

Universidad de las Ciencias Informáticas

Facultad 1



Interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales: Ecumene Pyxel

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

Autores: Isabel María García Hilarión
Sergio Antonio Segura Fernández

Tutores: Ing. Serguey González Garay
Ing. Yasel Almenares Alfonso

La Habana, junio de 2012

Declaración de autoría

**MINISTERIO DE EDUCACIÓN SUPERIOR
UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

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

Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2012.

Autora: Isabel María García Hilarión

Autor: Sergio Antonio Segura Fernández

Tutor: Ing. Serguey González Garay

Cotutor: Ing. Yasel Almenares Alfonso

Dedicatoria

Isabel

A mi mamita por brindarme la dicha de existir

A mi papi por creer en mí

Sergio

A mi mamita por saber comprenderme y por regalarme su vida incondicionalmente

A mi papá por haberse convertido en mi guía

Agradecimientos

Agradecimientos Isabel

A mis padres Isabel y Nico por ser siempre mi luz, por entregarme todo su amor, por mimarme, por consentirme, por su confianza y por regalarme la dicha de tenerlos siempre junto a mí.

A mis suegritos Sergio y Pilar por su derroche de cariño y comprensión.

Al amor de mi vida por regalarme diariamente su sonrisa que es lo que necesito para ser feliz.

A mis hermanitos Eli y Kike por dejarme ser su ejemplo.

A mis abuelitos Nico, Cuca y Papi por regalarme su ternura.

Y especialmente a mi abuelita Illa por entregarme su vida.

Agradecimientos Sergio

A mis padres Pilar y Sergio por representar guías en mi vida, por enseñarme a ser cada vez mejor, por el apoyo que han sabido darme en todo momento sin importar situación alguna, porque elegirlos mejores sería imposible, y porque se merecen mi amor, mi cariño y mi respeto por encima de cualquier cosa.

A mis suegros queridos Isabel y Nico por la confianza que han tenido en mí.

A la personita que llegó para quedarse en mi vida ocupando un lugar muy importante, mi corazón.

A Manito por permitirme ser su ejemplo a seguir.

A mis abuelos: Lesbia, Toñita, Isidoro, Toñito, por lograr una familia tan unida y por sus ejemplos.

Agradecimientos comunes

A Yanet y Soyma por ofrecernos su amistad incondicional.

A nuestros compañeros Jenlys, Elier, Mayde, Eddy y Jennifer por permitirnos compartir juntos una época maravillosa de nuestras vidas.

A nuestra familia por apoyarnos y querernos.

A nuestros tutores Serguey y Yasel por brindarnos su ayuda en los momentos más difíciles.

A nuestros profes Gilberto y Elieyis por su paciencia y apoyo incondicional.

A todos los que de una forma u otra pusieron su granito de arena durante todo el desarrollo de este trabajo.

Resumen

En la actualidad el tratamiento de las imágenes presenta dificultades empleándose rudimentarios sistemas organizativos que impiden un uso óptimo de estos recursos. Debido a que el sistema Ecumene Pyxel desarrollado en la Universidad de las Ciencias Informáticas, para la gestión de imágenes digitales, presenta algunas desventajas que impiden un óptimo desempeño del proceso de gestión documental de las mismas, se desarrolló una interfaz Web amigable al usuario que permita la interactividad con la aplicación Ecumene Pyxel. Para guiar el proceso de análisis, diseño, implementación y prueba de este sistema, se utilizó la metodología ágil de desarrollo SXP, como tecnologías se pueden mencionar el marco de trabajo Symfony 2, la librería jQuery, el Visual Paradigm como herramienta para el modelado y el sistema gestor de base de datos PostgreSQL. Con el sistema obtenido se mejora y optimiza la gestión de las imágenes digitales durante todo su ciclo de vida.

Palabras clave: gestión documental, imagen, interfaz Web, ECM Alfresco.

Índice de contenido

Introducción	1
1. Capítulo 1: Fundamentos teóricos que sustentan la propuesta de solución de la interfaz Web Ecumene Pyxel	6
1.1. Interfaz Web	6
1.1.1. Accesibilidad Web.....	7
1.2. Gestión documental.....	8
1.3. Imagen.....	9
1.4. Sistemas de gestión documental.....	10
1.4.1. ECM Alfresco	10
1.5. Interfaces Web desarrolladas para el ECM Alfresco	11
1.5.1. Alfresco Explorer	11
1.5.2. Opsoro	12
1.5.3. ifresco	13
1.5.4. DoCASU	14
1.5.5. ExtAlf	15
1.5.6. eXcriba.....	15
1.6. Tecnologías asociadas al desarrollo del sistema	17
1.6.1. Marcos de trabajo que soportarán el desarrollo	17
1.6.2. Entornos integrados de desarrollo (IDE).....	18
1.6.3. Los Sistemas de Gestión de Base de Datos (SGBD).....	19
1.6.4. Lenguajes para la Web	20
1.6.5. Ingeniería del Software asistida por computadoras (CASE).....	22
1.6.6. Metodologías de desarrollo.....	22
2. Capítulo 2: Propuesta de solución para la interfaz Web Ecumene Pyxel.....	24
2.1. Propuesta del sistema	24
2.2. Captura de los requisitos.....	27

2.2.1.	Lista de Reserva del Producto.....	27
2.3.	Historias de usuario.....	31
2.3.1.	Autenticar usuario.....	32
2.3.2.	Adicionar imagen.....	33
2.3.3.	Eliminar imagen.....	35
2.4.	Propuesta de arquitectura.....	36
2.4.1.	Patrones de arquitectura.....	38
2.4.2.	Patrones de diseño.....	38
2.5.	Diseño de la interfaz Web Ecumene Pyxel.....	40
2.5.1.	Diagrama de paquetes del sistema.....	40
2.5.2.	Diagrama de clases del diseño autenticar usuario.....	41
2.6.	Diseño de la base de datos.....	47
2.6.1.	Modelo Entidad-Relación de la base de datos.....	47
2.6.2.	Descripción de las tablas.....	48
3.	Capítulo 3: Construcción de la propuesta de solución a la interfaz Web Ecumene Pyxel y validación de los resultados.....	51
3.1.	Diagrama de componentes.....	51
3.1.1.	Representación gráfica del diagrama de componentes.....	52
3.1.2.	Descripción del diagrama de componentes.....	53
3.2.	Diagrama de despliegue.....	53
3.2.1.	Representación gráfica del diagrama de despliegue.....	54
3.2.2.	Descripción del diagrama de despliegue.....	54
3.3.	Diseño y ejecución de las pruebas de software.....	55
3.3.1.	Pruebas de funcionalidad.....	55
3.3.1.	Resultados de la pruebas funcionales.....	59
4.	Conclusiones.....	62
5.	Recomendaciones.....	63
6.	Bibliografía.....	64

7.	Anexos.....	68
7.1.	Anexo 1. Diagramas de procesos	68
7.2.	Anexo 2. Lista de riesgos	73
7.3.	Anexo 3. Prototipos no funcionales	75
7.4.	Anexo 4. Diagramas de clases.....	77
7.5.	Anexo 5. Diagramas de componentes	78
8.	Glosario de términos.....	80

Introducción

A medida que transcurre el tiempo es cada vez más imperiosa la necesidad de organizar la información, pues se multiplica vertiginosamente su volumen, y por lo que su contenido representa, esta se ha convertido en un recurso especial. Por tanto, se hace fundamental contar con una gestión adecuada, pues a mayor crecimiento de la información, mayor es el caos que se genera.

En Cuba, varias organizaciones que tienen a su cargo la manipulación de grandes volúmenes de archivos, poseen esta información en papel, dispersa en computadoras personales o en la red. En su mayoría fuera del margen de las políticas y las normas corporativas de seguridad, sin procedimientos de preservación y fuera del alcance de las herramientas de colaboración y consulta. Específicamente, la consulta, preservación y manipulación de las imágenes constituyen un problema, debido a que estos recursos se encuentran, mayormente, fuera de todo proceso de gestión.

Las imágenes desempeñan un papel importante, constituyen un material permanentemente utilizado por los profesionales del sector periodístico, artístico, investigativo y otras instituciones. Este recurso crece diariamente como parte de la rutina de producción de cada medio, por lo que sus fondos fotográficos se encuentran en constante aumento. Sin embargo, las unidades de información de la mayoría de los medios, emplean rudimentarios sistemas organizativos que impiden un uso óptimo de estos recursos y no se utilizan métodos probados que sistematicen su tratamiento durante todo el ciclo de vida (GEITEL, 2011).

Al considerar las dificultades mencionadas, surge la necesidad de crear un sistema integrado, ligero y manejable por los usuarios que permita la gestión y acceso a las imágenes digitales disponibles, de manera tal que garanticen la custodia del patrimonio icónico de cada medio y el acceso oportuno a todos los recursos de las entidades participantes. Como parte del programa de información para la prensa cubana desde la Universidad de las Ciencias Informáticas se ha impulsado el estudio y desarrollo de un sistema de gestión documental de imágenes digitales basado en las tecnologías que provee el Gestor de

Contenidos Empresariales (ECM por sus siglas en inglés) Alfresco¹.

Este sistema presenta importantes ventajas en cuanto a la organización y documentación de las imágenes, pero su interfaz Web presenta los siguientes inconvenientes:

- ✓ Tiene una curva de aprendizaje muy elevada.
- ✓ La interfaz Web es poco intuitiva al usuario.
- ✓ El usuario necesita realizar muchos procesos que pueden ser automatizados.
- ✓ El usuario debe poseer conocimientos informáticos.

Estos inconvenientes traen como consecuencia:

- ✓ Desmotivación por parte del usuario para trabajar con el sistema.
- ✓ Se hace más lenta la gestión y recuperación de las imágenes digitales.
- ✓ Se dificulta la búsqueda y recuperación de la información.

A partir de lo expresado anteriormente se identifica como **problema de la investigación**: ¿Cómo mejorar la gestión de las imágenes digitales en el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel?

El **objeto de estudio** se centra en el desarrollo de las interfaces Web.

Como **objetivo general** se plantea: Desarrollar una interfaz Web para la gestión de las imágenes digitales mediante el empleo de la capa de servicios Web que brinda el núcleo del Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.

Objetivos específicos

- ✓ Analizar las tendencias actuales de las interfaces Web para los Sistemas de Gestión Documental

¹<http://www.alfresco.com/es>

en Cuba y el mundo.

- ✓ Diseñar las funcionalidades de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Implementar las funcionalidades de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Validar las funcionalidades de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.

El **campo de acción** se basa en el desarrollo de interfaces Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.

Tareas de investigación

- ✓ Caracterización de las tendencias actuales de diseño e implementación de las interfaces Web para Sistemas de Gestión Documental.
- ✓ Identificación de los requerimientos funcionales y no funcionales de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Selección de las tecnologías, herramientas, estándares y metodologías necesarias para el desarrollo de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Confección del diseño de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Implementación de las funcionalidades de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.
- ✓ Aplicación de las pruebas de funcionalidad a la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.

Métodos científicos

Los métodos científicos a utilizar durante el desarrollo de la presente investigación, que permitirán

contribuir al cumplimiento de los objetivos propuestos son los siguientes:

Métodos Teóricos

El **Analítico-Sintético**: Se utiliza para hacer un análisis bibliográfico del tema relacionado con los sistemas de gestión documental, con el objetivo de realizar una síntesis de los aspectos necesarios para llevar a cabo el desarrollo de la interfaz Web.

Histórico-Lógico: Mediante esta práctica se analizaron las tendencias actuales en el ámbito nacional e internacional de las diferentes interfaces Web para el Sistema de Gestión Documental ECM Alfresco.

Métodos Empíricos

La Entrevista: Usada en la recopilación de información para el levantamiento de requisitos. Además para el estudio y comprensión de la comunicación entre la interfaz Web con el ECM Alfresco mediante los servicios Web.

Observación estructurada: Validar, mediante los casos de pruebas, que se cumplan los requisitos funcionales del sistema como pueden ser la correcta gestión de las imágenes en el producto.

Estructuración del contenido

El presente trabajo está estructurado en tres capítulos:

Capítulo 1 Fundamentos teóricos que sustentan la propuesta de solución de la interfaz Web

Ecumene Pyxel: En este capítulo se describen los principales conceptos utilizados en el trabajo. Además son caracterizadas las tendencias actuales de las interfaces Web para los sistemas de gestión documental, centrados principalmente en la gestión de imágenes digitales. También se identifican las herramientas, metodologías y tecnologías a utilizar para el desarrollo del producto.

Capítulo 2 Propuesta de solución para la interfaz Web Ecumene Pyxel

Con el objetivo de obtener el producto final se realiza la descripción de la solución propuesta, mediante los diagramas de procesos. Son identificados los requisitos funcionales, a los cuales se les describen las historias de usuario

correspondientes y también se determinan los requerimientos no funcionales así como la definición de escenarios. Además se concreta el diagrama de paquetes. Por último se describe el diseño del sistema mediante la arquitectura y patrones de diseño propuestos para el mismo.

Capítulo 3 Construcción de la propuesta de solución a la interfaz Web Ecumene Pyxel y validación de los resultados: Para desarrollar la solución propuesta se realiza todo el proceso de implementación del software. También son representados los diagramas de paquetes de componentes y mediante el diagrama de despliegue se define cómo quedan distribuidos físicamente los distintos nodos que componen el sistema. Además se muestra el diseño y aplicación de los casos de prueba realizados, así como los resultados obtenidos.

1. Capítulo 1: Fundamentos teóricos que sustentan la propuesta de solución de la interfaz Web Ecumene Pyxel

Este capítulo se realiza un estudio conceptual que sustenta los principales elementos necesarios para el desarrollo de la interfaz Web Ecumene Pyxel. Además se realiza un estudio del estado del arte de las interfaces Web desarrolladas para la gestión documental en el ECM Alfresco. También se exponen las principales herramientas, lenguajes, metodologías y tecnologías que se utilizarán en el desarrollo de la interfaz Web.

1.1. Interfaz Web

Con el avance de Internet, se ha hecho necesario desarrollar nuevos sistemas que satisfagan las necesidades de los internautas, lo cual influye en la manera de representar la información, específicamente en las interfaces Web.

Según la Real Academia de la Lengua Española el término interfaz es la conexión física y funcional entre dos aparatos o sistemas independientes (Real Academia de la Lengua Española, 2011). Una interfaz Web es un sistema o conjunto de elementos gráficos que permite a los usuarios tener control sobre las funcionalidades de un sistema, gestionar sus contenidos y realizar tareas específicas sin la necesidad de haber adquirido conocimientos avanzados.

Las interfaces Web son interfaces gráficas de usuario con elementos comunes de presentación y navegación que deben servir de intermediarias entre usuarios genéricos, no acostumbrados al uso de aplicaciones informáticas, sistemas de información y procesos transaccionales, lo cual facilita la localización de la información deseada, el entendimiento claro de las funcionalidades ofrecidas y la realización práctica de tareas específicas por parte de los usuarios (Moreno, 2005).

Un sitio Web que presente una interfaz poco intuitiva, el cual limita al usuario de un rápido y cómodo acceso al contenido, aunque muestre lo más actualizado sobre el tema que trata, no obtendrá tanto éxito comparado con otro que aborde el mismo tema a pesar de que presente un contenido de menor calidad y

tenga una interfaz que permita a los usuarios navegar de forma sencilla y acceder a la información deseada.

1.1.1. Accesibilidad Web

La accesibilidad Web es la posibilidad que presenta un producto o servicio Web de ser accedido y usado por el mayor número posible de personas, indistintamente de las limitaciones propias del individuo o de las derivadas del contexto de uso (Hassan, y otros, 2003).

Un diseño será accesible cuando sea usable para más personas en más situaciones o contextos de uso (Henry, 2002), lo que posibilita a todos los usuarios, de forma eficiente y satisfactoria, la realización y consecución de tareas (Nielsen, 2001).

La accesibilidad debe ser entendida como parte de la usabilidad.

1.1.2. Usabilidad Web

La usabilidad (dentro del campo del desarrollo Web) es la disciplina que estudia la forma de diseñar sitios Web para que los usuarios puedan interactuar con ellos de la forma más fácil, cómoda e intuitiva posible (Hassan, 2002).

Numerosos autores han propuesto diversas definiciones de usabilidad, normalmente a través de la enumeración de los diferentes atributos o factores mediante los que puede ser evaluada, cada definición depende del enfoque con el que pretende ser medida (Folmer, y otros, 2004).

Para este trabajo se utiliza la definición ofrecida por la ISO y que define usabilidad como el “grado de eficacia, eficiencia y satisfacción con la que usuarios específicos pueden lograr objetivos específicos, en contextos de uso específico” (ISO 9241-11, 1998).

Actualmente la mayoría de los sitios Web están centrados en los usuarios, para facilitarles la adquisición de la información deseada, mediante el uso de imágenes, multimedia y otros materiales de apoyo.

1.2. Gestión documental

La gestión documental engloba un conjunto de operaciones comprometidas con la búsqueda de la economía y la eficacia en la producción, el mantenimiento, uso y destino final de los documentos a lo largo de su ciclo de vida; es decir, desde el momento de su concepción en las oficinas administrativas hasta su ingreso en las instituciones de archivo (Fugueras, 2003).

