2.0. Mediante el diagrama se realiza una representación gráfica de la jerarquía y las relaciones que se generan a partir de la extensión del modelo de contenidos del Alfresco. El modelo definido por el núcleo del Alfresco se representa a través de las clases *Content*, *Aspect*, *Properties* y *Folder*. Fotografía, Galería Temática, Unidad Documental, Carpeta, Estado Fotografía, Estado Fotografía Asistente, Clasificación Unidad Documental, Estado Unidad Documental, Selección Foto Editorial, Estado Foto Almacenamiento, Pie de foto, Publicar Foto Editorial, Estado Espacio Selección, Estado UD Procesamiento, Estado Foto Procesamiento, Estado Foto Difusión, Estado Foto Conservada, Estado UD Conservada, Estado Foto Congelada, Estado UD Congelada, Estado GT Galería, Atributos Conceptuales y Atributos Ejemplares, representan la extensión del modelo de contenidos que se introducen en el núcleo del Alfresco a partir del desarrollo del módulo Ecumene Pyxel v2.0.

Cada una de estas clases está relacionada con el modelo original del núcleo del Alfresco en dependencia de la función que representa dentro del modelo de contenido. La clase Fotografía hereda de la clase *Content*, ya que es un tipo de contenido que posee atributos específicos para su definición. Fotografía también posee una relación de herencia con la clase *Aspect* porque además de ser un tipo de contenido es un aspecto que puede ser aplicado a los contenidos. Las clases Galería Temática, Unidad Documental y Carpeta solo heredan de *Folder*, ya que las tres representan carpetas. En el caso de las clases Estado Fotografía, Estado Fotografía Asistente, Clasificación Unidad Documental, Estado Unidad Documental, Selección Foto Editorial, Estado Foto Almacenamiento, Pie de foto, Publicar Foto Editorial, Estado Espacio Selección, Estado UD Procesamiento, Estado Foto Procesamiento, Estado Foto Difusión, Estado Foto Conservada, Estado UD Conservada, Estado Foto Congelada, Estado UD Congelada, Estado GT Galería, Atributos Conceptuales y Atributos Ejemplares heredan de *Aspect* ya que están definidas como un aspecto de contenido.

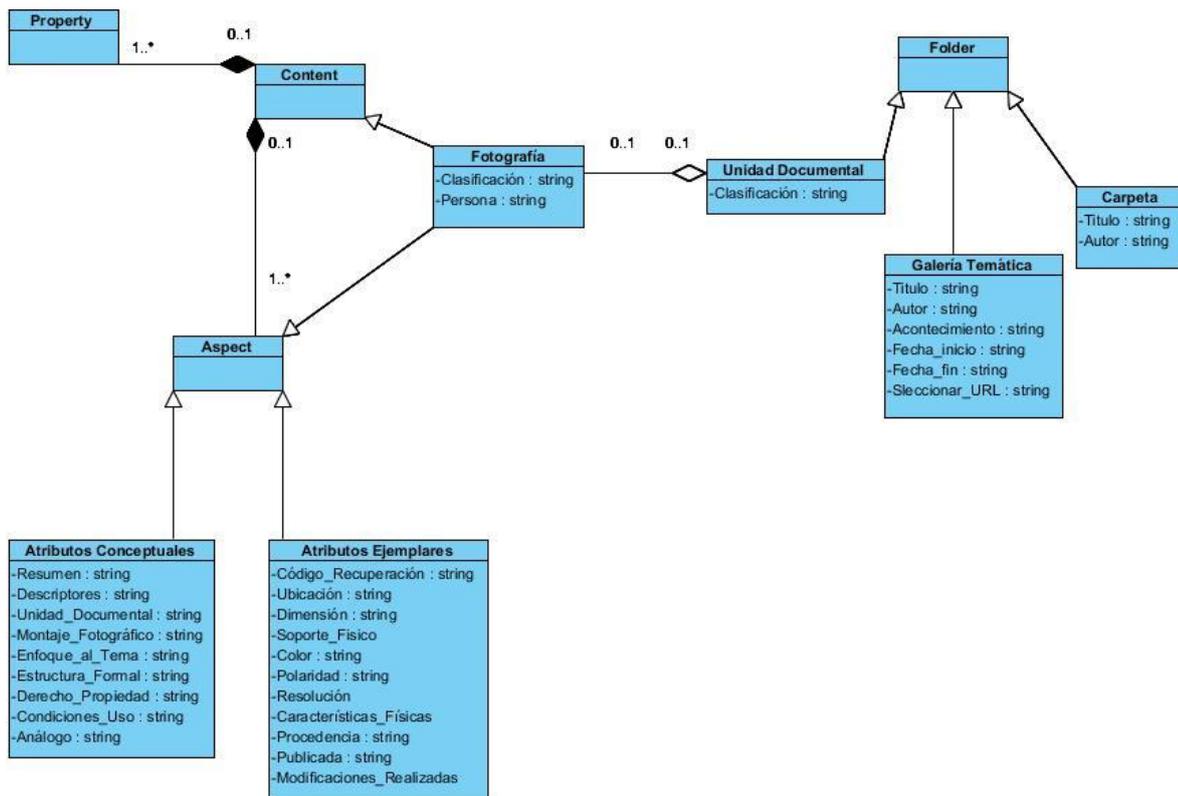


Figura 8 “Diagrama de clases”

2.8 Modelo de Despliegue

El modelo de despliegue se utiliza para capturar los elementos de configuración del procesamiento y las conexiones entre dichos elementos. El mismo incluye un artefacto fundamental, el Diagrama de Despliegue, el cual se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes. Es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo (Chacón, 2009).

En él se especifican tres elementos fundamentales: los nodos, los dispositivos y los conectores (Chacón, 2009).

- Los nodos describen los elementos de procesamiento con al menos un procesador, memoria y cualquier otro dispositivo.
- Los dispositivos son nodos que no tienen capacidad de procesamiento en el nivel de abstracción que se modela.

- Los conectores expresan el tipo de protocolo utilizado en el resto de los elementos del modelo.

Para el uso del sistema solamente es necesaria una PC Cliente del lado del cliente, que accederá al servidor del Alfresco mediante un navegador web y emplearán los protocolos HTTPs, FTP y CIFS. Cuenta con un servidor donde se encuentra el sistema gestor de bases de datos que contendrá todos los datos necesarios para la aplicación, a este servidor podrán conectarse mediante el protocolo TCP/IP. A continuación se muestra el diagrama de despliegue correspondiente al sistema que se desea implementar.



Figura 9 “Modelo de despliegue”

2.9 Conclusiones del Capítulo

Concluido el capítulo se han dado los primeros pasos necesarios para comenzar a implementar el sistema. Durante el transcurso del mismo se ha obtenido una visión sobre lo que debe hacer el núcleo a desarrollar, centrándose en los requisitos funcionales, los procesos por los que transita la imagen y por otro lado cómo el sistema cumple con los objetivos propuestos, enfocándose en los requisitos no funcionales. Una vez logrado esto puede decirse que se ha refinado y definido la arquitectura del sistema; permitiendo mediante el análisis y diseño un entorno de implementación más consistente.

Capítulo 3: Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel v2.0.

En el presente capítulo se describen los aspectos referentes a la implementación de la propuesta de solución, se expone cómo los elementos del modelo de diseño son implementados en términos de componentes. Presenta además el diseño de las pruebas del sistema y finalmente se exponen los resultados obtenidos.

3.1 Diagrama de Componentes

El diagrama de componentes define cómo las clases, artefactos y otros elementos de bajo nivel, se unen para formar componentes de alto nivel y las conexiones entre ellos. Los componentes son artefactos de software compilados que trabajan acoplados para brindar el comportamiento requerido dentro de las restricciones definidas en el proceso de captura de requisitos (Ltd, 2007).