Para Ponjuan Dante, la gestión documental es un proceso administrativo que permite analizar y controlar sistemáticamente, a lo largo de su ciclo de vida, la información registrada que crea, recibe, mantiene o utiliza la organización en correspondencia con su misión, objetivos y operaciones. También la considera como un proceso para mantener la información en un formato que permita su acceso oportuno, y por ello se requiere de la realización de tareas y procedimientos particulares para cada fase de su ciclo de vida y su explotación. La información registrada es evidencia de las actividades y transacciones de las organizaciones, y su uso oportuno permite a la organización una mayor eficacia en su labor (Ponjuan, 2005).

Para Mena Mugica, uno de los aspectos más importantes de la gestión documental desde el enfoque de los archivos totales, es la implantación de sistemas integrales de tratamiento de la documentación que se ocupen de su producción, conservación, uso, explotación y eliminación, según las necesidades de cada tipo de usuario, sean los propios productores o investigadores (Mugica, 2005).

La gestión documental eficiente implica suministrar el documento adecuado a la persona apropiada, en el momento oportuno con el menor costo posible. La correcta gestión documental permite a la institución realizar operaciones administrativas eficientes, tomar decisiones, crear, mantener, usar, guardar y hacer una evaluación de los documentos de manera eficaz (GRANMA, 2006).

Los autores de este trabajo entienden que gestión documental es el proceso o conjunto de operaciones realizadas sistemáticamente sobre los documentos de una organización durante su ciclo de vida, con el objetivo de preservarlos y tenerlos disponibles en el momento y en el formato deseado, para un correcto desempeño y eficacia en el desarrollo de las tareas de una institución.

1.3. Imagen

La gestión documental abarca todo tipo de archivos, entre los que se encuentran las imágenes. Una imagen es la representación visual de un objeto, una persona, un animal o cualquier otra cosa plausible de ser captada por el ojo humano a través de diferentes técnicas como puede ser la pintura, el diseño, la fotografía y el video (Definición ABC, 2011).

Los autores de esta investigación luego de realizar una revisión de la temática definen imagen como la figura, representación, semejanza o apariencia de algo. Una imagen también es la representación visual de un objeto a través de técnicas de la fotografía, la pintura, el diseño, el vídeo u otras disciplinas. La imagen puede ser almacenada en formato digital, con el objetivo de preservarlas a lo largo del tiempo.

Las imágenes digitales son fotos electrónicas tomadas de una escena o escaneadas de documentos, fotografías, manuscritos, textos impresos e ilustraciones. Se realiza una muestra de la imagen digital y se confecciona un mapa de ella en forma de cuadrícula de puntos o elementos de la figura (píxeles). A cada píxel se le asigna un valor tonal (negro, blanco, matices de gris o color), el cual está representado en un código binario (ceros y unos) (Biblioteca de la Universidad de Cornell, 2003).

Las instituciones que inauguran programas que utilizan imágenes digitales deben encarar asuntos de gestión. Los mismos pueden estar caracterizados de distintos modos, pero todos ellos se reducen a establecer correlaciones entre los recursos, procesos y los objetivos del proyecto (Biblioteca de la Universidad de Cornell, 2003).

En el proceso de gestión de la imagen digital se realizan tareas como la conservación, la selección, la difusión, la descripción, el control administrativo, la clasificación y la indexación. Esto aumenta potencialmente las posibilidades de explotación, con el objetivo de alargar el tiempo de disponibilidad de los recursos digitales.

1.4. Sistemas de gestión documental

Con el acelerado avance tecnológico, en el mundo se han creado soluciones para la organización de contenidos con el uso de las tecnologías de la información. Entre estas se encuentran los llamados sistemas de gestión documental.

Sistema de gestión documental (*Document Management System*) es un sistema computarizado, un conjunto de programas, utilizado para rastrear y almacenar documentos electrónicos y/o imágenes de documentos soportados en papel. El término puede ser relacionado con conceptos como sistemas de administración de contenido y es visto como un sistema de administración de contenido corporativo y relacionado con el término digital (Marrero, 2009).

La combinación de este tipo de bibliotecas de documentos con índices almacenados en una base de datos permite el acceso rápido mediante diversos métodos a la información contenida en los documentos. Estos generalmente se encuentran comprimidos y además de texto pueden contener cualquier otro tipo de documentos multimedia como imágenes o vídeos (Marrero, 2009).

1.4.1. ECM Alfresco

Entre los sistemas de gestión documental se encuentra el Gestor de Contenido Empresarial (ECM por sus siglas en inglés) Alfresco, el cual está basado en estándares abiertos y de escala empresarial, para Windows y sistemas operativos tipo Unix (de la Fuente, 2008). Éste ofrece un gran rango de soluciones a los usuarios que necesitan la gestión de información en una empresa y una amplia cantidad de funcionalidades para la gestión documental.

“Alfresco 3 is one of the most versatile open source Enterprise Content Management (ECM) platforms... It is also very well designed... The open source adoption allows developers to contribute on the project and that's why you can find more than 240 extensions in the Alfresco Forge” (Cei, y otros, 2010).

Traducción en español de los autores: “Alfresco 3 es una de las plataformas de ECM más versátiles de código abierto...además está muy bien diseñado... Que sea de código abierto permite a los

desarrolladores contribuir al proyecto y este es el por qué se pueden encontrar más de 240 extensiones de este.

Alfresco está orientado a ofrecer una solución centrada en la simplicidad, implementación rápida para ofrecer un valor de negocio inmediato y agilizar el desarrollo de aplicaciones mediante componentes pre-integrados y *scripts* ligeros (GEITEL, 2011).

Con el uso del ECM Alfresco se puede administrar todo tipo de documento como son HTML², texto, Adobe PDF³, Flash⁴, imágenes digitales, media y vídeo. Entre las ventajas del ECM Alfresco se encuentra que brinda una capa de comunicación con servicios que pueden variar según las necesidades del usuario.

El acceso al ECM Alfresco puede realizarse mediante diferentes protocolos y permite la interoperabilidad con otros sistemas.

1.5. Interfaces Web desarrolladas para el ECM Alfresco

1.5.1. Alfresco Explorer

Alfresco Explorer es la clásica aplicación Web del ECM Alfresco. Es su principal aplicación para la gestión de documentos. Ofrece algunas funcionalidades de administración del sistema. Entre sus principales características se encuentran:

- ✓ Es código abierto, liberado bajo la licencia LGPL⁵.
- ✓ Consume los servicios directamente del repositorio del ECM Alfresco (ver Figura 1).
- ✓ Está desarrollado en Java⁶.

² Lenguaje de marcado de hipertexto.

³ Formato de documento portátil.

⁴ <http://www.adobe.com/es/products/flash.html>

⁵ <http://www.gnu.org/copyleft/lgpl.html>

⁶ Lenguaje de programación Orientado a Objetos.

Algunas de sus limitaciones son las siguientes:

- ✓ Está diseñada para usuarios con conocimientos avanzados en las tecnologías informáticas.
- ✓ No ofrece los servicios necesarios para una correcta gestión de la imagen en todo su ciclo de vida.
- ✓ Es muy complicada su configuración.
- ✓ Escasa información de consulta disponible.

1.5.2. Opsoro

Opsoro es un cliente Web alternativo para la Plataforma del ECM Alfresco. Su objetivo es proporcionar una interfaz de usuario más fácil con una experiencia de usuario más rica, mediante el aprovechamiento de las últimas tecnologías Web. Al igual que el ECM Alfresco, Opsoro es un software de código abierto publicado bajo la licencia GPL⁷, está desarrollado en la librería ExtJS para AJAX y utiliza la tecnología Web Script (ver Figura 1). Su nombre se debe a la abreviatura de “*Open Source Rocks*” cuya traducción al español es “Rocas de código abierto” (Opsoro, 2010) (Alfresco, 2011).

Características actuales:

- ✓ Es amigable e intuitivo para el usuario.
- ✓ Permite la búsqueda a medida que se escribe.
- ✓ Contiene categorías y etiquetas de carpetas.
- ✓ Contiene etiquetado de documentos.
- ✓ Permite la vista previa de las imágenes.
- ✓ Permite la visualización en línea de textos, PDF e imágenes.
- ✓ Opsoro utiliza principalmente JSON⁸ como formato de respuesta para mantener los fragmentos de datos tan pequeños como sea posible.
- ✓ Proporciona una interfaz de usuario sencilla.

⁷ Licencia Pública General.

⁸ Formato ligero para el intercambio de datos.

Esta interfaz Web no presenta las funcionalidades requeridas para realizar una gestión de las imágenes digitales durante todo su ciclo de vida.

1.5.3. ifresco

El ifresco es un cliente Web de código abierto desarrollado para el ECM Alfresco, puede ser manejado con facilidad. Este sistema resulta muy sencillo de trabajar, pues posee funcionalidades como menús contextuales, listas ordenadas y permite arrastrar y soltar elementos (iFresco, 2011).

Arquitectura y características de ifresco:

- ✓ Se implementó en PHP5.
- ✓ Está desarrollado en Symfony 1.4, que es una versión obsoleta por lo cual resulta complicada su modificación.
- ✓ Utiliza Doctrine⁹ para la base de datos ORM¹⁰.
- ✓ Como Sistema Gestor de Base de Datos utiliza MySQL para salvar datos administrativos, las sesiones de usuarios y favoritos.
- ✓ Utiliza jQuery para diversas funciones de JavaScript como son arrastrar y soltar elementos.
- ✓ Maneja los ExtJS¹¹ con plug-ins para la interfaz de usuario.
- ✓ La integración con el ECM Alfresco está basada en servicios Web API y en su RESTful¹² API (ver Figura 1), al conectarse directo a estas API¹³ no necesita la adición de nuevos Web Scripts¹⁴ al servidor.

Como limitaciones se identificaron:

⁹ <http://www.doctrine-project.org/>

¹⁰ Mapeo Objeto-Relacional.

¹¹ <http://extjs.es/>

¹² Transferencia de Estado Representacional.

¹³ Interfaz de programación de aplicaciones.

¹⁴ Archivo de procesamiento por lotes.

- ✓ No posee las funcionalidades y características necesarias para una correcta gestión de las imágenes en todo su ciclo de vida.
- ✓ No existe documentación sobre el desarrollo de la tecnología
- ✓ No cuenta con una comunidad que le dé seguimiento y corrección a las deficiencias y bug de seguridad identificados.

1.5.4. DoCASU

DoCASU es una interfaz de usuario (UI por sus siglas en inglés) para el sistema de código abierto ECM Alfresco. Fue diseñado por la empresa Optaros¹⁵ y liberado bajo licencia GPL, es muy sencillo de usar, por lo simple que es. Con DoCASU los usuarios del ECM Alfresco tienen la posibilidad de simplificar y facilitar las soluciones para acceder, agregar, buscar y modificar documentos (Fernández, 2009).

El objetivo del proyecto DoCASU es proporcionar a la Comunidad del ECM Alfresco una interfaz personalizada enfocándose principalmente en la experiencia del usuario, es decir que posea:

- ✓ Mayor facilidad de uso.
- ✓ Capacidad de respuesta.
- ✓ Rapidez.

DoCASU puede ser tomado por los desarrolladores como punto de partida para la realización de interfaces de usuario más amplias, o simplemente como la base de aprendizaje para desarrollar aplicaciones basadas en los Web Scripts (Rotz, 2008).

Esta interfaz Web se comunica con el ECM Alfresco mediante los Web Scripts externos (ver Figura 1), lo que permite una mayor rapidez de comunicación entre los dos sistemas. A pesar de todas las ventajas de esta aplicación, tiene como inconvenientes que no maneja los tipos de contenidos necesarios para el sistema Ecumene Pyxel y modificar esto implicaría cambiar la lógica del negocio del mismo, pues este le da un mismo tratamiento a todos los documentos.

¹⁵ www.optaros.com/

1.5.5. ExtAlf

ExtAlf es una interfaz Web inspirada en Opsoro. Para la facilidad de uso, utiliza ExtJS, que es una librería JavaScript ligera y de alto rendimiento, que permite crear páginas Web dinámicas. Está creada para la administración del flujo de trabajo del ECM Alfresco integrándose a este mediante los Web Scripts (ver Figura 1), lo cual permite una comunicación rápida y eficiente para realizar una buena gestión de la información. Tiene como inconvenientes que:

- ✓ No cuenta con documentación sobre el desarrollo de la tecnología.
- ✓ No tiene una comunidad de respaldo que le dé seguimiento y corrección a sus deficiencias.

1.5.6. eXcriba

El 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. Este sistema está compuesto por un núcleo o motor que es el ECM Alfresco y una aplicación Web la cual se integra mediante el API¹⁶ Servicios Web y los Web Scripts externos que brinda el repositorio del ECM Alfresco (ver Figura 1). Esta interfaz guía y ayuda al usuario a realizar una gestión de los documentos de una forma adecuada, sencilla y organizada (Elejalde, 2009) (Blanco, y otros, 2012) (Fonseca, 2012). Este sistema presenta las siguientes desventajas:

- ✓ Está desarrollado para una versión del ECM Alfresco que no es compatible con las especificaciones del sistema.
- ✓ No tiene en cuenta la especialización de los tipos de contenidos, pues le da el mismo tratamiento a todos los documentos.
- ✓ no tiene implementado el estándar IPTC para la representación de las imágenes en el sistema.
- ✓ No es flexible al cambio de las interfaces de los tipos de usuarios pues todos los usuarios tiene una misma interfaz.

¹⁶ Interfaz de programación de aplicaciones.

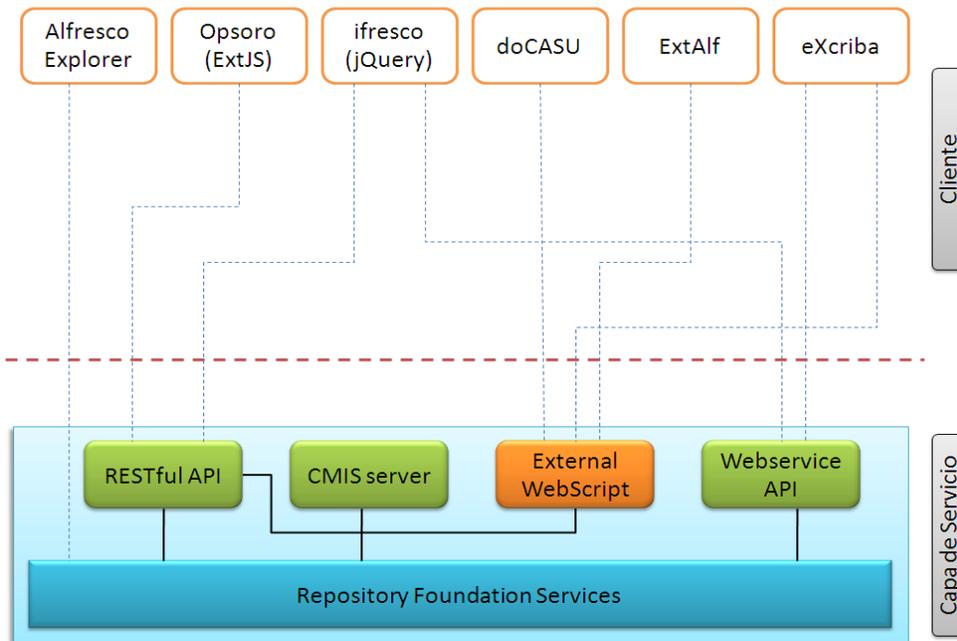


Figura 1. Comunicación de las interfaces Web con el ECM Alfresco

Luego del estudio realizado se identificó, que muchos de estos sistemas de una forma u otra dan tratamiento documental de manera general a un cúmulo considerable de contenidos, pero no cumplen con las especificaciones del proyecto en cuanto a la representación y manejo de las imágenes digitales en el ciclo de vida definido, aunque es válido resaltar que constituyen buenas prácticas en cuanto a los mecanismos de integración con el ECM Alfresco.

A partir de lo expuesto anteriormente se realiza la propuesta de implementación de una nueva interfaz Web que utilice las buenas prácticas que posee el eXcriba 1.0 para la comunicación con el ECM Alfresco:

- ✓ Conexión mediante servicios Web para aumentar la rapidez de respuesta a las peticiones realizadas por el usuario.
- ✓ Utilización principalmente de JSON como formato de respuesta para mantener los fragmentos de datos tan pequeños como sea posible y disminuir la carga.

1.6. Tecnologías asociadas al desarrollo del sistema

A continuación se exponen las herramientas y metodología de desarrollo seleccionados para el diseño e implementación de la interfaz Web Ecumene Pyxel.

1.6.1. Marcos de trabajo que soportarán el desarrollo

Con el objetivo de evitar la redundancia de código se hace indispensable el uso de un marco de trabajo. Para el desarrollo de la interfaz Web Ecumene Pyxel se propone utilizar Symfony 2.

1.6.1.1. Symfony 2

Symfony 2 es un marco de desarrollo para PHP, desarrollado completamente con PHP 5 diseñado para optimizar el desarrollo de las aplicaciones Web. Utiliza una licencia MIT¹⁷, con la que se puede hacer aplicaciones Web comerciales, gratuitas y/o de software libre. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación Web (Modelo - Vista - Controlador). Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web compleja. Además, automatiza las tareas más comunes, esto permite al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

Una aplicación desarrollada con el marco de trabajo Symfony 2 contará con código claro y organizado consistentemente. Symfony 2 promueve la reutilización y permite a los nuevos desarrolladores ser productivos y rápidos en la realización del proyecto. No es necesario desarrollar o mantener servicios públicos de bajo nivel tales como la carga automática de clases o la reproducción de controladores (Pacheco, 2011).

Symfony2 proporciona el acceso a herramientas de código abierto tales como Doctrine, plantillas, seguridad, formularios, validación y traducción. Permite que las URL sean totalmente flexibles gracias al componente *Routing*. La arquitectura centrada en HTTP de Symfony 2 da acceso a poderosas herramientas como la memoria caché HTTP (Pacheco, 2011).

¹⁷ Licencia de código abierto y libre.

Symfony 2 hace uso de línea de comandos (consola - CLI), para la creación de los proyectos, si bien esto puede crear confusión al principio, esta característica permite el ahorro en tiempo de muchas de las tareas repetitivas de una aplicación Web.

Se pueden crear aplicaciones en varios idiomas. La internacionalización que está integrada en el marco de trabajo, es estable, sigue los estándares XLIFF¹⁸, es muy completa y está probada en aplicaciones reales. Symfony 2 desecha a PHP 4 y tiene como restricción PHP 5.3.8 o superior, con lo que mejora su rendimiento (Tupe, y otros, 2008).

1.6.1.2. jQueryUI 1.8

jQueryUI es una librería para JavaScript, liberada bajo la licencia GPL. Consiste en un único fichero JavaScript que contiene las funcionalidades comunes de DOM¹⁹, eventos, efectos y AJAX. La característica principal de la biblioteca es que permite cambiar el contenido de una página Web sin necesidad de recargarla, mediante la manipulación del árbol DOM y peticiones AJAX.

Se propone utilizar en el desarrollo de la interfaz Web pues gestiona funcionalidades necesarias para el proyecto, de esa manera no es necesario programarlas y se ahorra tiempo. Garantiza la creación de páginas dinámicas y disminuye las recargas al servidor Web. Esto aumenta en gran medida la usabilidad del sistema.

1.6.2. Entornos integrados de desarrollo (IDE)

Para el desarrollo de la interfaz Web Ecumene Pyxel se utiliza como entorno integrado de desarrollo NetBeans 7.1, a continuación se expondrán algunas de sus principales características.

1.6.2.1. NetBeans 7.1

¹⁸ Formato basado en XML creado para estandarizar localización.

¹⁹ Modelo de Objetos del Documento.

Es un software liberado bajo la licencia dual CDDL²⁰ en su versión 1.0 y GNU/GPL versión 2. NetBeans es un entorno de desarrollo integrado (IDE por sus siglas en inglés), una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Existe además un número importante de módulos para extender el IDE NetBeans. Es un producto libre y gratuito sin restricciones de uso. Es de código abierto escrito completamente en Java. Soporta el desarrollo de todos los tipos de aplicaciones Java (J2SE, Web, EJB y aplicaciones móviles). NetBeans cuenta con un complemento para integrarlo con Twig²¹, que es el motor que utiliza Symfony 2 para el desarrollo de plantillas Web, además soporta PHP 5, CSS 3, HTML 5 y es sencillo de integrar con el marco de trabajo Symfony 2.

1.6.3. Los Sistemas de Gestión de Base de Datos (SGBD)

Los Sistemas de Gestión de Base de Datos (SGBD) son herramientas de software que proporcionan una interfaz entre los datos almacenados y las aplicaciones que permiten acceder a los mismos. Posibilitan crear y mantener una base de datos, que asegure su integridad, confidencialidad y seguridad. De acuerdo a las especificaciones del cliente se define como restricción el uso del SGBD PostgreSQL con versiones superiores a la 8.4.

1.6.3.1. PostgreSQL 8.4

PostgreSQL es un SGBD liberado bajo la licencia BSD. Este gestor soporta alta concurrencia mediante un sistema denominado acceso concurrente multi-versión (MVCC por sus siglas en inglés). PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otros SGBD, esto elimina la necesidad del uso de bloqueos explícitos. Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables. A pesar de que consume una gran cantidad de recursos y es un poco lento, PostgreSQL posee una gran escalabilidad, pues es capaz de ajustarse al número de CPU 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. Implementa el uso de sub-

²⁰ Licencia de código abierto y libre.

²¹ Motor de plantilla HTML.

consultas y transacciones, esto permite que su funcionamiento sea más eficaz, y ofrece soluciones en campos en los que otros SGBD no podrían. Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel (GEITEL, 2011).

Una de las principales causas por la cual se propone utilizar este SGBD es su utilización por el núcleo del producto Ecumene Pyxel para el almacenamiento de las imágenes digitales.

1.6.4. Lenguajes para la Web

Actualmente existen diferentes lenguajes para desarrollar en la Web, estos han surgido debido a las tendencias y necesidades de las plataformas. Desde los inicios de Internet, emergieron diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. A medida que pasó el tiempo, las tecnologías fueron desarrollándose y aparecieron nuevos problemas que originaron nuevos lenguajes dinámicos. A continuación presentaremos los lenguajes de programación utilizados.

1.6.4.1. PHP 5.3.8

PHP 5 es un lenguaje de *script* interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse.

Es fácil de aprender, se caracteriza por ser un lenguaje rápido, soporta la orientación a objeto, clase y herencia, es un lenguaje multiplataforma. Tiene una capacidad de conexión con la mayoría de los manejadores de bases de datos, es libre y se representa como alternativa de fácil acceso para todos.

PHP 5 es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor Web en forma de módulo o ejecutado como un binario separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. PHP 5 es un lenguaje seguro si se seleccionan correctamente las opciones de configuración en tiempos de compilación y ejecución, y se siguen las prácticas correctas de programación (Gallús, 2009).

1.6.4.2. HTML 5

Es el lenguaje de marcado predominante para la elaboración de páginas Web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de etiquetas, también puede describir, hasta cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores Web y otros procesadores de HTML. Es sencillo de comprender, presenta un texto estructurado de forma agradable, no se necesita grandes conocimientos cuando se cuenta de un editor de páginas Web, contiene archivos pequeños, despliegue rápido, es un lenguaje fácil de aprendizaje y lo admiten todos los navegadores.

1.6.4.3. CSS 3

Proporciona estilo y formato a las páginas Web. CSS 3 es un lenguaje de hojas de estilos para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML²². CSS es la mejor forma de separar los contenidos y su presentación. Es imprescindible para crear páginas Web complejas. Al crear una interfaz Web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos. Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página.

1.6.4.4. JavaScript

JavaScript es un lenguaje de programación interpretado, no requiere compilación. Utilizado principalmente en páginas Web. Lenguaje basado en prototipos, imperativo, débilmente tipado y dinámico. La mayoría de los navegadores en sus últimas versiones interpretan código JavaScript. Para evitar incompatibilidades la *World Wide Web Consortium* (W3C²³) diseñó un estándar denominado DOM (en inglés *Document Object Model*, su traducción al español Modelo de Objetos del Documento). Permite mejoras en la interfaz de

²² Lenguaje marcado de hipertexto extensible.

²³ www.w3c.es/

usuario y páginas Web dinámicas.

Añadir propiedades interactivas a las páginas Web de una forma rápida y sencilla es un deseo de muchos desarrolladores de la *World Wide Web*. Gracias a su facilidad de uso, el lenguaje de programación JavaScript ha eclipsado en popularidad a otros populares lenguajes de Internet como Java, ActiveX e incluso CGI (Interfaz de pasarela común). Al estar más próximo al Lenguaje de Marcado de Hipertexto (HTML), JavaScript se encuentra en más páginas Web que cualquier otro lenguaje de guiones o de programación (Fretzen, y otros, 2004).

1.6.5. Ingeniería del Software asistida por computadoras (CASE)

1.6.5.1. Visual Paradigm 8.0

Como herramienta para el modelado se utiliza Visual Paradigm 8.0 para UML (Lenguaje de Modelado Unificado). Producto liberado bajo la licencia *Creative Common*²⁴. Permite representar modelos de procesos, modelos de clases, diagramas de componentes, diagrama de despliegue y modelos entidad-relación. Brinda funcionalidades para la generación de código fuente a partir de diagramas de clases. Posibilita crear las bases de datos directamente en el Sistema Gestor de Base de Datos (SGBD) a partir de un diagrama entidad-relación o crear este último a partir de la base de datos mediante el uso de los controladores específicos del SGBD empleado.

1.6.6. Metodologías de desarrollo

1.6.6.1. SXP 2

Propuesta de metodología de desarrollo de software ágil compuesta por las metodologías SCRUM y XP, utilizada para proyectos de breve plazo, pequeño equipo y cuyo plazo de entrega es bien corto. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto (Romero, 2008).

²⁴ <http://creativecommons.org/>

Se propone utilizar la metodología SXP pues es un sistema a corto plazo, con un equipo de desarrollo pequeño. Además de la simplicidad al desarrollar y codificar los módulos del sistema. La retroalimentación concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales, es indispensable para la satisfactoria realización del sistema. Esta metodología promueve el desarrollo de la creatividad y aumenta la preocupación y responsabilidad de cada uno de los miembros del equipo.

2. Capítulo 2: Propuesta de solución para la interfaz Web Ecumene Pyxel

En este capítulo se realiza la descripción de la solución propuesta, mediante los diagramas de procesos correspondientes al sistema. Son enunciados los requisitos funcionales y no funcionales con los cuales debe cumplir el producto final, además de mostrar las historias de usuarios con sus prototipos no funcionales y el diagrama de paquetes. Igualmente se define la arquitectura propuesta, así como los patrones de diseño y arquitectónicos que serán utilizados. Por último se expone el diagrama de clases y el diseño de la base de datos utilizado por el sistema.

2.1. Propuesta del sistema

En la actualidad, coexisten en el mundo los más diversos sistemas de gestión documental: desde el simple registro manual de la correspondencia que entra y sale, hasta los más sofisticados sistemas informáticos. Por lo que es necesario que las empresas opten por las facilidades que brindan estos sistemas de integración y los adapten a las especificidades y necesidades de cada institución, para entender así cuestiones como la forma de visualizarse los contenidos y las funcionalidades.

En la Universidad de las Ciencias Informáticas se desarrolló un sistema para la gestión de los recursos fotográficos basado en el ECM Alfresco. Este sistema ofrece ventajas pero no cuenta con una capa de presentación amigable para el usuario, esto trae como consecuencia:

- ✓ Difícil manejo del sistema por los usuarios.
- ✓ Demora en la realización de los procesos pues estos pueden ser automatizados.
- ✓ El sistema no cuenta con la arquitectura de información requerida para la gestión de las imágenes digitales.
- ✓ La curva de aprendizaje del sistema es muy elevada.

Debido a esta problemática se propone desarrollar una interfaz Web amigable para la interactividad del usuario con la aplicación Ecumene Pyxel. Por lo cual se hace necesario el desarrollo de una capa de

presentación que consuma todos los servicios que brinda el Gestor de Contenido Empresarial Alfresco y específicamente el núcleo del sistema, mediante el uso de los servicios Web para la comunicación.

La interfaz Web para el sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel es un conjunto de soluciones informáticas para la Web, basada en estándares abiertos para la gestión de imágenes que debe permitir seleccionar, identificar, describir, representar, recuperar, publicar estos recursos y generar automáticamente diferentes perfiles a partir del formato original que se ha incorporado al sistema. Con esta interfaz se espera lograr mayor calidad e inmediatez en la gestión documental de las imágenes digitales y por consiguiente un ahorro en cuanto a las duplicaciones innecesarias de estos recursos.

La interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales debe permitir la gestión de imágenes, unidades documentales, galerías temáticas y carpetas. A continuación se observa el ciclo de vida de una imagen en el sistema (ver Figura 3), donde intervienen los grupos de usuarios fotógrafo, editor, gestor digitalizador, gestor de la información e invitado.

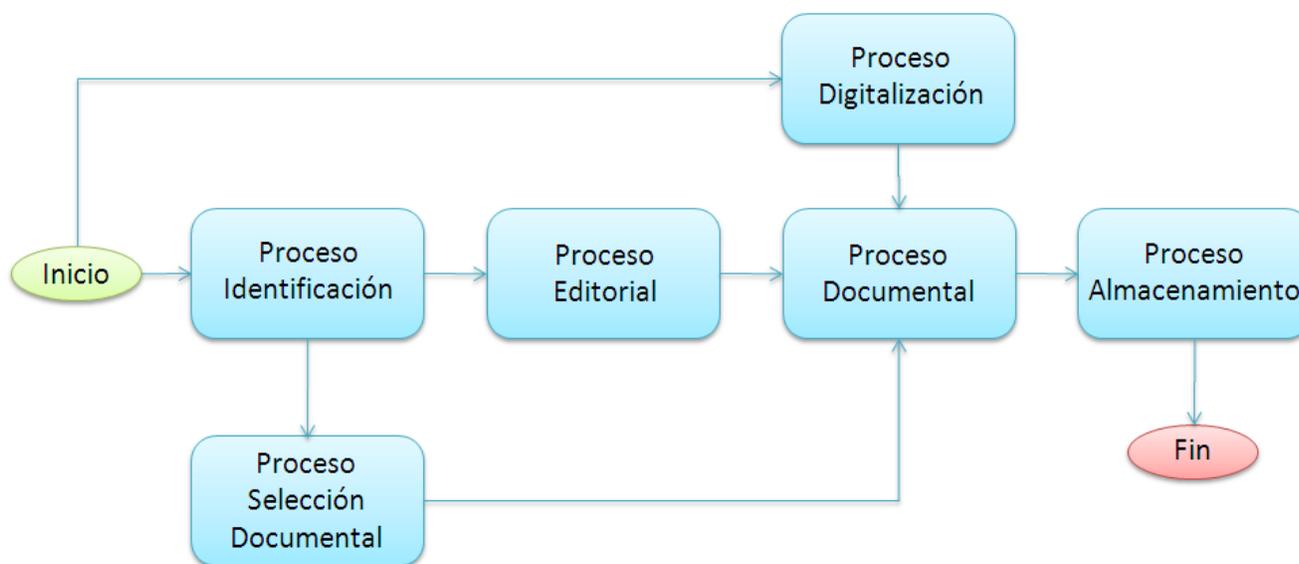


Figura 2. Ciclo de vida de una imagen en el sistema

Proceso Identificación: Es el proceso inicial del ciclo de vida de una imagen. Tiene como objetivo obtener de una fuente fiable, el fotógrafo, la información relativa a la imagen, mediante los atributos identificativos que constituyen los elementos que servirán para la descripción posterior de las imágenes. El fotógrafo puede enviar la imagen al proceso editorial si desea publicar la imagen o de lo contrario enviarla a selección documental (ver Anexo 1).

Proceso Digitalización: Proceso inicial del ciclo de vida de una imagen. Tiene como objetivo obtener de una fuente fiable, el gestor digitalizador, la información relativa a la imagen, mediante los atributos identificativos que constituyen los elementos que servirán para la descripción posterior de las imágenes. El gestor digitalizador envía la imagen al proceso documental (ver Anexo 1).

Proceso Editorial: Se realiza una selección de la imagen en cuanto a si va a ser publicada o no. Si esta es publicada el editor modifica sus propiedades. Luego la imagen es enviada a procesamiento (ver Anexo 1).

Proceso Selección Documental: Se realiza una selección de la imagen según los intereses de la institución, características, funciones, atributos y valores del material. El gestor toma decisiones en cuanto a la conservación o descarte definitivo del material (ver Anexo 1).

Proceso Documental: Se realiza la descripción formal y de contenido, indización, el resumen, así como la organización y almacenamiento, específicamente aquellos elementos relacionados con la ordenación y custodia, así como con la conservación y restauración de dichos documentos (ver Anexo 1).

Proceso Almacenamiento: En este proceso se almacenan las imágenes en las colecciones difusión, congelada o conservada. Es la etapa en la cual el material se encuentra disponible y accesible al usuario, según sus necesidades particulares. Una vez que esta etapa se completa, comienza una nueva fase del ciclo de vida documental (ver Anexo 1).

2.2. Captura de los requisitos

Los requisitos surgen como ideas que pueden originarse como una respuesta a una amenaza percibida o competencia del mercado, de una imposición legal, del deseo de crear un nuevo sistema, de la necesidad de reemplazar un sistema existente o alguna otra necesidad percibida (Romero, 2008).

2.2.1. Lista de Reserva del Producto

La Lista de Reserva del Producto (LRP) es una lista priorizada que define el trabajo que se va a realizar en el proyecto (requisitos funcionales y no funcionales). Esta lista puede crecer y modificarse a medida que se obtiene más conocimiento acerca del producto y del cliente. El objetivo es asegurar que el producto definido al terminar la lista sea correcto, útil y competitivo (Romero, 2008). Los requisitos funcionales son declaraciones de las funcionalidades que debe hacer el sistema.