Un diagrama de componentes tiene un nivel más alto de abstracción que un diagrama de clase, usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, como eventualmente un componente puede comprender una gran porción de un sistema. Los componentes se representan como un clasificador rectangular con la clave «componente», opcionalmente el componente se puede mostrar como un rectángulo con un icono de componente en la parte superior derecha (Ltd, 2007).

Para el correcto funcionamiento del sistema se definen varios paquetes, los cuales conforman la lógica del negocio, la vista y la modelación de los datos.

A continuación se presenta el diagrama de componentes para el núcleo Ecumene Pyxel v2.0 que se propone desarrollar. El cual estará formado por el paquete principal llamado EPyxel2, que estará compuesto a su vez por un paquete SDK donde están contenidas las librerías necesarias a utilizar. En el paquete Config, se define toda la estructura de componentes y paquetes necesarios para formar grupos, carpetas, permisos, tipos de contenidos y roles que serán incluidos en el sistema mediante ficheros XML. También existe el componente build.xml el cual es imprescindible para el funcionamiento y compilación del módulo EPyxel2.

Capítulo 3: "Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0"

Paquete	Descripción
EPyxe12	Es el paquete fundamental, incluye toda la estructura de la propuesta de solución, además de los ficheros module.properties, donde se describen las propiedades del módulo y build.xml, mediante la ejecución de este fichero se construye toda la estructura de carpetas una vez que se realiza el despliegue del módulo.
SDK	Contiene todas las librerías necesarias para la extensión del modelo de contenidos de Alfresco, así como las librerías necesarias para trabajar con el módulo EPyxe12.
Config	Contiene toda la estructura de carpetas correspondiente a la definición del módulo EPyxe12.
Model	Contiene el fichero permissionDefinitions.xml mediante el cual se extiende el modelo de permisos atendiendo a los procesos identificados en el capítulo anterior.
Bootstrap	Se define la estructura del módulo mediante la declaración de espacios, así como la accesibilidad de cada uno de los grupos de usuarios a cada uno de estos espacios mediante el fichero epyxe12-spaces.xml.
Messages	Se definen las propiedades correspondientes a cada uno de los espacios mediante el fichero epyxe12-bootstrap-spaces.properties.

Tabla 5 "Descripción de Componentes"

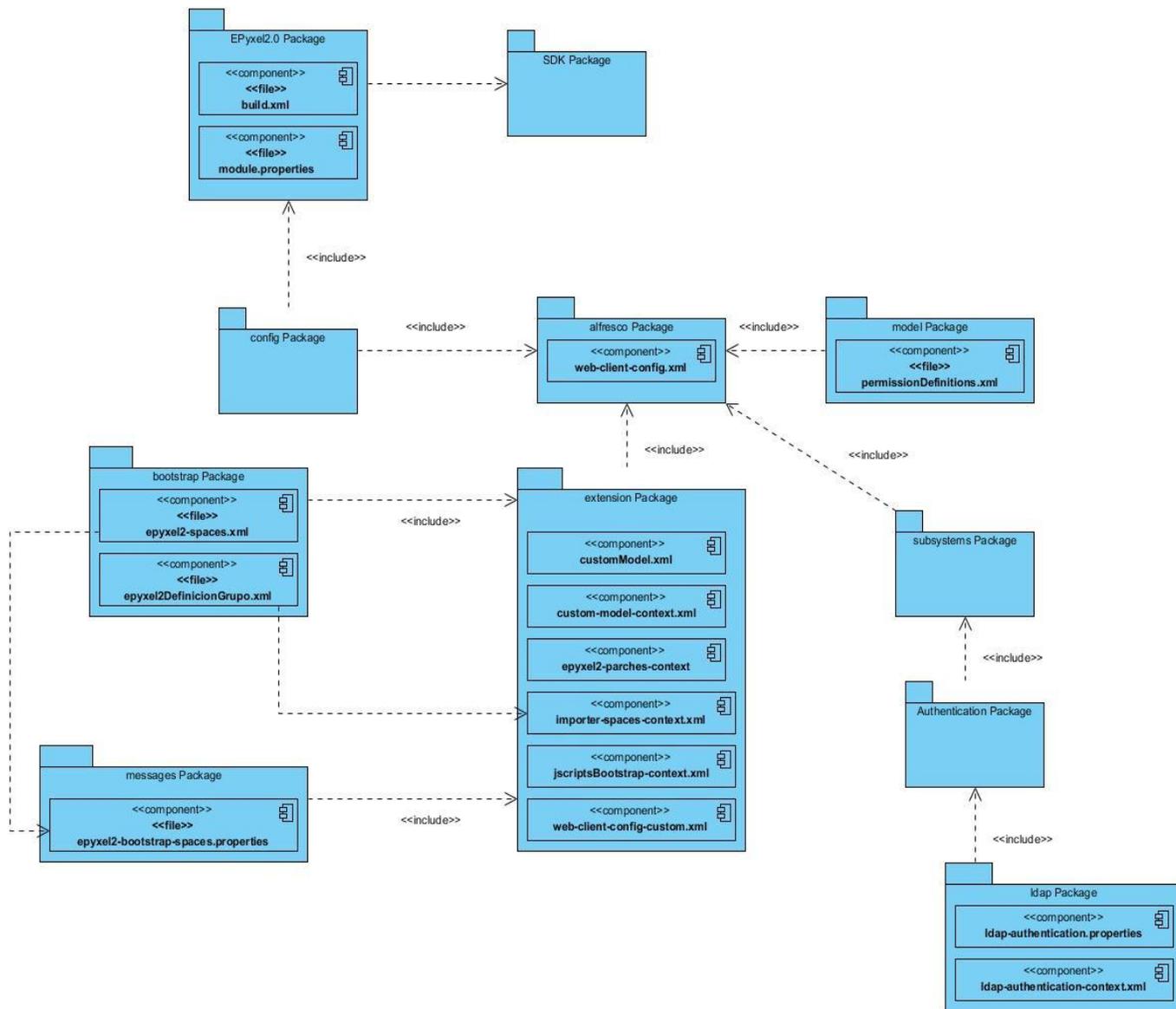


Figura 10 “Diagrama de Componentes”

3.2 Código Fuente

El código fuente de un programa informático es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa, el código fuente escrito por el programador es un programa que no puede ser ejecutado directamente por la computadora, sino que necesita de traductores que si pueden ser ejecutados, como son los llamados compiladores, ensambladores, intérpretes y otros sistemas de traducción⁷.

⁷ <http://www.mastermagazine.info/termino/6394.php> codigo fuente.

3.2.1 Estándares de Codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Un código fuente completo debe reflejar un estilo legible, como si todo el código lo hubiera programado una sola persona. Cuando comienza un proyecto, se debe establecer inicialmente un estándar de codificación para asegurar el trabajo de forma coordinada. El estándar de codificación debe ser capaz de establecer como se debe operar con la base de código existente. A continuación se describe el estándar de codificación empleado para la implementación del módulo propuesto.

Declaración XML

Inicialmente, la declaración XML aparece normalmente como la primera línea en un documento XML. La declaración XML no es necesaria; sin embargo, si se utiliza tiene que ser la primera línea en el documento y ningún otro contenido o espacio en blanco deben precederla.

```
<?xml version="1.0" encoding="UTF-8"?>
```

Para que el estándar antes mencionado cuente con la estructura requerida es necesario que se auxilie de los siguientes elementos:

- **Tamaño de las líneas**

Las líneas deben ser de máximo 99 caracteres. Si es necesario partir la línea, la siguiente debe alinearse dejando doble sangría.

- **Comentarios**

El código debe comentarse empleando la estructura siguiente:

```
<!-- Descripción del comentario -->
```

- **Nombramiento de los constraint, los property y los aspect**

Inicialmente el nombre de los constraint, los property y los aspect deben declararse en minúscula y si está compuesto por dos o más palabras, en el intervalo entre palabra y palabra debe existir un guión bajo.