Ítem	Descripción	Estimación (Semanas)	Estimado por
Muy Alta			
1	Autenticar usuario.	2	Analista
2	Adicionar imagen.	1	Analista
3	Modificar detalles de la imagen.	1	Analista
4	Eliminar imagen.	0.5	Analista
5	Visualizar detalles de la imagen.	0.5	Analista
6	Adicionar unidad documental.	0.5	Analista
7	Modificar detalles de la unidad documental.	1	Analista
8	Eliminar unidad documental.	0.5	Analista
9	Visualizar detalles de la unidad documental.	0.5	Analista
10	Adicionar galería temática.	0.5	Analista
11	Modificar detalles de la galería temática.	1	Analista

12	Eliminar galería temática.	0.5	Analista
13	Visualizar detalles de la galería temática.	0.5	Analista
14	Adicionar carpeta.	0.5	Analista
15	Modificar detalles de la carpeta.	1	Analista
16	Eliminar carpeta.	0.5	Analista
17	Visualizar detalles de la carpeta.	0.5	Analista
18	Mover imagen desde espacio personal.	1	Analista
19	Mover imagen desde espacio institucional.	1	Analista
20	Mover imagen desde selección editorial.	1	Analista
21	Mover imagen desde editorial.	1	Analista
22	Mover imagen desde selección documental.	1	Analista
23	Mover imagen desde procesamiento.	1	Analista
24	Mover imagen desde colecciones.	1	Analista
25	Restaurar imagen.	1	Analista
26	Realizar búsqueda simple.	2	Analista
27	Mostrar resultados de la búsqueda.	2	Analista
Alta			
28	Visualizar imágenes añadidas a la colección difusión.	0.5	Analista
29	Visualizar imágenes añadidas a la colección conservada.	0.5	Analista
30	Visualizar imágenes añadidas a la colección congelada.	0.5	Analista
31	Visualizar galerías temáticas.	0.5	Analista
32	Visualizar unidades documentales.	1	Analista
33	Visualizar carpetas.	1	Analista
34	Visualizar imágenes añadidas al espacio personal del usuario.	0.5	Analista

35	Visualizar imágenes añadidas al espacio institucional.	0.5	Analista
36	Visualizar imágenes añadidas a selección editorial.	0.5	Analista
37	Visualizar imágenes añadidas a selección documental.	0.5	Analista
38	Visualizar imágenes añadidas a editorial.	0.5	Analista
39	Visualizar imágenes añadidas a procesamiento.	0.5	Analista
40	Descargar Imagen.	1	Analista
Media			
41	Adicionar asociación a una galería temática.	0.5	Analista
42	Eliminar asociación a una galería temática.	0.5	Analista
43	Adicionar acceso directo.	1	Analista
44	Eliminar acceso directo.	0.5	Analista
45	Visualizar acceso directo.	0.2	Analista
Baja			
46	Visualizar ayuda en línea.	1	Analista
47	Visualizar contáctenos.	0.5	Analista

Tabla 1. LRP – Requisitos funcionales

Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requisitos no funcionales se aplican al sistema en su totalidad.

Ítem	Requisito	Observaciones
Requisitos de Software		
50	Sistema operativo a utilizar: Ubuntu 10.04 o superior.	
51	Sistema Gestor de Bases de Datos:	

	PostgreSQL con una versión mayor o igual a la versión 8.4.	
52	Debe utilizarse como navegador Mozilla Firefox en su versión 3.6 o superior.	
53	El producto debe implementarse hasta el máximo posible, con herramientas de código abierto, sobre la filosofía de Software Libre.	
Requisitos de Hardware		
54	El servidor de la interfaz debe tener al menos: 1 GB RAM, Core 2 Duo, HDD (80 - 100 GB).	La cantidad de memoria RAM depende de la cantidad de usuarios concurrentes en el sistema, igual que la cantidad de microprocesadores.
55	El servidor de base de datos debe tener al menos: 2 GB RAM, Core i3, HDD (entre 250 y 500 GB), soporte para el modo RAID ²⁵ .	
56	El servidor del núcleo del sistema debe tener al menos: 2 GB RAM, Core i3, HDD (entre 250 y 500 GB), soporte para el modo RAID.	
Requisitos de Soporte		
57	La documentación resulta requisito indispensable como evidencia de la evolución del Proyecto. <ul style="list-style-type: none"> ✓ Historias de usuario. ✓ Casos de prueba de funcionalidad. ✓ Arquitectura de software. 	
Requisitos de Seguridad		
58	Se controlarán los permisos de acceso, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema.	

²⁵ Sistema de almacenamiento que usa múltiples discos duros.

Requisitos Legales		
59	El sistema debe ser sometido a un análisis legal por parte de los abogados y personal autorizado con vistas a declarar su autenticidad y evitar restricciones legales para su uso y comercialización; así mismo se debe proceder a una evaluación y certificación por parte del cliente del producto.	
Usabilidad		
60	Especificación de la terminología utilizada: el sistema debe adaptarse al lenguaje y términos utilizados por los usuarios en la rama abordada con vista a una mayor comprensión por parte del cliente de la herramienta de trabajo.	
61	Facilidad de uso por parte de los usuarios: el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe además, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.	

Tabla 2. LRP - Requisitos no funcionales

2.3. Historias de usuario

Las historias de usuario son la técnica utilizada en la metodología SXP para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado. Las mismas describen las tareas que el sistema debe hacer y su construcción depende principalmente de la habilidad con la que son definidas. Son utilizadas como el único documento de requisitos que se genera en SXP. Son escritas en lenguaje natural, sin un formato predeterminado y en pocas líneas de texto (Romero, 2008).

A continuación se exponen algunas historias de usuario especificadas para guiar la implementación de la interfaz Web.

2.3.1. Autenticar usuario

Historia de usuario	
Número: 01	Usuario: imgarcia
Nombre historia: Autenticar usuario.	
Prioridad en negocio: Muy Alta.	Riesgo en desarrollo: 3, 5, 7, 8, 13. (ver Anexo 2)
Puntos estimados: 2	Iteración asignada: Primera iteración.
Programador responsable: Sergio Antonio Segura Fernández.	
<p>Descripción:</p> <p>Un usuario necesita autenticarse en el sistema. Para hacerlo, abre su navegador y escribe la dirección URL del Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel (http://epyxel.uci.cu), el sistema responde con una página de invitado donde muestra las últimas imágenes añadidas en la colección de difusión y las galerías temáticas disponibles para el uso público, en la parte superior derecha aparece un formulario de autenticación, donde debe llenar los siguientes campos:</p> <p>Nombre de usuario* (TEXT) (20 caracteres): <i>gestor</i></p> <p>Contraseña* (PASSWORD) (20 caracteres): <i>gestor</i></p> <p>Luego de introducir los datos el usuario debe presionar el botón que se encuentra a la derecha del formulario para enviar los datos al sistema. Si los datos son introducidos y validados correctamente, el sistema cargará el espacio de trabajo de dicho usuario, en caso contrario mostrará un mensaje de error.</p>	
<p>Observaciones:</p> <ul style="list-style-type: none"> ✓ Si el usuario y contraseña introducidos son válidos en el ECM Alfresco pero no en la interfaz Web, el sistema mostrará un mensaje de error que indique que el usuario no es válido en la interfaz Web. ✓ Si el campo "Nombre de Usuario" está vacío el sistema mostrará un mensaje de error que le señala al usuario que ese campo no puede estar vacío y continúa en la página inicial. 	

<ul style="list-style-type: none"> ✓ Si el campo “Contraseña” está vacío el sistema mostrará un mensaje de error que le señala al usuario que ese campo no puede estar vacío y continúa en la página inicial. ✓ Si el usuario introducido es incorrecto el sistema mostrará un mensaje de error que indicara que el usuario o la contraseña son incorrectos y continúa en la página inicial. ✓ Si la contraseña introducida es incorrecta el sistema mostrará un mensaje de error que indicara que el usuario o la contraseña son incorrectos y continúa en la página inicial. ✓ Esta funcionalidad está habilitada para todos los usuarios del sistema. ✓ Los campos que tienen el símbolo “*”, son obligatorios.
Prototipo: (ver Anexo 3)

Tabla 3. HU - Autenticar usuario

2.3.2. Adicionar imagen

Historia de usuario	
Número: 02	Usuario: imgarcia
Nombre historia: Adicionar imagen.	
Prioridad en negocio: Muy Alta	Riesgo en desarrollo: 3, 5, 7, 8, 13 (ver Anexo 2)
Puntos estimados: 1	Iteración asignada: Segunda iteración
Programador responsable: Sergio Antonio Segura Fernández	
<p>Descripción:</p> <p>Para la adición de imágenes el usuario debe estar autenticado en el sistema. Una vez que acceda a su espacio de trabajo, según los permisos asignados, tendrá la opción de adicionar una imagen a su espacio personal:</p> <p>El usuario debe seleccionar el ícono adicionar imagen que se encuentra en la parte superior derecha del bloque “Mis Imágenes” del espacio personal de éste. A continuación debe aparecer una ventana con el siguiente campo:</p>	

Dirección de la imagen* (FILE) (50 caracteres): Imagen.jpg

Luego se debe oprimir el botón “Explorar”, el cual abre un explorador para que el usuario seleccione la imagen que desea adicionar. Luego debe oprimir el botón “Adicionar” si desea realizar la operación, en caso contrario debe oprimir el botón “Cancelar” y no se realiza ninguna operación, el sistema debe volver a la página inicial del usuario. Si el usuario oprime la opción “Aceptar” la imagen se adiciona al espacio personal del usuario.

Observaciones:

- ✓ Precondición: El usuario debe estar autenticado en el sistema para poder acceder a estas opciones.
- ✓ Solo tendrán acceso a esta funcionalidad los usuarios que pertenezcan a los grupos “fotógrafo” y “gestor”.
- ✓ Si la imagen no tiene el formato requerido (jpg²⁶, png²⁷, gif²⁸, tiff²⁹), debe devolver un mensaje de error en el que se especifica que el formato no puede ser añadido al sistema.
- Si el nombre de la imagen ya existe en el sistema, este debe devolver un mensaje de error en el que se especifica que la imagen ya existe.
- Si la imagen tiene un tamaño mayor que 4MB el sistema debe mostrar un mensaje de error en el que se especifica que la imagen es demasiado grande.
- Si la imagen es adicionada correctamente el sistema debe recargar el bloque “Mis imágenes”.
- El nombre de la imagen solo podrá contener letras y debe tener menos de 30 caracteres y más de 3. En caso contrario el sistema debe mostrar un mensaje de error.

Prototipo: (ver Anexo 3)

Tabla 4. HU - Adicionar imagen

²⁶ Estándar de compresión y codificación de archivos de imágenes fijas.

²⁷ Formato gráfico basado en un algoritmo de compresión sin pérdida.

²⁸ Formato gráfico para imágenes y animaciones.

²⁹ Formato de fichero para imágenes.

2.3.3. Eliminar imagen

Historia de usuario	
Número: 04	Usuario: imgarcia
Nombre historia: Eliminar imagen	
Prioridad en negocio: Muy Alta	Riesgo en desarrollo: 3, 5, 7, 8, 13 (ver Anexo 2)
Puntos estimados: 0,5	Iteración asignada: Segunda iteración
Programador responsable: Sergio Antonio Segura Fernández	
<p>Descripción:</p> <p>Para la eliminación de imágenes el usuario debe estar autenticado en el sistema. Una vez que acceda a su espacio de trabajo, según los permisos asignados, el usuario puede eliminar una o varias imágenes.</p> <p>Primeramente el usuario debe escoger la carpeta de su espacio personal donde se encuentran las imágenes que desea eliminar. Luego debe seleccionarlas a través del “<i>CHECKBOX</i>” correspondiente a cada una de estas imágenes, luego debe presionar el ícono “eliminar imagen” que se encuentra en la parte superior derecha del cuadrante “Mis imágenes” donde se muestran las imágenes en miniatura en el espacio personal del usuario. A continuación debe aparecer un mensaje pidiéndole al usuario una confirmación de la acción que desea realizar, si el usuario desea continuar debe oprimir el botón “Aceptar”, en ese caso se elimina la o las imágenes seleccionadas del espacio personal del usuario. Si éste oprime la opción “Cancelar” no se realiza la acción y regresa a la página inicio del usuario autenticado.</p>	
<p>Observaciones:</p> <ul style="list-style-type: none"> • Como precondition el usuario debe estar autenticado en el sistema para poder acceder a esta funcionalidad. • El usuario que pertenezca al grupo “editor” no puede eliminar imágenes del sistema. • Esta funcionalidad estará habilitada únicamente en los espacios personales de los usuarios que pertenezcan a los grupos de fotógrafo y gestor. 	

Prototipo: (ver Anexo 3)

Tabla 5. HU - Eliminar imagen

2.4. Propuesta de arquitectura

El sistema se va a conectar al ECM Alfresco, por eso se propone que la interfaz Web Ecumene Pyxel utilice la misma arquitectura de ese gestor de contenidos (ver Figura 4). La arquitectura consiste en una serie de componentes divididos en capas las cuales tienen a su cargo funciones diferentes.



Figura 3. Arquitectura del ECM Alfresco

Esta arquitectura está dividida en tres capas diferentes:

- ✓ Clientes que interactúan con los usuarios finales.
- ✓ Servidores de aplicaciones que procesan los datos para los clientes.
- ✓ Servidores de la base de datos que almacenan los datos para los servidores de aplicación.

Esta arquitectura separa el proceso de la información, lo cual mejora la carga en los diversos servidores, por lo tanto es más escalable. La separación entre las capas es de tipo lógica, por lo que pueden estar

distribuidas en el mismo sistema físico o no, esto proporciona un modelo adaptable a las tecnologías del entorno donde se implemente; aunque si se distribuyen las capas entre múltiples sistemas físicos mejoraría la escalabilidad y el rendimiento. No se replica la lógica del negocio en los clientes, lo que significa que las modificaciones y mejoras pueden ser aprovechadas automáticamente por el conjunto de usuarios, lo cual reduce los costes de mantenimiento, permite aislar los cambios en tecnologías a ciertas capas para reducir el impacto en el sistema total (Incomparable, 2011).

La arquitectura del sistema Ecumene Pyxel (ver Figura 5) está conformada por varias capas, el usuario se conecta mediante el protocolo HTTP a la interfaz Web, la cual contiene un proxy inverso que permite disminuir las peticiones al servidor y guarda información en una caché. Luego realiza peticiones a la capa donde se encuentra implementado el núcleo mediante los servicios Web que este brinda, para obtener datos de la capa de almacenamiento que serán procesados.

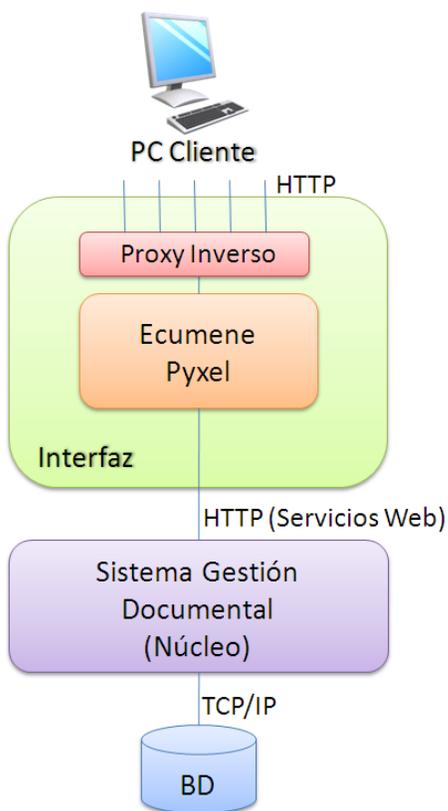


Figura 4. Propuesta de arquitectura para Ecumene Pyxel

2.4.1. Patrones de arquitectura.

Modelo-Vista-Controlador (MVC)

Symfony 2 está basado en un patrón clásico de arquitectura Web conocido como Modelo-Vista-Controlador (MVC). La utilización de este marco de trabajo para el desarrollo del producto conlleva a la implementación de este patrón, que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos:

- ✓ **Modelo:** Esta es la representación específica de la información con la cual el sistema opera. En este se encuentran las clases que son generadas de forma automática según la estructura de la base de datos.
- ✓ **Vista:** Esta presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario. La vista es la encargada de originar las páginas que son mostradas como resultado de las acciones.
- ✓ **Controlador:** Este responde a eventos, usualmente acciones del usuario, que son la lógica de la aplicación, e invoca peticiones al modelo y a la vista.

La arquitectura MVC proporciona ventajas, como la organización del código, la reutilización y la flexibilidad. Symfony 2 toma lo mejor de la arquitectura MVC y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo. Separa las capas más allá del MVC, por ejemplo, la capa del modelo se divide en la capa de acceso a los datos y en la capa de abstracción de la base de datos.

2.4.2. Patrones de diseño

Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de software. En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares. En el desarrollo de la interfaz Web Ecumene Pyxel son utilizados varios de estos patrones, a continuación son expuestos.

Patrón experto en información

Principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto o la implementación de un método debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo obtendremos un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento).

Se mantiene el encapsulamiento, los objetos utilizan su propia información para llevar a cabo sus tareas. Se distribuye el comportamiento entre las clases que contienen la información requerida. Son más fáciles de entender y mantener.

En Symfony 2 este patrón es utilizado en la inclusión de Doctrine para mapear la Base de Datos. Se utiliza para crear una capa de abstracción en el modelo, encapsular toda la lógica de los datos y generar las clases en todas las funcionalidades comunes de las entidades, las clases de abstracción de datos poseen un grupo de funcionalidades que están relacionadas directamente con la entidad que representan y contienen la información necesaria de la tabla que representan.

Patrón creador

Identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que:

- ✓ Tiene la información necesaria para realizar la creación del objeto.
- ✓ Usa directamente las instancias creadas del objeto.
- ✓ Almacena o maneja varias instancias de la clase.
- ✓ Contiene o agrega la clase.

Puede soportar un bajo acoplamiento, mayor claridad, encapsulación y reutilización.

Este patrón es utilizado en las clases controladoras donde se encuentran las acciones definidas para el sistema y se ejecutan en cada una de ellas. En dichas acciones se crean los objetos de las clases que

representan las entidades, lo que evidencia que las clases controladoras son creadoras de dichas entidades.

Patrón registro

Este patrón es útil para los desarrolladores en la Programación Orientada a Objetos. Este patrón es un medio sencillo y eficiente de compartir datos y objetos en la aplicación sin la necesidad de preocuparse por conservar numerosos parámetros o hacer uso de variables globales. Este patrón se aplica en la clase de configuración que es la encargada de guardar todas las variables globales del sistema.

Patrón decorador

En Symfony 2 la vista se separa en una plantilla base y varias plantillas que heredan de esta. Normalmente, la plantilla base es global en toda la aplicación y contiene el código HTML que es común en la mayoría de las páginas, lo cual es una implementación del patrón decorador.

2.5. Diseño de la interfaz Web Ecumene Pyxel

2.5.1. Diagrama de paquetes del sistema

A continuación se muestra el diagrama de paquetes que conforman la interfaz Web Ecumene Pyxel:



Figura 5. Diagrama de paquetes

El Sistema de Gestión de Imágenes Digitales Ecumene Pyxel está dividido en tres paquetes fundamentales para su funcionamiento:

- ✓ El Núcleo del sistema Ecumene Pyxel en el cual se realiza la gestión de todos los recursos de la aplicación. Este paquete es el resultado de la personalización del ECM Alfresco para garantizar la correcta gestión de las imágenes digitales.
- ✓ El paquete Alfresco API el cual permite la comunicación entre la interfaz Web y el núcleo mediante el empleo de los servicios Web implementados sobre el núcleo del sistema.
- ✓ El paquete interfaz Web Ecumene Pyxel que permite al usuario interactuar con los recursos del sistema de una manera sencilla y amigable.

2.5.2. Diagrama de clases del diseño autenticar usuario

A continuación se muestra el diagrama de estereotipos Web correspondiente a la funcionalidad autenticar usuario de la interfaz Web Ecumene Pyxel. Mediante este diagrama se realiza una representación gráfica de la jerarquía y las relaciones que se generan a partir de cada una de las clases del diseño.

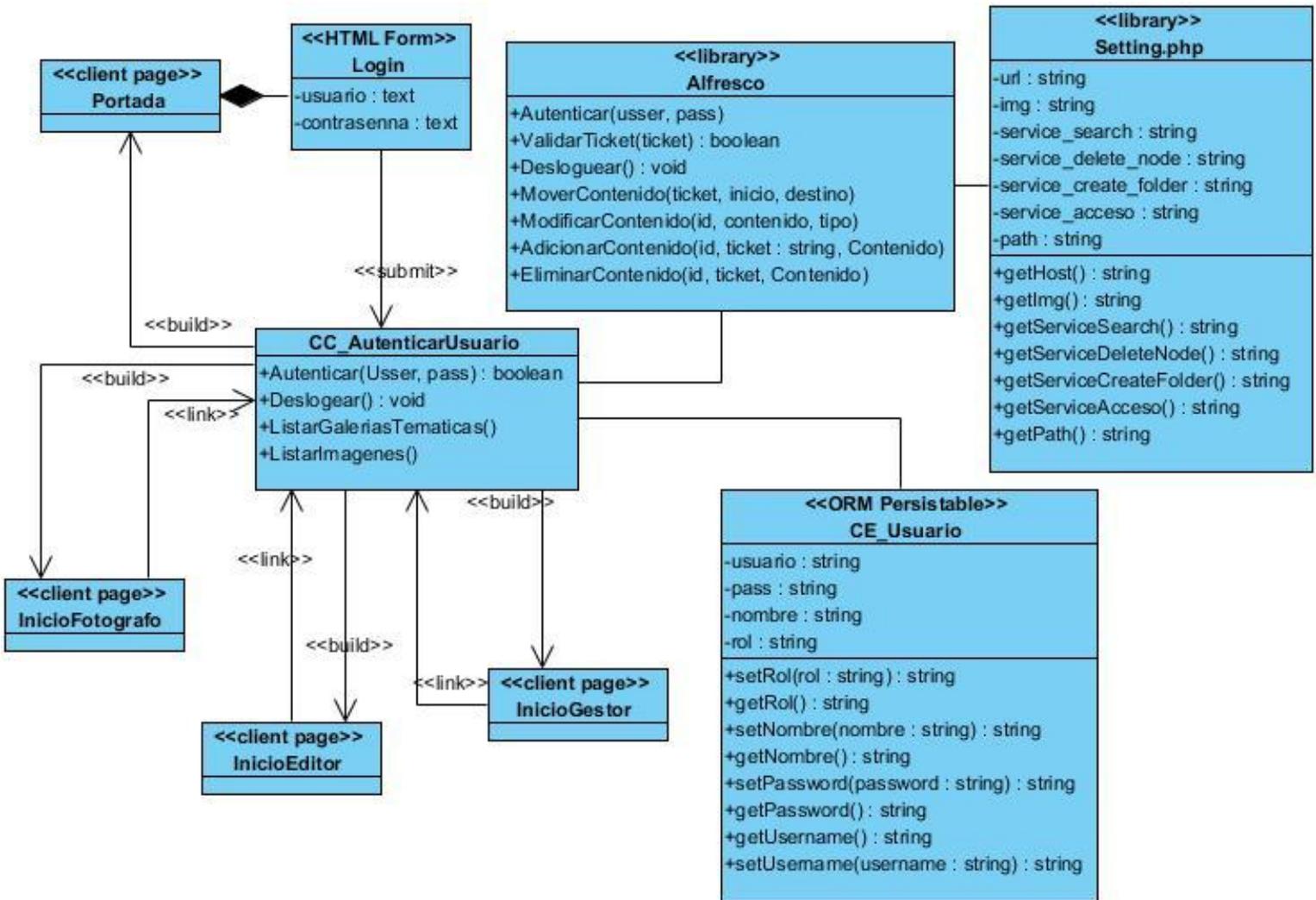


Figura 6. Diagrama de clases autenticar usuario

Para ver los restantes diagramas de clases del diseño ver Anexo 4.

2.5.3. Descripción de las clases de diseño

CC_AutenticarUsuario
Tipo de clase: Clase controladora
Para cada responsabilidad:

Nombre:	Autenticar
Descripción:	Encargada de enviar los datos correspondientes a la librería Alfresco para la autenticación del usuario en el sistema.
Nombre:	Desloguear
Descripción:	Encargada de enviarle los datos correspondientes a la librería Alfresco para que el usuario deje de estar autenticado en el sistema.
Nombre:	ListarGaleriasTematicas
Descripción:	Esta funcionalidad devuelve las imágenes que están en la colección difusión y se las envía a la plantilla Portada para mostrárselas al usuario.
Nombre:	ListarImagenes
Descripción:	Esta funcionalidad devuelve las galerías temáticas que se encuentran públicas para el usuario invitado y se las envía a la plantilla Portada.

Tabla 6. Clase CC_AutenticarUsuario

CE_Usuario	
Tipo de clase: Clase persistente.	
Atributo	Tipo
usuario	string
pass	string
nombre	string
rol	string
Para cada responsabilidad:	
Nombre:	getUsername
Descripción:	Funcionalidad encargada de devolver el usuario.
Nombre:	setUsername
Descripción:	Funcionalidad que permite modificar el usuario.
Nombre:	getPassword
Descripción:	Funcionalidad encargada de devolver la contraseña del usuario.
Nombre:	setPassword
Descripción:	Funcionalidad que permite modificar la contraseña del usuario.

Nombre:	getNombre
Descripción:	Funcionalidad encargada de devolver el nombre del usuario.
Nombre:	setNombre
Descripción:	Funcionalidad que permite modificar el nombre del usuario.
Nombre:	getRol
Descripción:	Funcionalidad encargada de devolver el rol del usuario.
Nombre:	setRol
Descripción:	Funcionalidad que permite modificar el rol del usuario.

Tabla 7. Clase CE_Usuario

Setting	
Tipo de clase: Librería.	
Atributo	Tipo
url	string
img	string
service_search	string
service_delete_node	string
service_create_folder	string
service_acceso	string
path	string
Para cada responsabilidad:	
Nombre:	getHost
Descripción:	Funcionalidad encargada de devolver el atributo url, el cual contiene el IP y el puerto donde se encuentra el núcleo del sistema.
Nombre:	getImg
Descripción:	Funcionalidad encargada de devolver el atributo img, el cual contiene la dirección para acceder a las imágenes en el núcleo del sistema.
Nombre:	getServiceSearch
Descripción:	Funcionalidad encargada de devolver el atributo service_search, el cual contiene la dirección para acceder al servicio de búsqueda.

Nombre:	getServiceDeleteNode
Descripción:	Funcionalidad encargada de devolver el atributo service_delete_node, el cual contiene la dirección para acceder al servicio para eliminar un elemento.
Nombre:	getServiceCreateFolder
Descripción:	Funcionalidad encargada de devolver el atributo service_create_folder, el cual contiene la dirección para acceder al servicio para adicionar una carpeta.
Nombre:	getServiceAcceso
Descripción:	Funcionalidad encargada de devolver el atributo service_acceso, el cual contiene la dirección para acceder a los accesos directos en el núcleo del sistema.
Nombre:	getPath
Descripción:	Funcionalidad encargada de devolver el atributo path, el cual contiene la dirección para acceder a los diferentes espacios que se encuentran en el núcleo del sistema.

Tabla 8. Clase Setting

Alfresco	
Tipo de clase: Librería.	
Para cada responsabilidad:	
Nombre:	Autenticar
Descripción:	Funcionalidad que permite crear una sección a un usuario del sistema.
Nombre:	ValidarTicket
Descripción:	Funcionalidad que permite validar la sección anteriormente creada.
Nombre:	Desloguear
Descripción:	Funcionalidad que permite cancelar una sección a un usuario del sistema.
Nombre:	MoverContenido
Descripción:	Funcionalidad que permite mover un contenido (imagen o carpeta) desde un origen a un destino.
Nombre:	ModificarContenido
Descripción:	Funcionalidad que permite modificar un contenido (imagen, unidad documental o galería temática).
Nombre:	AdicionarContenido

Descripción:	Funcionalidad que permite adicionar un contenido (imagen, unidad documental, galería temática o carpeta).
Nombre:	EliminarContenido
Descripción:	Funcionalidad que permite eliminar un contenido (imagen, unidad documental, galería temática o carpeta).

Tabla 9. Clase Alfresco

Login	
Tipo de clase: Formulario HTML.	
Atributo	Tipo
usuario	text
contrasenna	text

Tabla 10. Clase Login

Portada
Tipo de clase: Clase interfaz.

Tabla 11. Clase Portada

InicioFotógrafo
Tipo de clase: Clase interfaz.

Tabla 12. Clase InicioFotógrafo

InicioGestor
Tipo de clase: Clase interfaz.

Tabla 13. Clase InicioGestor

InicioEditor
Tipo de clase: Clase interfaz.

Tabla 14. Clase InicioEditor

2.6. Diseño de la base de datos

2.6.1. Modelo Entidad-Relación de la base de datos

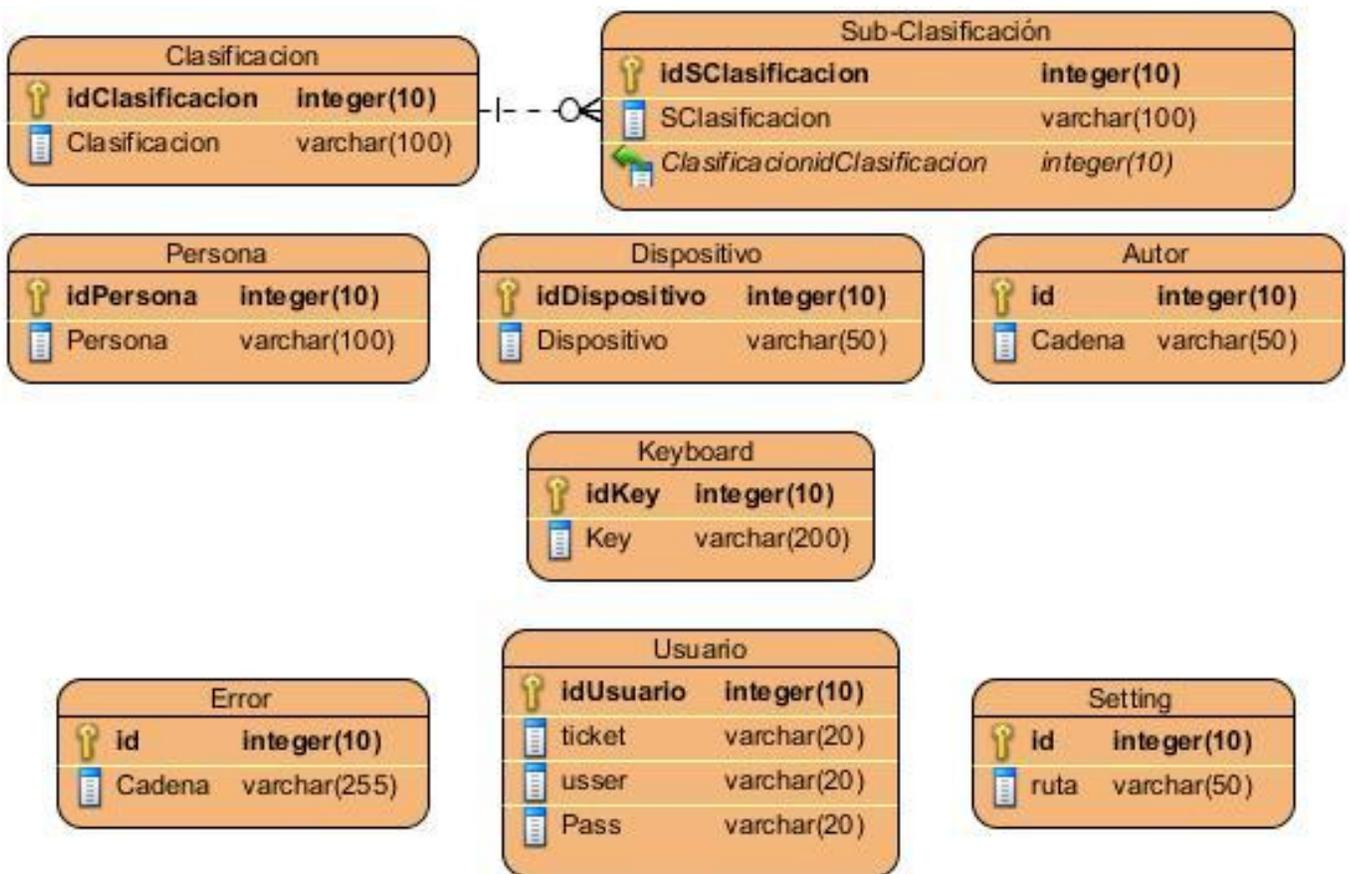


Figura 7. Modelo Entidad-Relación

2.6.2. Descripción de las tablas

Nombre: Usuario		
Descripción: Almacena temporalmente los usuarios del sistema para realizar la autenticación.		
Atributo	Tipo	Descripción
idUsuario	Integer(10)	Es un identificador del usuario.
ticket	Varchar(20)	Es una cadena que permite validar el usuario en el ECM Alfresco.
usser	Varchar(20)	Nombre de usuario con el cual se identifica para entrar en el sistema.
Pass	Varchar(20)	Contraseña necesaria para autenticarse en el sistema.

Tabla 15. Descripción de la tabla de la BD-Usuario

Nombre: Clasificación		
Descripción: Almacena las clasificaciones que puede tomar una imagen. Es utilizada para que el usuario no pueda introducir una clasificación incorrecta y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
idClasificación	Integer(10)	Es el identificador de la Clasificación.
Clasificación	Varchar(100)	Clasificaciones que puede tomar la imagen.

Tabla 16. Descripción de la tabla de la BD-Clasificación

Nombre: Sub-Clasificación		
Descripción: Almacena las sub-clasificaciones que puede tomar una imagen, es decir cada clasificación puede contener varias sub-clasificaciones. Es utilizada para que el usuario no pueda introducir una sub-clasificación incorrecta y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
idSClasificación	Integer(10)	Es el identificador de la sub-Clasificación.
SClasificación	Varchar(100)	Sub-clasificaciones que puede tomar la imagen.
ClasificacionidClasificacion	Integer(10)	Es el identificador de la Clasificación a la que pertenece.