Ejemplo: `<constraint name="sc: estado_foto_asistente"> </constraint>`

```
<property name="sc: clasificacion1"> </property>
```

```
<aspect name="sc: perfiles"></aspect>
```

- **Nombramiento de los type**

Los type deben empezar con minúscula y si están compuestos por dos palabras, la segunda siempre debe empezar con mayúscula.

Ejemplo: `<type name="sc: galeriaTematica"></type>`

3.3 Pantallas Principales de la Aplicación

El núcleo de Ecumene Pyxel v2.0 cuenta con la interfaz de Alfresco la cual permite acceder a los contenidos de forma rápida. A continuación se muestran imágenes de la aplicación.

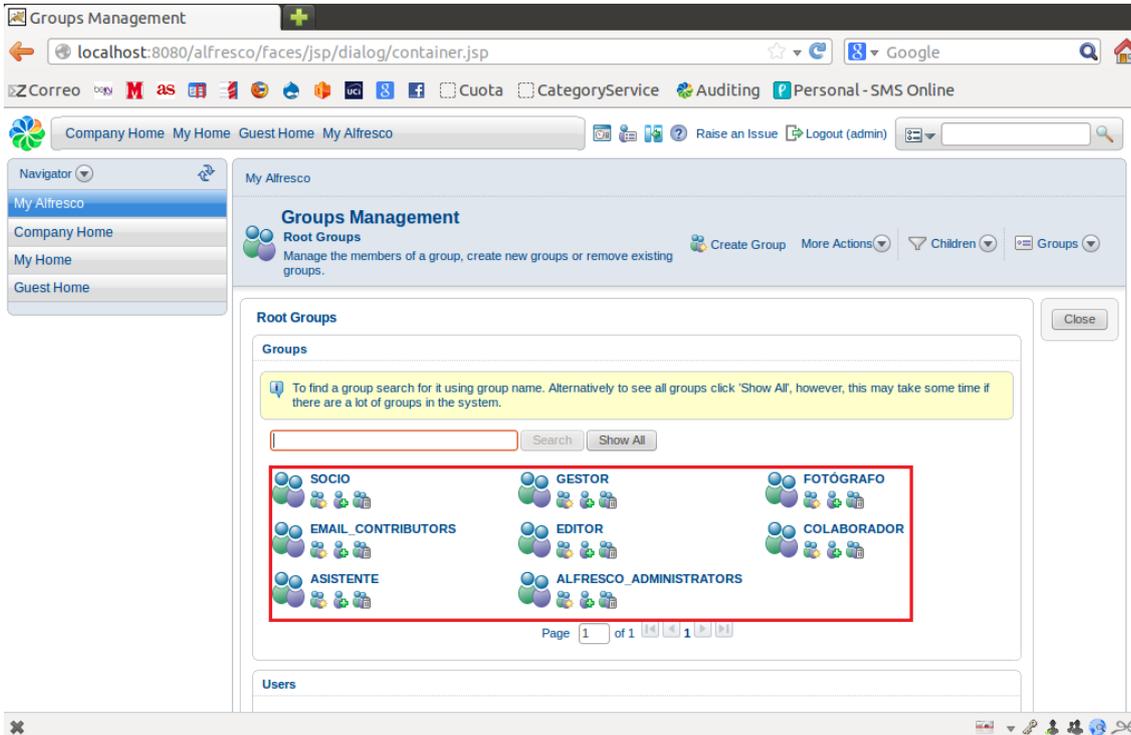
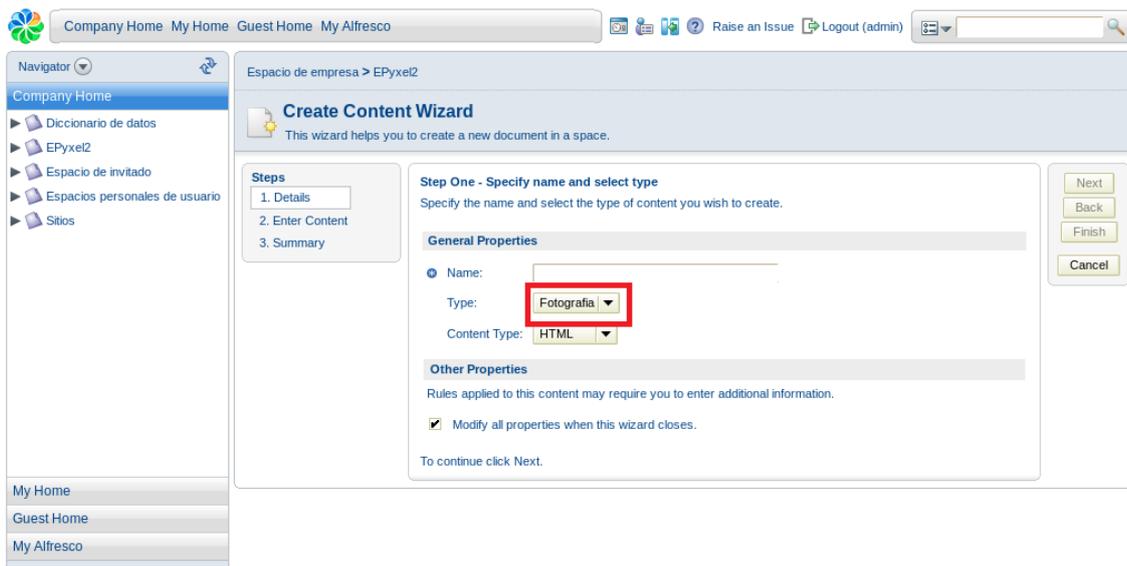


Figura 11 “Grupos de usuario del Sistema”

En la figura 11 se muestran todos los grupos de usuarios que se encuentran en el sistema.



Supplied free of charge with no support, no certification, no maintenance, no warranty and no indemnity by Alfresco or its Certified Partners. Click here for support. Alfresco Software Inc. © 2005-2010 All rights reserved.

Figura 12 “Tipo de Contenido: Fotografía”

La figura 12 muestra una imagen de la aplicación donde se selecciona el tipo de contenido fotografía.

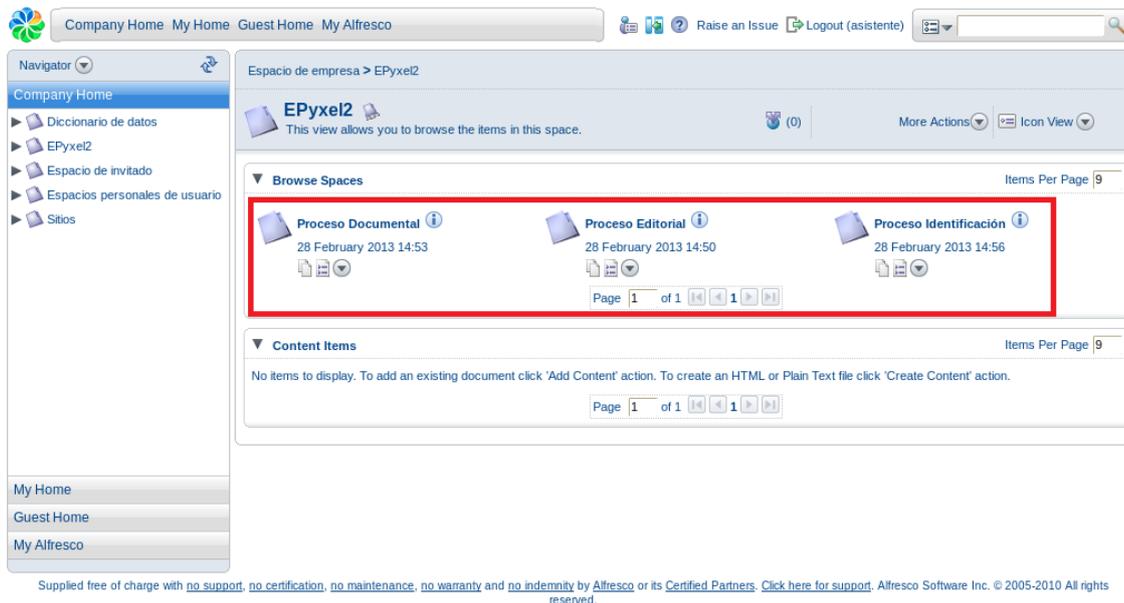


Figura 13 “Espacios Relacionados con los Procesos”

La figura 13 representa una imagen del sistema, donde se muestran los espacios relacionados con los diferentes procesos.