Tabla 17. Descripción de la tabla de la BD-Sub-Clasificación

Nombre: Error		
Descripción: Almacena los posibles errores en caso que no funcionen los servidores requeridos.		
Atributo	Tipo	Descripción
idError	Integer(10)	Es el identificador del error.
Cadena	Varchar(255)	Cadena que explica exactamente de qué se trata el error.

Tabla 18. Descripción de la tabla de la BD-Error

Nombre: Autor		
Descripción: Almacena los autores que puedan haber creado una imagen. Es utilizada para que el usuario no pueda introducir un autor incorrecto y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
id	Integer(10)	Es el identificador del Autor.
Cadena	Varchar(50)	Nombre del posible autor de la imagen.

Tabla 19. Descripción de la tabla de la BD- Autor

Nombre: Setting		
Descripción: Almacena las direcciones necesarias para la conexión y consumo de los servicios del ECM Alfresco.		
Atributo	Tipo	Descripción
id	Integer(10)	Es el identificador de cada dirección.
ruta	Varchar(50)	Ruta de cada una de las opciones.

Tabla 20. Descripción de la tabla de la BD-Setting

Nombre: Persona		
Descripción: Almacena las personas que puedan aparecer en una imagen. Es utilizada para que el usuario no pueda introducir una persona incorrecta y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
idPersona	Integer(10)	Es el identificador de la Persona.

Persona	Varchar(100)	Nombre de la persona que aparece en la imagen.
---------	--------------	--

Tabla 21. Descripción de la tabla de la BD-Persona

Nombre: Dispositivo		
Descripción: Almacena los dispositivos con que puede ser tomada una imagen. Es utilizada para que el usuario no pueda introducir un dispositivo incorrecto y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
idDispositivo	Integer(10)	Es el identificador del dispositivo.
Dispositivo	Varchar(50)	Dispositivos con que puede ser tomada una imagen.

Tabla 22. Descripción de la tabla de la BD-Dispositivo

Nombre: Keyboard		
Descripción: Almacena las palabras claves que puede tener una imagen. Es utilizada para que el usuario no pueda introducir una palabra clave incorrecta y autocompletar el campo del formulario con los valores de la tabla.		
Atributo	Tipo	Descripción
idKey	Integer(10)	Es el identificador de la palabra clave.
Key	Varchar(200)	Palabras clave que puede tener una imagen.

Tabla 23. Descripción de la tabla de la BD-Keyboard

3. Capítulo 3: Construcción de la propuesta de solución a la interfaz Web Ecumene Pyxel y validación de los resultados

En este capítulo se expone el diagrama de componentes que permite modelar, de manera estructurada, los elementos que componen el sistema. Además se define el diagrama de despliegue en el cual se especifica cómo quedarán distribuidos físicamente los recursos utilizados por el sistema para su correcto desempeño. Conjuntamente se proponen los casos de prueba funcionales, los cuales acreditarán el correcto desempeño de la aplicación.

3.1. Diagrama de componentes

Los diagramas de componentes representan cómo el sistema está dividido en un conjunto de componentes, además refleja la organización y las dependencias entre estos componentes. En él se presentan librerías, archivos, ejecutables, tablas y documentos que forman parte del sistema.

A continuación se presenta el diagrama de componentes de la interfaz Web Ecumene Pyxel.

3.1.1. Representación gráfica del diagrama de componentes

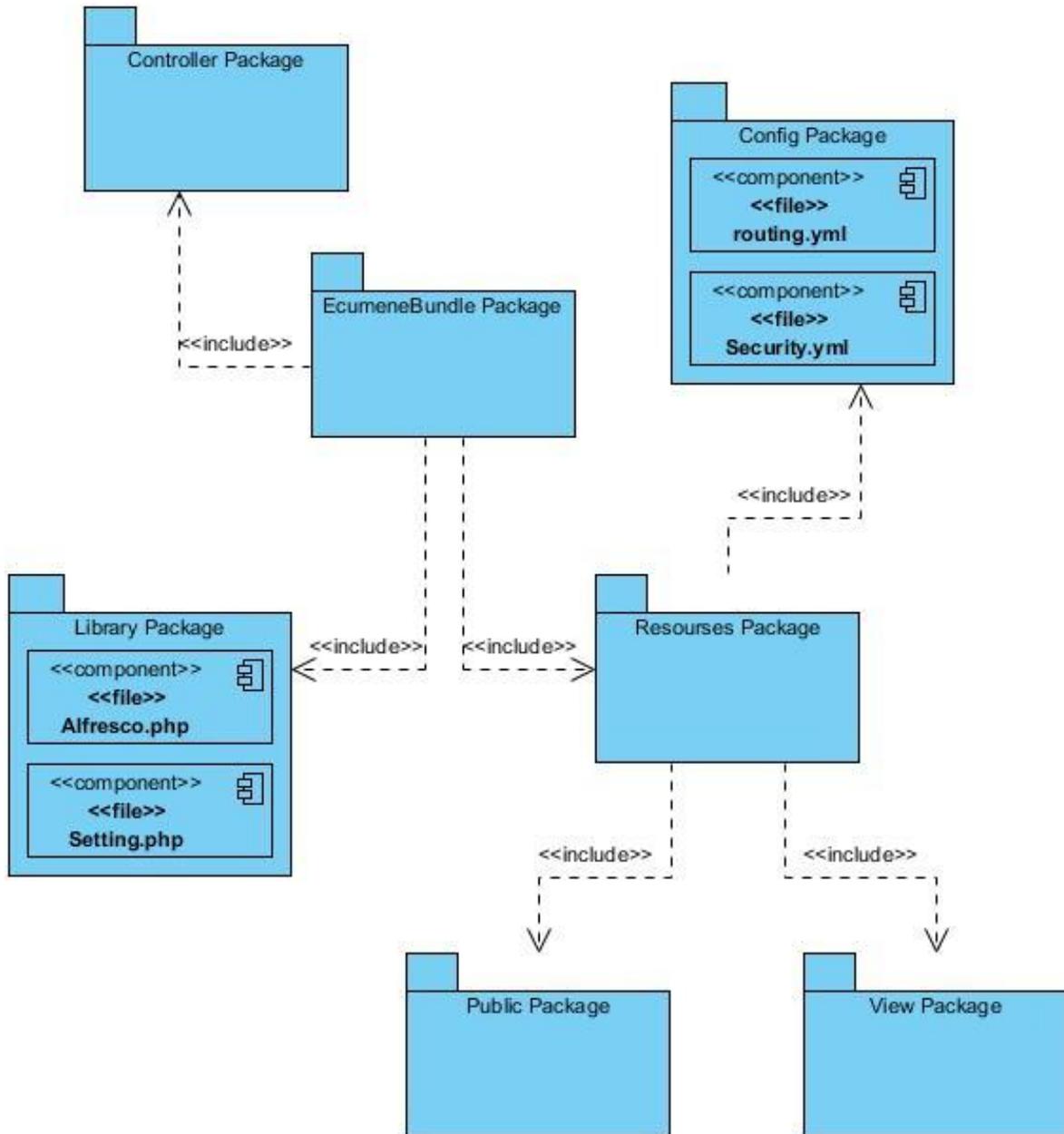


Figura 8. Diagrama de componentes

Para ver los restantes diagramas de componentes ver Anexo 5.

3.1.2. Descripción del diagrama de componentes

Para el correcto funcionamiento de la aplicación se definen varios paquetes en los cuales se dividen la lógica, la vista, la modelación de los datos, la comunicación con el núcleo, las configuraciones del sistema y los elementos públicos.

- ✓ El paquete *Public* contiene todos los componentes que posibilitan que la interfaz sea amigable, como son las imágenes, los íconos, los CSS, y los archivos JavaScript.
- ✓ El paquete *View* está conformado por las plantillas HTML de la aplicación, las cuales serán las encargadas de mostrar la información al usuario.
- ✓ El paquete *Config* almacena los archivos de configuración del sistema.
- ✓ El paquete *Resources* es el encargado de agrupar todos los paquetes mencionados anteriormente.
- ✓ El paquete *Library* contiene los componentes encargados de la comunicación con el núcleo del sistema.
- ✓ El paquete *Controller* está formado por las clases controladoras de la aplicación.
- ✓ En el paquete *EcumeneBundle* se agrupan todos los componentes, dándole una estructura organizativa al producto.

3.2. Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como una computadora, un dispositivo o memoria. Los nodos se interconectan mediante soportes bidireccionales que pueden a su vez estereotiparse (Huallpara, y otros, 2009).

A continuación se muestra el diagrama de despliegue propuesto para el sistema Ecumene Pyxel, donde quedan definidas las relaciones entre los diferentes nodos físicos que lo componen (ver Figura 10).

3.2.1. Representación gráfica del diagrama de despliegue

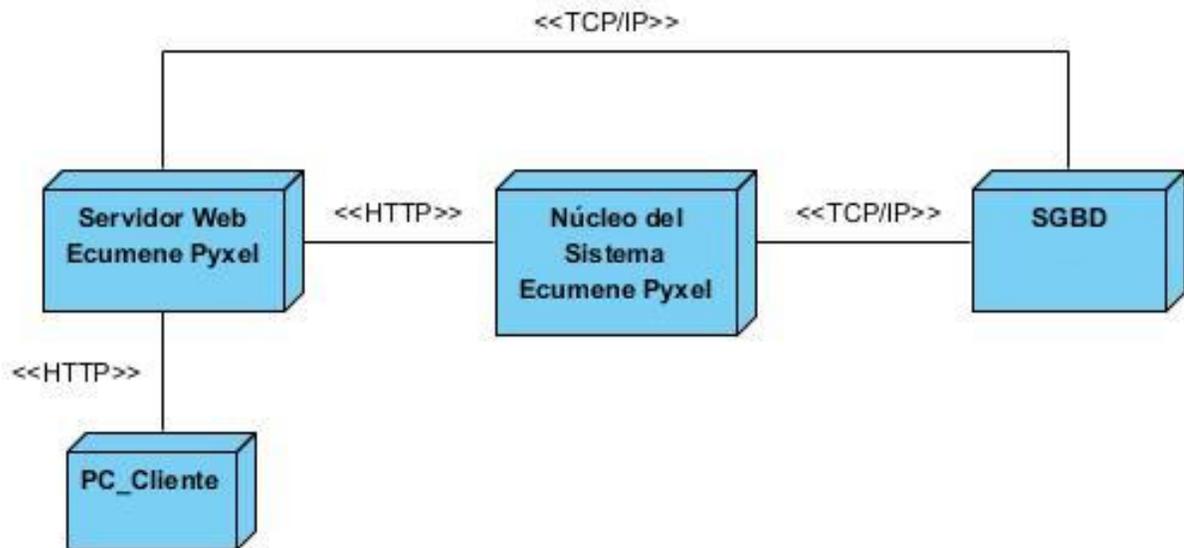


Figura 9. Diagrama de despliegue

3.2.2. Descripción del diagrama de despliegue

El despliegue del sistema estará compuesto por una PC³⁰ del lado del cliente, que accederá al Servidor Web Ecumene Pyxel mediante un navegador Web, y empleará el protocolo HTTP.

El servidor Web Ecumene Pyxel podrá comunicarse con el núcleo del sistema a través el protocolo HTTP y consumir los servicios Web que este brinda.

Existe un servidor donde se encuentra el Sistema Gestor de Base de Datos que contendrá las dos bases de datos necesarias para la aplicación, a este servidor podrán conectarse tanto el núcleo del sistema como la interfaz Web mediante el protocolo TCP/IP³¹.

³⁰ Computadora personal.

³¹ Modelo de descripción de protocolos de red.

3.3. Diseño y ejecución de las pruebas de software

3.3.1. Pruebas de funcionalidad

Se denominan pruebas funcionales a las pruebas de software que tienen por objetivo probar que los sistemas desarrollados cumplan con las funciones específicas para los cuales han sido creados. A este tipo de pruebas se les denomina también pruebas de comportamiento y para realizarlas se emplea el método de caja negra, donde los probadores o analistas de pruebas no enfocan su atención en cómo se generan las respuestas del sistema, sino en el funcionamiento de la interfaz del sistema. Básicamente el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y en los de salida, esto generalmente se define en los casos de prueba preparados antes del inicio de las pruebas (Calidad y software, 2009).

El objetivo final de éstas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable.

A continuación se muestran la propuesta de los casos de prueba funcionales a realizar a la interfaz Web Ecumene Pyxel según las historias de usuarios anteriormente expuestas.

Caso de Prueba		
Código Caso de Prueba: EP-101-1	Nombre Historia de Usuario: Autenticar usuario.	
Nombre de la persona que realiza la prueba: Isabel María García Hilarión		
Descripción de la Prueba: Funcional		
Condiciones de ejecución: <ul style="list-style-type: none"> ✓ El usuario debe entrar al navegador Web y conectarse al sistema mediante la dirección http://epyxel.uci.cu/usuario/inicio. ✓ Esta funcionalidad está habilitada para todos los usuarios del sistema. 		
Entrada / Pasos de ejecución: Se deben llenar los siguientes campos: <ul style="list-style-type: none"> ✓ Usuario* (TEXT) (20 caracteres) ✓ Contraseña*(PASSWORD) (20 caracteres) <p>Luego presionar el botón autenticar.</p>		
Escenarios:	Resultados Esperados	Evaluación
Entrada de los datos de autenticación correctamente. Usuario: "gestor" Contraseña: "gestor"	Entrar al espacio personal del usuario autenticado.	Satisfactoria
El campo de usuario es incorrecto. Usuario: "asd" Contraseña: "gestor"	Mensaje de error que especifica "El usuario o la contraseña están incorrectos."	Satisfactoria
El campo de contraseña es incorrecto. Usuario: "gestor" Contraseña: "asd"	Mensaje de error que especifica "El usuario o la contraseña están incorrectos."	Satisfactoria
El campo Usuario está vacío.	Mensaje de error que especifica "El campo	Satisfactoria

Usuario: "" Contraseña: "gestor"	Usuario no puede estar vacío."	
El campo Contraseña está vacío. Usuario: "gestor" Contraseña: ""	Mensaje de error que especifica "El campo Contraseña no puede estar vacío."	Satisfactoria
Usuario y contraseña válidas para el ECM Alfresco pero no válidos para la interfaz Web. Usuario: "admin" Contraseña: "admincip"	Mensaje de error que especifica "El usuario no está permitido en la interfaz Web."	Satisfactoria

Tabla 24. CP - Autenticar usuario

Caso de Prueba		
Código Caso de Prueba: EP-202-2	Nombre Historia de Usuario: Adicionar imagen.	
Nombre de la persona que realiza la prueba: Isabel María García Hilarión		
Descripción de la Prueba: Funcional		
Condiciones de ejecución:		
<ul style="list-style-type: none"> ✓ El usuario debe estar autenticado en el sistema. ✓ El usuario debe estar autenticado con el rol "fotógrafo" o "gestor". 		
Pasos: Presionar la opción adicionar imagen, presionar el botón examinar de la ventana emergente, elegir la imagen en un explorador de archivos y oprimir el botón "aceptar".		
Escenarios	Resultados Esperados	Evaluación
Selección de una imagen con nombre y formato válido en el sistema. Nombre: "Imagen1.jpg"	Adición de la imagen al espacio personal del usuario autenticado.	Satisfactoria
Formato de la imagen diferente de "jpg, gif,	Mensaje de error que especifica "La extensión de	Satisfactoria

png, tiff". Nombre: "Imagen1.psd"	la imagen debe ser .jpg, .gif, .png o .tiff."	
Nombre de la imagen existente en el sistema. Nombre: "ImagenExistente.jpg"	Mensaje de error que especifica "La imagen ya existe en el sistema."	Satisfactoria
Campos con caracteres no válidos. Nombre: "Imagen 1.jpg"	Mensaje de error que especifica "El nombre de la imagen contiene caracteres no válidos."	Satisfactoria
Imagen con tamaño superior a los 4MB.	Mensaje de error que especifica "La imagen es demasiado grande."	Satisfactoria
Nombre de la imagen con menos de 3 caracteres: Nombre: "A.jpg"	Mensaje de error que especifica "El nombre debe contener entre 3 y 30 caracteres."	Satisfactoria
Nombre de la imagen con más de 30 caracteres: Nombre: "jiiilijftjlilifijtiljjfttilfff.jpg"	Mensaje de error que especifica "El nombre debe contener entre 3 y 30 caracteres."	Satisfactoria
Campo nombre de la imagen vacío: Nombre: ""	Mensaje de error que especifica "Primeramente debe seleccionar una imagen."	Satisfactoria

Tabla 25. CP - Adicionar imagen

Caso de Prueba	
Código Caso de Prueba: EP-204-4	Nombre Historia de Usuario: Eliminar imagen.
Nombre de la persona que realiza la prueba: Isabel María García Hilarión	
Descripción de la Prueba: Funcional	
Condiciones de ejecución:	
<ul style="list-style-type: none"> ✓ El usuario debe estar autenticado en el sistema con el rol de fotógrafo o gestor. 	

Pasos: Realizar la búsqueda de la carpeta en donde se encuentra las imágenes que desea eliminar/ Seleccionar la carpeta/ Luego se despliegan las imágenes contenidas dentro de esa carpeta/ el usuario debe seleccionar la o las imágenes a eliminar/ Presionar el ícono “Eliminar imagen”.		
Escenarios	Resultados Esperados	Evaluación
Se oprime el botón eliminar y no se seleccionan imágenes.	Mostrar mensaje que especifica “No se ha seleccionado ninguna imagen.”	Satisfactoria
Se seleccionan las imágenes y se oprime el botón “eliminar”.	El sistema muestra un mensaje de confirmación, si el usuario acepta se eliminan las imágenes seleccionadas y se recarga el bloque “Mis imágenes” del espacio personal del usuario. En caso contrario, el sistema vuelve a la página inicial del usuario y no se realiza ninguna operación.	Satisfactoria

Tabla 26. CP - Eliminar imagen

3.3.1. Resultados de la pruebas funcionales

Para la validación de los requisitos funcionales se realizaron tres iteraciones de pruebas. A continuación se muestran los resultados obtenidos en cada una de las iteraciones realizadas a la interfaz Web Ecumene Pyxel (ver Tabla 27).

No conformidades	Primera iteración	Segunda iteración	Tercera iteración
Detectadas	45	25	8
Resueltas	25	20	8
Pendientes	20	5	0

Tabla 27. Comportamiento de las no conformidades por iteraciones

En el gráfico 1 se puede observar el comportamiento de las no conformidades por cada una de las iteraciones realizadas (ver Gráfico 1).

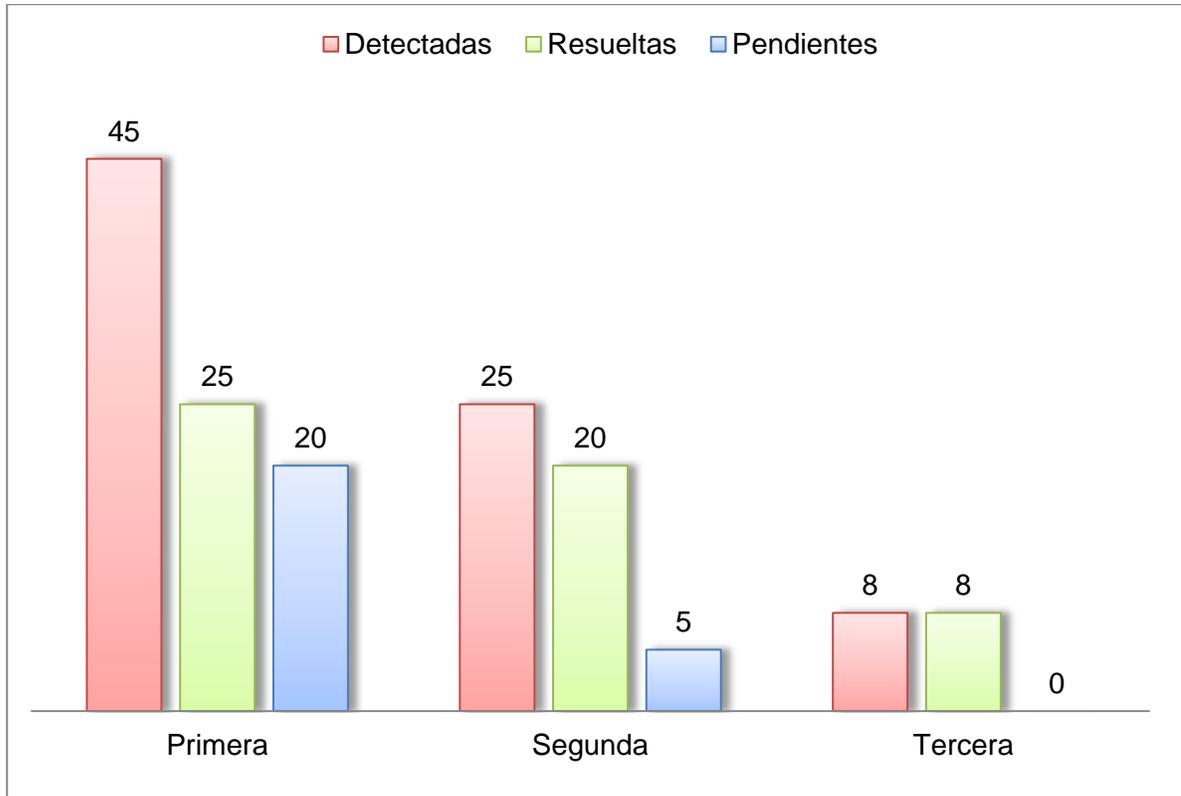


Gráfico 1. Comportamiento de las no conformidades en las pruebas funcionales

Algunas de las no conformidades detectadas en las pruebas funcionales realizadas se presentan a continuación (ver Tabla 28).

Iteración	No conformidades
Primera	No elimina carpeta en el espacio personal del fotógrafo.
Primera	No se muestran los mensajes de error en la autenticación al sistema.
Primera	El editor no puede adicionar asociación a una galería temática.
Segunda	No se muestra la página acceso denegado cuando los usuarios no tienen permiso sobre una acción específica.

Segunda	No realiza correctamente las validaciones de los campos de los formularios.
Tercera	Los mensajes de las validaciones no se corresponden con los errores identificados.
Tercera	Los mensajes de error se muestran durante muy corto tiempo, esto dificulta su lectura por los usuarios.

Tabla 28. Principales no conformidades detectadas

4. Conclusiones

Terminado el desarrollo de la Interfaz Web Ecumene Pyxel, se arriba a las siguientes conclusiones:

- ✓ Con la fundamentación teórica realizada que sustenta la investigación, se identificaron sistemas con un alto nivel de aceptación, que aunque no satisfacen las expectativas del proyecto completamente para realizar un uso correcto de la gestión documental, le facilitaron al presente trabajo adquirir nuevos elementos que tributan al mejoramiento del sistema.
- ✓ Con la obtención de los esquemas y los artefactos del sistema, se generó una propuesta de solución que permitió obtener una visión del adecuado funcionamiento del sistema que guió el desarrollo de la aplicación.
- ✓ La interfaz Web Ecumene Pyxel desarrollada proporciona a los usuarios la posibilidad de consultar y buscar los recursos fotográficos, de forma rápida y sencilla, contribuyendo de esta manera a mejorar el trabajo de los profesionales del sector periodístico, artístico, investigativo y otras instituciones.
- ✓ Las evaluaciones internas realizadas por el proyecto arrojaron resultados satisfactorios, el 100% de las no conformidades detectadas durante las iteraciones definidas fueron resueltas, esto permitió validar el adecuado funcionamiento de la interfaz Web para el Sistema de Gestión Documental de Imágenes Digitales Ecumene Pyxel.

5. Recomendaciones

Una vez concluida la investigación y el desarrollo del sistema se recomienda:

- ✓ Incluirle nuevas funcionalidades a la interfaz Web Ecumene Pyxel, para facilitarle la gestión de las imágenes digitales a los usuarios, tales como:
 - Permitir la edición en línea de las imágenes digitales.
 - Mover contenido mediante el uso de la técnica arrastrar y soltar.
 - Permitir mover unidades documentales.
- ✓ Implementar un módulo de administración que permita la gestión de las configuraciones del sistema.

6. Bibliografía

- Alfresco. 2011.** Opsoro. *alfrescowiki*. [En línea] Alfresco, 2011. [Citado el: 12 de 02 de 2012.] <http://wiki.alfresco.com/wiki/Opsoro>.
- Biblioteca de la Universidad de Cornell. 2003.** Llevando la teoría a la Práctica. *Tutorial de digitalización de imágenes*. [En línea] 28 de 04 de 2003. [Citado el: 10 de 03 de 2012.] <http://www.library.cornell.edu/preservation/tutorial-spanish/contents.html>.
- Blanco, Dayelis y Santana, Yoani. 2012.** eXcriba como solución para la gestión documental. [En línea] 2012. [Citado el: 02 de 05 de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3875.
- Blanco, Jaime y Bernaus, Albert. 2006.** *Aprender a crear páginas Web*. 2006.
- Cabrera Díaz, Dayani. 2006.** *Propuesta de lineamientos para el tratamiento documental de las fotografías de prensa en los medios cubanos*. La Habana : s.n., 2006.
- Calidad y software. 2009.** Calidad y software. *Pruebas funcionales*. [En línea] 2009. [Citado el: 23 de 03 de 2012.] http://www.calidadyssoftware.com/testing/pruebas_funcionales.php.
- Cei, Ugo y Lucidi, Piergiorgio. 2010.** *Alfresco 3 Web Services*. Olton : Packt Publishing Ltd, 2010. 1110810.
- de la Fuente, Toni. 2008.** *Alfresco: gestión documental y gestión de contenidos Web*. 2008.
- Definición ABC. 2011.** Conceptos en Definición ABC. *Definición de Imágenes*. [En línea] 2011. [Citado el: 05 de 02 de 2012.] <http://www.definicionabc.com/comunicacion/imagenes.php>.
- Eguíluz, Javier. 2009.** librosweb. *Introducción a JavaScript*. [En línea] 25 de 03 de 2009. [Citado el: 01 de 12 de 2011.] <http://www.librosweb.es/javascript>.
- Elejalde, Reinier. 2009.** Módulo para la creación de modelos de contenido para Alfresco. [En línea] 07 de 2009. [Citado el: 26 de 02 de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_2385_09.
- Facundo, Héctor. 2006.** *Sitios Web bajo Linux*. Buenos Aires : MP Ediciones, 2006. ISBN 987-526-074-6.
- Fernández, Manuel. 2009.** Docasu, un interfaz para Alfresco. [En línea] Optaros, 24 de 07 de 2009. [Citado el: 25 de 11 de 2011.] <http://mfernandez.es/wordpress/2009/07/24/docasu-un-interfaz->

para-alfresco-por-optaros/.

- Flores, Katherine. 2010.** Conceptos Básicos Relacionados al Internet. *Internet y sus avances*. [En línea] 09 de 2010. [Citado el: 20 de Febrero de 2012.] <http://katherineflores.com/2010/09/00002-conceptos-basicos-relacionados-al-Internet.html>.
- Folmer, E. y Bosch, J. 2004.** Architecting for usability: a survey. *Journal of Systems and Software*. 2004. Vol. 70, 1-2.
- Fonseca, Misael. 2012.** eXcriba, gestor de documentos administrativos. [En línea] 2012. [Citado el: 02 de 05 de 12.] http://repositorio_institucional.uci.cu/jspui/handle/ident/4116.
- Fretzen, Jeff y Sobotka, Henry. 2004.** *Superutilidades para JavaScript*. Madrid : McGRAW-HILLINTERAMERICAN, 2004. ISBN: 0-07-882364-1.
- Fugueras, Alberch. 2003.** *Los archivos entre la memoria histórica y la sociedad del conocimiento*. Barcelona : UOC, 2003.
- Gallús, C. 2009.** PHP orientado a objeto. *Web estilo*. [En línea] 10 de 03 de 2009. [Citado el: 20 de 01 de 2012.] <http://www.webestilo.com/php/articulo.phtml?art=27>.
- GEITEL. 2011.** *Estudio de factibilidad Tecnológica para la Gestión Documental de Imágenes Digitales*. La Habana : s.n., 2011.
- González, Osvany y Jorge, Salomón. 2009.** *Implementación de la Interfaz Web para el Gestor de Contenido Empresarial Alfresco*. La Habana : s.n., 2009.
- GRANMA. 2006.** Archivo de Gestión Administrativos. *Archivo de Gestión Administrativos*. [En línea] Granma, 2006. [Citado el: 12 de 09 de 2011.] <http://www.archivo.granma.inf.cu/archivo/index.html>.
- Hassan, Yusef. 2002.** Introducción a la Usabilidad. [En línea] 01 de 11 de 2002. [Citado el: 02 de 11 de 2011.] http://www.nosolousabilidad.com/introduccion_usabilidad.html. 1886-8592.
- Hassan, Yusef y Martín, Francisco. 2003.** Qué es la Accesibilidad Web. [En línea] 14 de 07 de 2003. [Citado el: 15 de 05 de 2012.] <http://www.nosolousabilidad.com/accesibilidad.html>. 1886-8592.
- Henry, Shawn. 2002.** Understanding Web Accessibility. En *Constructing Accessible Web Sites*. [En línea] 04 de 2002. [Citado el: 02 de 02 de 2012.] http://www.macromedia.com/macromedia/accessibility/pub/acc_sites_chap01.pdf.

1904151000.

Hualpara, Marca y Limachi, Quisbert. 2009. *Diagrama de Despliegue*. 2009.

iFresco. 2011. iFresco. *iFresco*. [En línea] iFresco, 2011. [Citado el: 02 de diciembre de 2011.] <http://www.ifresco.at/de/products/client/overview.html>.

Incomparable. 2011. Arquitecturas n-tier. *Ventajas y Desventajas*. [En línea] 2011. [Citado el: 10 de 02 de 2012.] <http://incomparable.blogspot.com/2008/11/arquitecturas-n-tier-ventajas-y.html>.

Introducción a Symfony 2 - Jornada Symfony 2011. **Eguiluz, Javier. 2011.** España : s.n., 2011.

ISO 9241-11. 1998. *Ergonomic requirements for office work with visual display terminals - Part 11 Guidance on usability*. 1998.

José, Lozano. 2008. La Web 2.0. *Revista Avances*. [En línea] 2008. [Citado el: 30 de noviembre de 2011.] http://adide.org/revista/index.php?option=com_content&task=view&id=255&Itemid=62.

Marrero, Liliam. 2009. Gestión documental de la información y del conocimiento en los escenarios ciberperiodísticos. [En línea] 2009. [Citado el: 20 de 03 de 2012.] <http://dialnet.unirioja.es/servlet/articulo?codigo=3168529>.

Moreno, Luciano. 2005. Componentes de una Interfaz Web. *Interfaces. Desarrollo Web*. [En línea] 07 de 2005. [Citado el: 17 de 11 de 2011.] <http://www.desarrolloWeb.com/articulos/2171.php>.

Mugica, Mena. 2005. *Gestión documental y organización de archivos*. La Habana : Félix Varela, 2005.

Nielsen, Jakob. 2001. Beyond Accessibility: Treating People with Disabilities as People. [En línea] Alertbox, 11 de 11 de 2001. [Citado el: 02 de 11 de 2011.] <http://www.useit.com/alertbox/20011111.html>.

Opsoro. 2010. Opsoro. *Opsoro*. [En línea] 2010. [Citado el: 06 de enero de 2012.] <http://www.opsoro.org/home>.

O'Reilly, Tim. 2011. Qué es web 2.0. *Patrones del diseño y modelos del negocio para la siguiente generación del software*. [En línea] 2011. [Citado el: 15 de 02 de 2012.] <http://sociedaddelainformacion.telefonica.es/>.

Pacheco, Nacho. 2011. *Symfony2-es - Release 2.0.6*. Madrid : s.n., 2011.

Ponjuan, Dante. 2005. Gestión documental, gestión de información y gestión del conocimiento: evolución y sinergias. [En línea] 2005. [Citado el: 16 de noviembre de 2011.]

<http://www.cinfo.cu/Userfiles/file/Cinfo/CINFO/diciembre2005/Comunicación.doc>.

Real Academia de la Lengua Española. 2011. Diccionario Usual. *Imagen*. [En línea] Real Academia Española, 02 de 12 de 2011. [Citado el: 15 de 05 de 2012.] <http://buscon.rae.es/drael/SrvltGUIBusUsual?LEMA=imagen>.

Redondo Duque, Miguel y Bravo Santos, Crescencio. 2006. *Sistemas interactivos y colaborativos en la Web*. 2006.

Romero, Gladys. 2008. *Metodología ágil para proyectos de software libre*. La Habana : UCI, 2008.

Rotz, B. 2008. DoCASU. *DoCASU*. [En línea] EOSDirectory, 2008. [Citado el: 05 de 12 de 2011.] <http://www.eosdirectory.com/blogs/2008/07/15/>.

Saco, Lisandra y Ibarria, Juviel. 2011. *Interfaz Web del Sistema de Gestión Documental de Fotografías para el Centro de Información de la Prensa*. La Habana : s.n., 2011.

Serrano, Juan. 2008. Conferencia Arquitectura del software. *Parte I. Introducción, Universidad Rey Juan Carlos*. [En línea] 2008. [Citado el: 21 de 10 de 2011.] <http://zenon.etsii.urjc.es/grupo/docencia/as>.

Shariff, Munwar. 2010. *Alfresco Enterprise Content Management Implementation*. s.l. : PACKT, 2010.

Tore, Horacio Aldo. 2010. Patrones de diseño de software. *Patrones de diseño de software*. [En línea] 2010. [Citado el: 02 de 03 de 2012.] <http://www.dipneo.com.ar>.

Tupe, Cirley y Cisneros, Juan. 2008. *Evaluación y Selección de Framework de Desarrollo PHP*. 2008.