3.4 Validación del Sistema

Las pruebas de software se desarrollan como un conjunto de actividades dentro del proceso de desarrollo, en el cual un sistema o uno de sus componentes se ejecutan en circunstancias previamente especificadas, registrándose los resultados obtenidos. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representan una revisión final de las especificaciones del diseño y de la codificación. Luego de las pruebas se realiza un proceso de evaluación en el que los resultados obtenidos se comparan con los resultados esperados para localizar fallos en el software. El objetivo de las pruebas no es asegurar la ausencia de defectos en un software, sino demostrar que existen defectos en el software.

El proceso de validación del módulo de la versión 2 de Ecumene Pyxel está compuesto por 3 tipos de prueba: Funcionalidad, Integración y Seguridad. A continuación se presentan las pruebas realizadas y los resultados obtenidos luego de su ejecución.

3.4.1 Pruebas de Funcionalidad

Las pruebas de funcionalidad permiten confirmar que los casos de pruebas basados en casos de uso y requisitos han sido implementados correctamente al final de cada iteración. El objetivo final es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable. Por el cual existen dos técnicas para probar cualquier producto construido:

1. Técnicas de caja blanca o estructural, que se basan en un minucioso examen de los detalles procedimentales del código a evaluar, por lo que es necesario conocer la lógica del programa.
2. Técnicas de caja negra o funcional, que realizan pruebas sobre la interfaz del programa a probar, entendiendo por interfaz las entradas y salidas de dicho programa. No es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar.

De ellas se describe la técnica de caja negra o funcionales para generar casos de prueba en el sistema (Natalia Juristo, 2005).

Pruebas de caja negra o funcional

También conocidas como Pruebas de Comportamiento, estas pruebas se basan en la especificación del programa o componente a ser probado para elaborar los casos de prueba. El componente se ve como una "Caja Negra" cuyo comportamiento sólo puede ser determinado estudiando sus entradas y las salidas obtenidas a partir de ellas. No obstante, como el estudio de todas las posibles entradas y salidas de un programa sería impracticable se selecciona un conjunto de ellas sobre las que se realizan las pruebas. Para seleccionar el conjunto de entradas y salidas sobre las que trabajar, hay que tener en cuenta que en todo programa existe un conjunto de entradas que causan un comportamiento erróneo en el sistema, y como consecuencia producen una serie de salidas que revelan la presencia de defectos. Entonces, dado que la prueba exhaustiva es imposible, el objetivo final es encontrar una serie de datos de entrada cuya probabilidad de pertenecer al conjunto de entradas que causan dicho comportamiento erróneo sea lo más alto posible (Natalia Juristo, 2005).

A continuación se exponen los casos de prueba de funcionalidad correspondientes al núcleo Ecumene Pyxel en su versión 2. El resto de los casos de prueba se pueden encontrar en la planilla diseño de casos de prueba basado en caso de uso la cual se encuentra en el Expediente de Proyecto del núcleo del sistema de gestión documental Ecumene Pyxel v2.0.

Capítulo 3: “Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales fcumene Pyxel v2.0”

Caso de Prueba: Autenticar usuario

Descripción General: Permite registrar en el sistema al usuario administrador					
Condiciones de ejecución: Para autenticarse en el sistema los usuarios escriben en el navegador la dirección donde se encuentra el sistema					
Escenario	Descripción	Usuario	Contraseña	Respuesta del sistema	Flujo Central
EC 1.1 Los usuarios deben autenticarse en el sistema introduciendo usuario y contraseña correctos.	Muestra la interfaz de autenticación.	admin	admin	Muestra el espacio de trabajo de dicho usuario.	El usuario llena los campos correctamente y presiona el botón Aceptar.
EC 1.2 Los datos introducidos en el campo usuario son incorrectos	El usuario llena los campos incorrectamente.	administ	admin	Muestra un mensaje de error. No se puede iniciar sesión- nombre de usuario o contraseña desconocidos	El usuario llena los campos pero introduce el dato de la usuario que no coincide con el que hay registrado en la base de datos y presiona el botón.
EC 1.3 Los datos introducidos en el campo Contraseña son incorrectos	El usuario llena los campos incorrectamente.	admin	administ	Muestra un mensaje de error. No se puede iniciar sesión- nombre de usuario o contraseña desconocidos	El usuario llena los campos pero introduce el dato de la contraseña que no coincide con el que hay registrado en la base de datos y presiona el botón.
EC 1.4 El campo Nombre de usuario se encuentra vacío.	El usuario deja el campo del usuario vacío.		admin	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña	El usuario deja el campo usuario sin llenar y presiona el botón Aceptar.
EC 1.5 El campo Contraseña está vacío.	El usuario deja el campo de la contraseña vacío.	admin		Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña	El usuario deja el campo contraseña sin llenar y presiona el botón Aceptar.

Tabla 6 “Caso de prueba Autenticar usuario”

Caso de Prueba: Adicionar usuario

Descripción General: Permite adicionar en el sistema un usuario.							
Condiciones de ejecución: El usuario debe estar autenticado. El usuario debe estar autenticado con el rol de administrador.							
Escenario	Nombre	Apellidos	Correo Electrónico	Usuario	Contraseña	Confirmar Contraseña	Respuesta del sistema
EC 1.1 Llena los campos obligatorios correctamente para crear el usuario	Ariel	Rodriguez Arias	ariel@uci.cu	Editor	Editor	Editor	El sistema responderá con un formulario

Capítulo 3: “Implementación y Prueba del Núcleo para el Sistema de Gestión Documental de Imágenes Digitales Jcumene Poxel v2.0”

EC 1.2 El campo Nombre se encuentra vacío		Rodríguez Arias	ariel@uci.cu	Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.3 El campo Apellido se encuentra vacío	Ariel		ariel@uci.cu	Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.4 El campo Correo Electrónico se encuentra vacío	Ariel	Rodríguez Arias		Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.5 El campo Nombre de usuario se encuentra vacío.	Ariel	Rodríguez Arias	ariel@uci.cu		Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.6 El campo Contraseña está vacío	Ariel	Rodríguez Arias	ariel@uci.cu	Editor		Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.7 El campo Confirmar Contraseña está vacío.	Ariel	Rodríguez Arias	ariel@uci.cu	Editor	Editor		Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña

Tabla 7“Caso de prueba Adicionar usuario”

Caso de Prueba: Editar usuario

Descripción General: Permite editar en el sistema un usuario.							
Condiciones de ejecución: El usuario debe estar autenticado. El usuario debe estar autenticado con el rol de administrador, fotógrafo o editor.							
Escenario	Nombre	Apellidos	Correo Electrónico	Usuario	Contraseña	Confirmar Contraseña	Respuesta del sistema
EC 1.1 Presiona el vínculo Perfil de Usuario y edita el campo q dese cambiar.	Lola	Pérez Fernández	lola@uci.cu	Editor	Editor	Editor	El sistema responderá con una vista de los detalles del usuario. Debe especificar nombre de usuario y contraseña
EC 1.2 El campo Nombre se encuentra vacío		Pérez Fernández	lola@uci.cu	Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.3 El campo Apellido se encuentra vacío	Lola		lola@uci.cu	Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.4 El campo Correo Electrónico se encuentra vacío	Lola	Pérez Fernández		Editor	Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.5 El campo Nombre de usuario se encuentra vacío.	Lola	Pérez Fernández	lola@uci.cu		Editor	Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña

EC 1.6 El campo Contraseña está vacío	Lola	Pérez Fernández	lola@uci.cu	Editor		Editor	Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña
EC 1.7 El campo Confirmar Contraseña está vacío.	Lola	Pérez Fernández	lola@uci.cu	Editor	Editor		Muestra un mensaje de error. Debe especificar nombre de usuario y contraseña

Tabla 8“Caso de prueba Editar usuario”

3.4.2 Pruebas de Integración

Aunque los módulos de un programa funcionen bien por separado es necesario probarlos conjuntamente: un módulo puede tener un efecto adverso o inadvertido sobre otro módulo; las subfunciones, cuando se combinan, pueden no producir la función principal deseada; la imprecisión aceptada individualmente puede crecer hasta niveles inaceptables al combinar los módulos; los datos pueden perderse o malinterpretarse entre interfaces, etc. Por lo tanto, es necesario probar el software ensamblando todos los módulos probados previamente. Éste es el objetivo de las pruebas de integración.

Las pruebas de integración abarcan todo el sistema y pretenden cubrir plenamente la especificación de requisitos del usuario, verificando que los componentes o subsistemas interactúen correctamente a través de sus interfaces (Natalia Juristo, 2005).

Escenario	Respuesta del sistema
EC 1.1 Sistema integrado correctamente.	Se despliegan todas las funcionales, grupos de usuarios y tipos de contenidos del sistema.
EC 1.2 Sistema integrado incorrectamente.	El sistema no se inicia.

Tabla 9“Caso de prueba Integración”

Al integrar el módulo desarrollado a Epyxel2 se verificó que se ejecutaran correctamente las configuraciones de los ficheros model.xml y web-client-config.xml, además se comprobó que se cargaron los nuevos modelos de contenidos en el sistema y se ejecutaron las funcionalidades previstas.

3.4.3 Resultados de las pruebas

La realización de estas pruebas permitió detectar no conformidades, las cuales fueron erradicadas a medida que se concluía cada una de las iteraciones que se definieron. Dichas pruebas, permitieron

corroborar que el módulo cumple con las especificaciones que se trazaron y que se le dio cumplimiento a cada uno de los requisitos definidos. A continuación se muestra la relación de no conformidades (detectadas y resueltas) por iteración:

No Conformidades	Primera Iteración	Segunda Iteración	Tercera Iteración
Identificadas	26	15	5
Resueltas	17	13	5
Pendientes	9	2	0

Tabla 10 “Resultados de las No Conformidades”

Como se puede observar en la tabla anterior se realizaron tres iteraciones de pruebas, en las que se encontraron 46 no conformidades, las cuales fueron solucionadas en su totalidad. También, mediante las pruebas de integración realizadas al módulo se demostró un adecuado funcionamiento del mismo, ya que en la actualidad existe una capa de servicios que facilita la integración con otros sistemas que requieren consultar las fotografías almacenadas en el sistema de gestión documental de imágenes digitales, quedando el sistema listo para su explotación.

3.5 Conclusiones del Capítulo

Después de realizado el capítulo se obtiene como resultado el diagrama de componentes, el cual generó una vista con las librerías y ficheros que integran el módulo. Una vez desarrolladas las funcionalidades del software, se procedió a la documentación y ejecución de las Pruebas Funcionales basadas en la técnica de caja negra, con el objetivo de verificar la calidad del producto, obteniéndose resultados satisfactorios. A partir de los resultados de las pruebas se comprobó que el núcleo se encuentra listo para ser integrado al Alfresco y ser usado en entornos reales.

Conclusiones

De manera general se ha llevado a cabo un proceso de desarrollo del software completo, dividido en flujos de trabajos e iteraciones, con el objetivo de lograr un producto de calidad en el tiempo establecido. Una vez completada la presente investigación, se pueden declarar las siguientes conclusiones:

- Los fundamentos teóricos de la investigación realizada evidencian la necesidad de desarrollar un núcleo utilizando una serie de herramientas que conjuntamente a sus características y ventajas ayudaron a establecer las bases de la presente investigación.
- Luego de analizar la estructura organizativa del sistema propuesto se lograron modelar los procesos que se realizan en los medios de comunicación u otras entidades, que hagan uso de la gestión documental de imágenes digitales y los roles que intervienen en cada uno de ellos; lo antes planteado permite identificar y describir los escenarios a desarrollar.
- Se desarrolló el núcleo del sistema de gestión documental de imágenes digitales a partir del módulo Ecumene Pyxel v2.0 lo que permitió una mejora en la gestión documental de los recursos fotográficos.
- Las pruebas de funcionalidad e integración aplicadas al módulo Ecumene Pyxel v2.0 permitieron identificar y corregir un total de 46 no conformidades, lo que posibilitó comprobar que el módulo cumple con las especificaciones que se trazaron.

Recomendaciones

Después de la investigación realizada y del desarrollo del módulo se recomienda:

- Implementar nuevas funcionalidades para perfeccionar el producto, tales como:
 - ✓ Editar Grupo.
 - ✓ Incluir marca de agua a los contenidos.
- Configurar el núcleo del sistema de gestión documental de imágenes digitales Ecumene Pyxel v2.0 para envío/recepción de correo electrónico.

Bibliografías Referenciadas

Badillo Goy, Juan Carlos y Pérez Martínez, Ramón . 2007. *AVILADOC. CONTROL DE DOCUMENTOS.* 2007.

Borrego, Daniel. 2009. Herramientas para Pymes.com. [En línea] 2009. [Citado el: 15 de Octubre de 2012.] <http://www.herramientasparapymes.com/que-es-ecm-gestion-de-contenido-empresarial>.

Camacho, Lisbeth. 2008. Monografias.com. [En línea] 12 de Diciembre de 2008. [Citado el: 8 de Octubre de 2012.] <http://www.monografias.com/trabajos7/arch/arch.shtml>.

Campillo Torres , Irima , y otros. 2009. Diseño y aplicación de una herramienta informática para la gestión integral de documentos electrónicos en las organizaciones empresariales. [En línea] 2009. [Citado el: 16 de Noviembre de 2012.] <http://eprints.rclis.org/bitstream/10760/15315/1/sistema%20de%20gestion%20documental%20empresarial.pdf>.

Castells, Pablo. 2012. [En línea] 6 de Marzo de 2012. [Citado el: 11 de Octubre de 2012.] <http://arantxa.ii.uam.es/~castells/docencia/poo/2-java-esp.pdf>.

Chacón, Reinier Elejalde. 2009. *Módulo para la creación de Modelos de Contenidos para Alfresco.* La Habana: s.n., 2009. (MarcadorDePosición1).

Commons, Creative. 2006. Capítulo 2. IAGP 2005/06. Metodologías de desarrollo de software. [En línea] 30 de 12 de 2006. <http://www.um.es/docencia/barzana/IAGP/Iagp2.html>.

Consultores, Testhouse. 2011-2012. Pruebas de Seguridad - Testhouse. [En línea] 2011-2012.

Díaz, Dayani Cabrera. 2006. *Propuesta de lineamientos para el tratamiento documental de las fotografías en los medios de prensa cubanos.* 2006.

Figuroa, Inalbis Rodriguez. 2012. *Módulo de Textos para el Sistema de Gestión.* 2012.

Hernández, Yulainne. *Configuración de la Metodología OpenUP.* La Habana : s.n., 2012.

Gomez, Gabriela. 2010. Slideshare. [En línea] 10 de Julio de 2010. [Citado el: 13 de Septiembre de 2012.] <http://www.slideshare.net/gabrielagomez156/concepto-de-imagen>.

Gómez, Rafael Suárez. 2012. Tesis Doctorales en Red. [En línea] 1 de Enero de 2012. [Citado el: 8 de Octubre de 2012.] <http://www.tdx.cat/handle/10803/83343>.

Gómez, Ruth Priscila Landeros. 2011. Herramientas CASE. [En línea] 2011. [Citado el: 19 de Octubre de 2012.] <http://www.monografias.com>.

Isabel, Sergio. marzo 2012. *Interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel .* marzo 2012.

Karen. 2008. Kasyles.blogspot. [En línea] 27 de Septiembre de 2008. [Citado el: 17 de Octubre de 2012.] <http://todotecnology.blogspot.com>.

- Kinderman, Hans. 2011.** PostGreSQL. [En línea] 26 de Noviembre de 2011. [Citado el: 14 de Noviembre de 2012.] <http://postgresql-ads.blogspot.com/2011/11/ampliamente-popular-ideal-para.html>.
- Lamarca, María. 2011.** Tesis doctoral. [En línea] 19 de Noviembre de 2011. [Citado el: 15 de Octubre de 2012.] <http://www.hipertexto.info/documentos/html.htm>.
- Lapunte, María Jesús Lamarca. 2011.** Hipertexto, el nuevo concepto de documento en la cultura de la imagen. [En línea] 19 de Noviembre de 2011. [Citado el: 15 de Octubre de 2012.] <http://www.hipertexto.info/documentos/xml.htm>.
- Larman, Craig. 1999.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. s.l. : Prentice Hall, primera edición, 1999. ISBN 970-17-0261-1. 507 pp.
- Leyva, Yusmila de la Cruz. 2011.** Sistema de Gestión Documental. Módulo Fototeca para las Fotografías de la Prensa. 2011.
- Ltd, Sparx Systems Pty. 2000-2007.** Sparx Systems - Tutorial UML 2 - Diagrama de Componentes. [En línea] 2000-2007.
- Maggi, Conrado. 2011.** KnowledgeTree: Software libre para gestión documental. [En línea] 7 de Abril de 2011.
- Martínez, Rafael. 2009.** PostgreSQL-es. Portal en español sobre PostgreSQL. [En línea] 2009. [Citado el: 29 de Noviembre de 2012.] http://www.postgresql-es.org/sobre_postgresql.
- Martínez, Yaumarelis Betancourt. 2011.** *Extensión del paquete IPTC/EXIF para el Sistema de Gestión Documental de Fotografías para la Prensa*. La Habana : s.n., 2011.
- Natalia Juristo, Ana M. Moreno, Sira Vegas. 2005.** *TÉCNICAS DE EVALUACIÓN DE SOFTWARE*. 2005.
- Navarrete, Toni. 2007.** [En línea] 10 de Agosto de 2007. [Citado el: 11 de Octubre de 2012.] <http://www.dtic.upf.edu/~tnavarrete/fcsig/javascript.pdf>.
- Notte, Gastón. 2009.** Tbs-telecon. [En línea] 11 de Septiembre de 2009. [Citado el: 11 de Octubre de 2012.] <http://www.tbs-telecon.es/importancia-gestion-documental>.
- Olivares, Freddy Egdamar Paez. 2009.** Diseño UML. [En línea] 22 de 5 de 2009. <http://egdamar877.blogspot.com/2009/05/expocicion.html>.
- Olivera, Angel. 2010.** Requerimientos funcionales y no funcionales. [En línea] 06 de septiembre de 2010.
- Peña, Pedro Oramas. 2011.** *Diseño arquitectónico para la búsqueda semántica en la Universidad de las Ciencias Informáticas*. La Habana : s.n., 2011.
- Pressman, Roger S. 2002.** *Ingeniería de Software, un enfoque práctico*. s.l. : McGraw-Hill Companies, 2002. 8448132149.

Prieto, Miguel Angel Godoy. 2009. Slideshare. [En línea] 25 de Octubre de 2009. [Citado el: 11 de Octubre de 2012.] <http://www.slideshare.net/NILENSE/la-importancia-de-la-gestion-documental-en-el-desempeno-profesional>.

Pyme. 2012. Tecnologia Pyme. [En línea] 8 de 2 de 2012.
<http://www.tecnologiapyme.com/software/alfresco-entreprise-4-la-nueva-version-del-gestor-de-contenidos>.

Selection, Software. 2009. Software Selection. [En línea] 2009. www.softwareseleccion.com.

Software, Alfresco. [En línea] <http://www.alfresco.com/es/>.

Rivera Calero, Álvaro Enmanuel, Rodríguez Almache, Martha Gudelia. 2012. Repositorio Digital. [En línea] 27 de Julio de 2012. [Citado el: 22 de Octubre de 2012.]
<http://dspace.ups.edu.ec/handle/123456789/2697>.

Sierra, Manuel. 2012. Aprender a programar. [En línea] 2012. [Citado el: 15 de Octubre de 2012.]
http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=492.

Society, IEEE Computer. 2000. *recommended practice for architectural description of software intensive systems*. 2000.

Telecon, Grupo. 2007. TBS - Telecon Business Solutions. [En línea] 2007. <http://www.tbs-telecon.es/que-es-gestion-documental>.

UCI. Curso 2007-2008. SOFTWARE, C. D. A. D. I. D. Flujo de Trabajo de requerimientos. En Tema No 2. Conferencia 4. Flujo de Trabajo de requerimientos. Ciudad de La Habana. Cuba : s.n., Curso 2007-2008. p. 15..

Valdés, Damián Pérez. 2007. Maestrosdelweb. [En línea] 10 de Julio de 2007. [Citado el: 11 de Octubre de 2012.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.

Visbal, Sara M. Artilles. 2009. La gestión documental, de información y el conocimiento en la empresa. El caso de Cuba. [En línea] [Citado el: 15 de Noviembre de 2012.]
http://bvs.sld.cu/revistas/aci/vol19_5_09/aci02509.htm.

Weitzenfeld, Alfredo. 2005. Ingenieria de Software Orientada a Objetos Con Uml, Java E Internet. [En línea] 2005. [Citado el: 19 de Octubre de 2012.] <http://books.google.com>.

Bibliografías Consultada

Alfresco Admin. 2011. Arquitectura de Alfresco [En Línea] [Citado el: 4 de febrero del 2013.]
<http://alfrescoadmin.blogspot.com/2011/05/arquitectura-de-alfresco-i.html>.

Álvarez, Miguel Angel. 2001. desarrolladoresweb.com. ¿Qué es XML? [En línea] 13 de junio de 2001.
<http://www.desarrolloweb.com/>.

Álvarez Marañón, Gonzalo. 1997-1999. ¿Qué es Java? [En línea] 1997-1999.
<http://www.iec.csic.es/criptonomicon/java/quesjava.html>.

Cabrera Díaz, Dayani. 2006. Propuesta de lineamientos para el tratamiento documental de las fotografías de prensa en los medios cubanos. La Habana.

Cerda, Felipe. 2009. NetBeans, el único IDE que necesitas. [En línea] 25 de Mayo de 2009. [Citado el: 22 de Octubre de 2012.]

Ferré Grau, Xavier. 2004. Desarrollo Orientado a Objetos con UML. [Online] 2004.
<http://www.clikear.com/manuales/uml/index.aspx>.

Ferriol Marchena, Martha Marina, y otros. 2008. Manual de procedimientos para el tratamiento documental. La Habana. Archivo Nacional: s.n.

González Ríos, Osvanys y Salomón Rodríguez., Jorge Ariel. 2009. Implementación de la Interfaz Web para el Gestor de Contenido Empresarial Alfresco. Ciudad de La Habana: s.n., 2009.

Mena Mugica, Mayra. 2005. Gestión Documental y organización de archivos. La Habana, Editorial Félix Varela, 2005.

Moreno-Torres, Fernando. Gestión documental para gente normal. [En línea] [Citado el: 1 de diciembre de 2012.] <http://gestiondocumentalparagentenormal.com/>.

Pecos, Daniel. 2011. PostGreSQL y MySQL. [En línea]
http://danielpecos.com/docs/mysql_postgres/x108.html.

Pérez, Tereza Garzon. 2010. Sistemas Gestores de Bases de Datos. 2010.

Visual Paradigm Internacional. 2005. Visual Paradigm [En línea] 13 de diciembre de 2010.
<http://www.visual-paradigm.com/product/vpuml/>.

Xhidnoda. 2011. Supremacia Linux. [En línea] 22 de Abril de 2011. [Citado el: 22 de Octubre de 2012.]
<http://www.supremacialinux.com/las-nuevas-caracteristicas-de-netbeans-7-0/>.

Glosario de Términos

A

Alfresco: Gestor de Contenido Empresarial de código abierto.

API: *Application Programming Interface*, no es más que una serie de servicios o funciones que el Sistema Operativo ofrece al programador, como por ejemplo, imprimir un carácter en pantalla, leer el teclado, escribir en un fichero de disco, etc.

Archivo: Espacio que se reserva en el dispositivo de memoria de un computador para almacenar porciones de información que tienen la misma estructura y que pueden manejarse mediante una instrucción única.

C

CASE: *Computer Aided Software Engineering* (Ingeniería de Software Asistida por Ordenador), aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

CPU: Central Processing Unit (unidad de proceso central), es el cerebro del ordenador y donde se producen la mayoría de los cálculos. En términos de potencia del ordenador, la CPU es el elemento más importante de un sistema informático.

CIFS: Common Internet File System (Sistema de Archivo Común de Internet)), Un protocolo de red que proporciona la fundación para el uso compartido de archivos basado en Windows y otras utilidades de red.

E

ECM: *Enterprise Content Manager* (Manejo de Contenido empresarial). Plataforma que incluye un repositorio de contenidos a escala empresarial.

Ecumene Pyxel: Lugar donde se almacenan cada una de las imágenes digitales.

F

Framework: Plataformas o herramientas del mundo de la informática que le proveen a los programadores un grupo de facilidades en el ámbito para la cual han sido creadas.

H

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

HTML: HyperText Markup Language (lenguaje de marcas hipertextuales), diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.

HTTP: Hypertext Transfer Protocol (protocolo de transferencia de hipertexto), protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.

Hardware: Dispositivos que están conectados físicamente al ordenador.

I

IDE: *Integrate Development Enviroment* (Entorno de desarrollo integrado), herramienta que se usa para facilitar el desarrollo del software.

ISO: Organización Internacional de Estándares.

J

Java: Lenguaje de programación orientado a objetos.

JavaScript: Es un lenguaje interpretado orientado a las páginas web, para realizar tareas y operaciones en el marco de la aplicación cliente.

L

LDAP: *Lightweight Directory Access Protocol* es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

M

Modelo: Arquetipo o punto de referencia para imitarlo o reproducirlo.

P

PDF: *Portable Document Format*, (formato de documento portátil) es un formato de almacenamiento de documentos.

Proceso: Secuencia de actividades que tienen un marcado inicio y fin.

R

Requisitos: Capacidades, condiciones o cualidades que el sistema debe cumplir y tener.

S

SAMBA: Es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado SMB, renombrado recientemente a CIFS) para sistemas de tipo UNIX. Es capaz de servir colas de impresión, directorios compartidos y autenticar con su propio archivo de usuarios.

SDK: Development Kit, es un conjunto de herramientas de desarrollo que le permite a un programador crear aplicaciones para un sistema concreto, por ejemplo ciertos paquetes de software, plataformas de hardware, ordenadores.

SGBD: Sistema Gestor de Bases de Datos.

Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

T

TCP/IP: Conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre redes de computadoras.

TIC: Tecnologías de la información y las comunicaciones.

U

UML: Unified Modeling Language, lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software.

UCI: Universidad de las Ciencias Informáticas.

W

WebDAV: Es un estándar que describe cómo, a través de la extensión del protocolo HTTP 1.1, pueden realizarse acciones de gestión de archivos tales como escribir, copiar, eliminar o modificar.

WebScripts: Proporcionan una API de servicios en Alfresco, exponiendo el repositorio para la gestión de documentos y contenidos web, y proporcionando medios de búsqueda personalizados, favoreciendo la inter-operabilidad de clientes externos con el gestor documental.

Word: es un software destinado al procesamiento de textos.

X

XML: Extensible Markup Language (lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas, una manera de definir lenguajes para diferentes necesidades.

Anexos

Anexo #1: Entrevista a especialistas del Centro de Ideo Informática (CIDI)

La presente entrevista persigue realizar el levantamiento de requisitos y entender mejor cuáles son los principales problemas de los medios de comunicación con las imágenes digitales. Así se logrará entender con más claridad el negocio para luego poder realizar los flujos de trabajo y demás factores que van a intervenir en el sistema.

Entrevistado: _____

Cargo: _____

Fecha _____

Descripción de Procesos y Roles

¿Qué modificaciones se contemplan en el ciclo de vida actual de la fotografía?

¿Cuáles son los roles que intervienen en cada proceso?

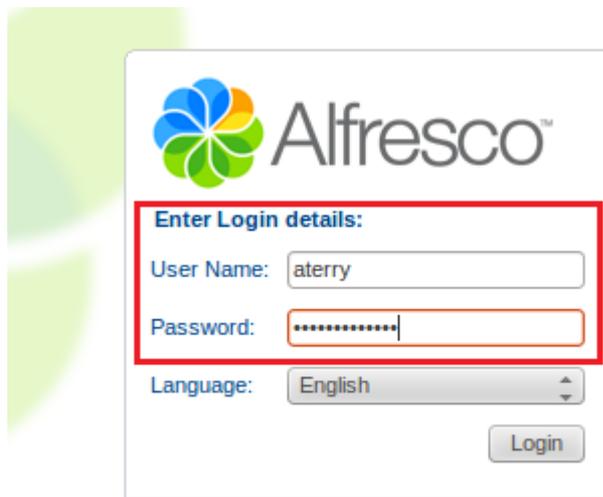
¿Qué perfiles se generarán según el proceso?

¿Qué procesos presentarán cambios en su ciclo de vida?

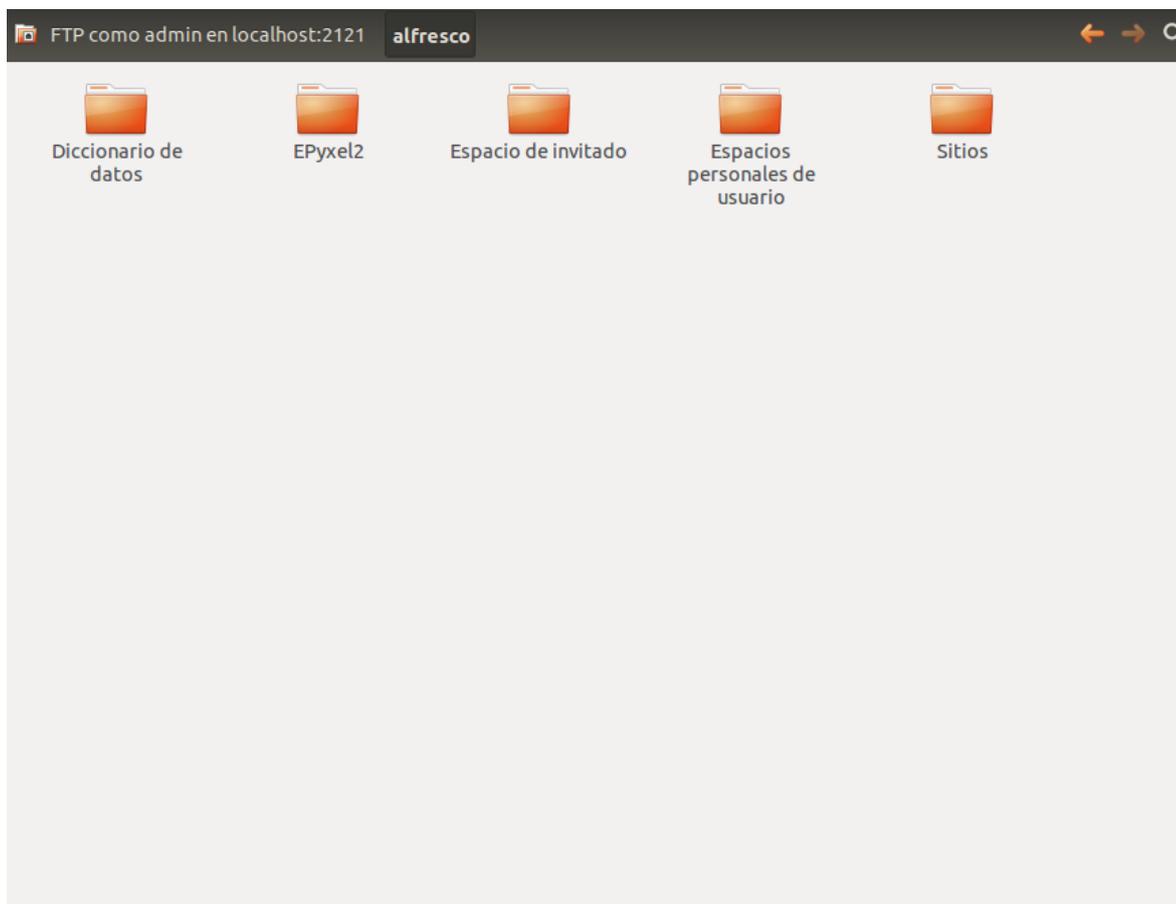
Roles y funcionalidades en cada proceso.

- ¿Es necesario que se agreguen otras funcionalidades a ese rol en específico o qué se le remueva alguna?
- Función que desempeña cada rol en el ciclo de vida, según las modificaciones.
- ¿Si se van a realizar búsquedas, cómo quieren realizarse?

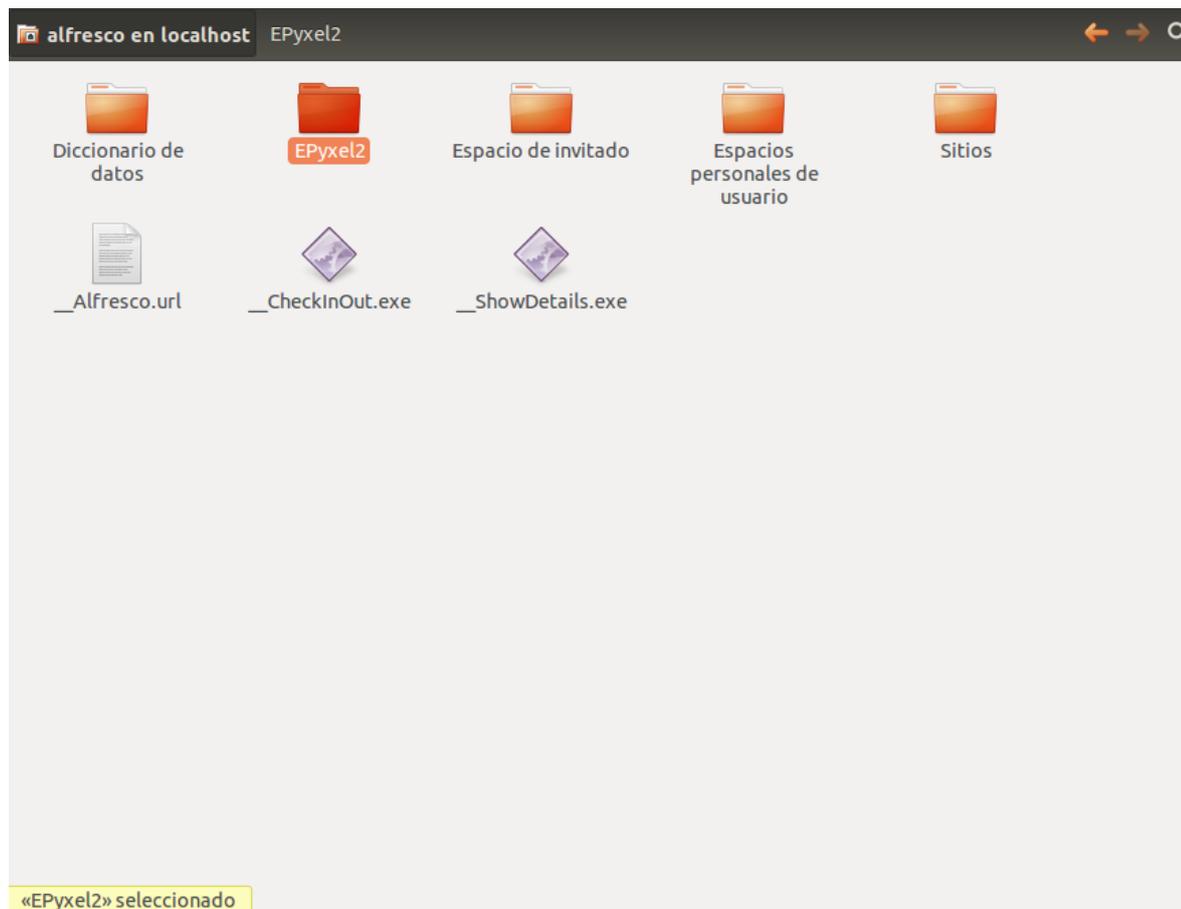
Anexo #2: Autenticando usuario por LDAP



Anexo #3: Acceso a Alfresco por FTP



Anexo #4: Acceso a Alfresco por Samba



Anexo #5: Acceso a Alfresco por WebDAV.

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

Repositorio de contenidos de ...

localhost:8080/alfresco/webdav/

Google

Correo as UC3 f Cuota CategoryService Auditing Personal - SMS Online

Listado de directorio para /

Nombre	Tamaño	Tipo	Fecha de modificación
Diccionario de datos			Mon, 04 Mar 2013 19:31:38 GMT
Espacio de invitado			Mon, 04 Mar 2013 19:31:31 GMT
Espacios personales de usuario			Thu, 14 Mar 2013 00:56:35 GMT
Sitios			Mon, 04 Mar 2013 19:31:57 GMT
EPyxel2			Thu, 14 Mar 2013 01:00:43 GMT

13

Anexo #6: Imagen del código definiendo los grupos

```
<!--Creando Grupos-->
<view:reference view:pathref="${system.authorities_container.childname}">
  <sys:children>
    <cm:authorityContainer view:childName="cm:GROUP_ASISTENTE">
      <view:aspects>
        <sys:referenceable />
      </view:aspects>
      <view:properties>
        <sys:node-uuid>GROUP_ASISTENTE</sys:node-uuid>
        <cm:name>GROUP_ASISTENTE</cm:name>
        <cm:authorityName>GROUP_ASISTENTE</cm:authorityName>
      </view:properties>
      <view:associations>
        <cm:member>
          </cm:member>
        </view:associations>
    </cm:authorityContainer>

    <cm:authorityContainer view:childName="cm:GROUP_GESTOR">
      <view:aspects>
        <sys:referenceable />
      </view:aspects>
      <view:properties>
        <sys:node-uuid>GROUP_GESTOR</sys:node-uuid>
        <cm:name>GROUP_GESTOR</cm:name>
        <cm:authorityName>GROUP_GESTOR</cm:authorityName>
      </view:properties>
      <view:associations>
        <cm:member>
          </cm:member>
        </view:associations>
    </cm:authorityContainer>
```