Van Der Henst, Christian. 2010. ¿Qué es la Web 2.0? *Maestros de la Web*. [En línea] 2010. [Citado el: 30 de noviembre de 2011.] <http://www.maestrosdelWeb.com/editorial/Web2/>.

7. Anexos

7.1. Anexo 1. Diagramas de procesos

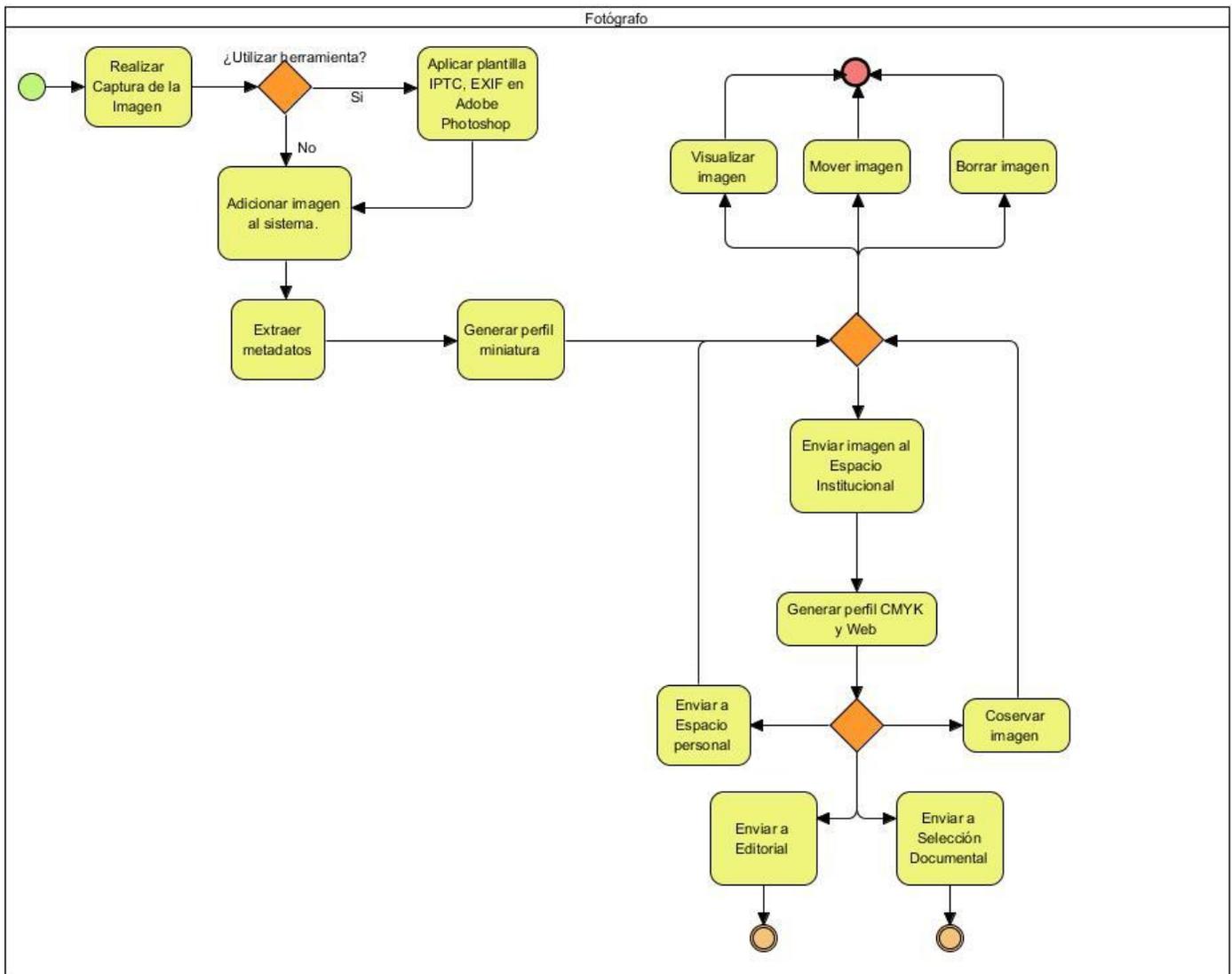


Figura 10. Proceso identificación

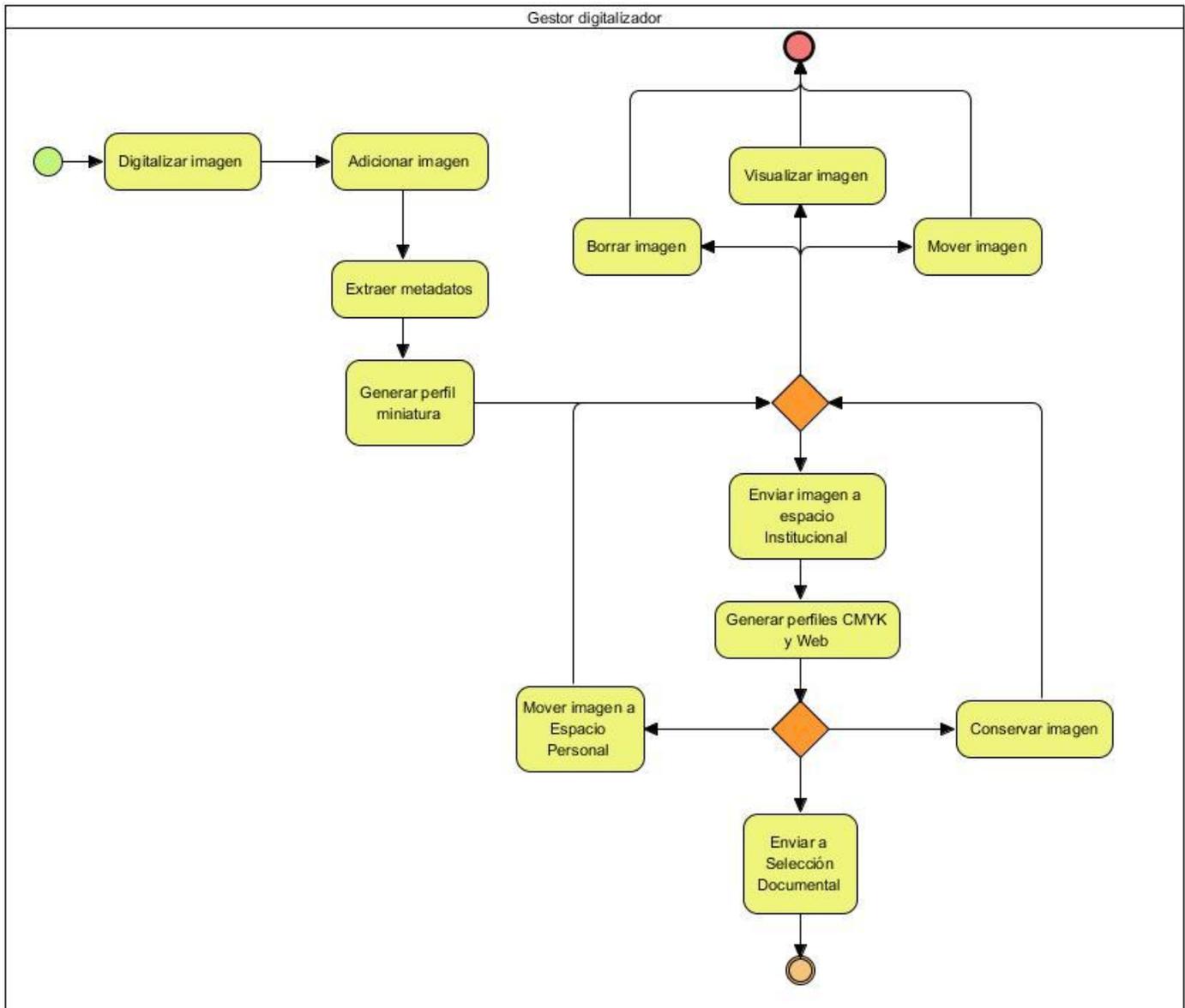


Figura 11. Proceso digitalización

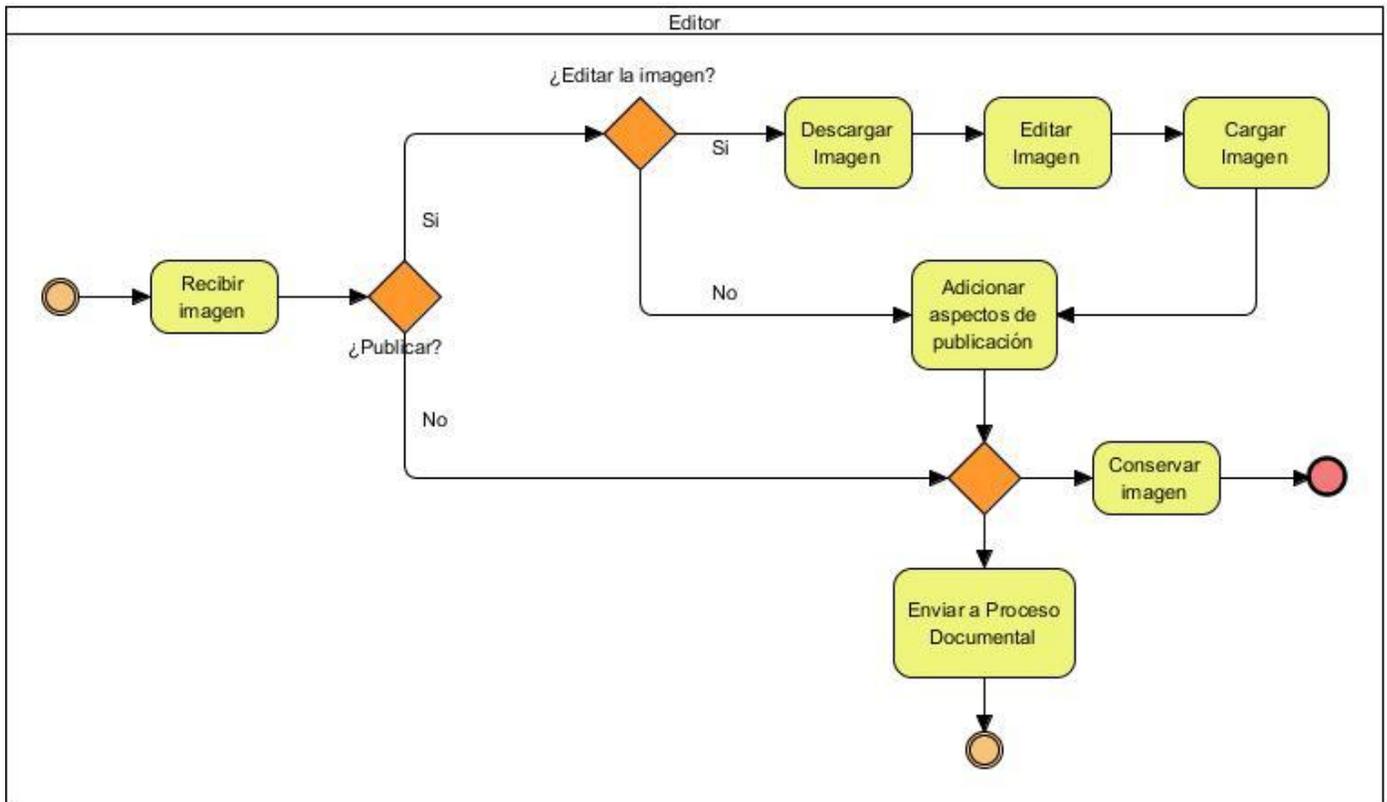


Figura 12. Proceso editorial

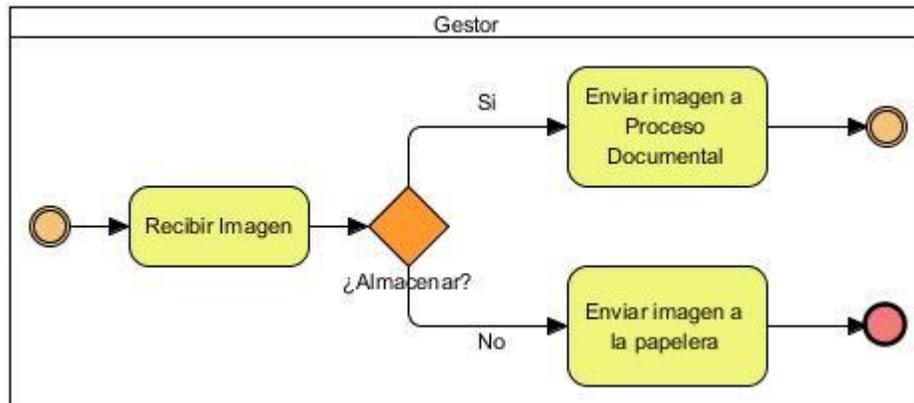


Figura 13. Selección documental

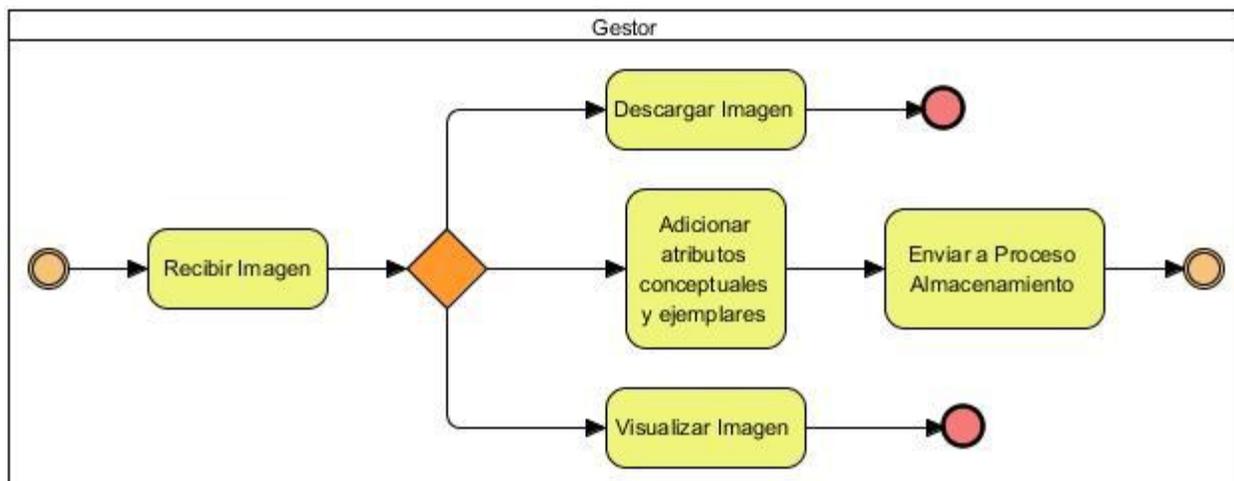


Figura 14. Proceso documental

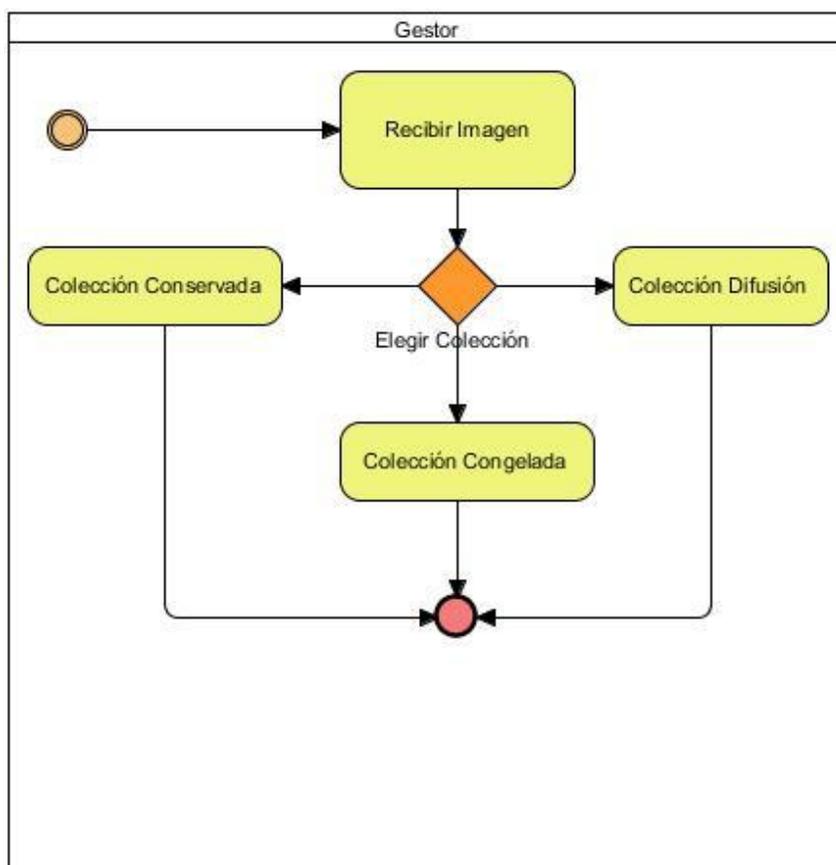


Figura 15. Proceso almacenamiento

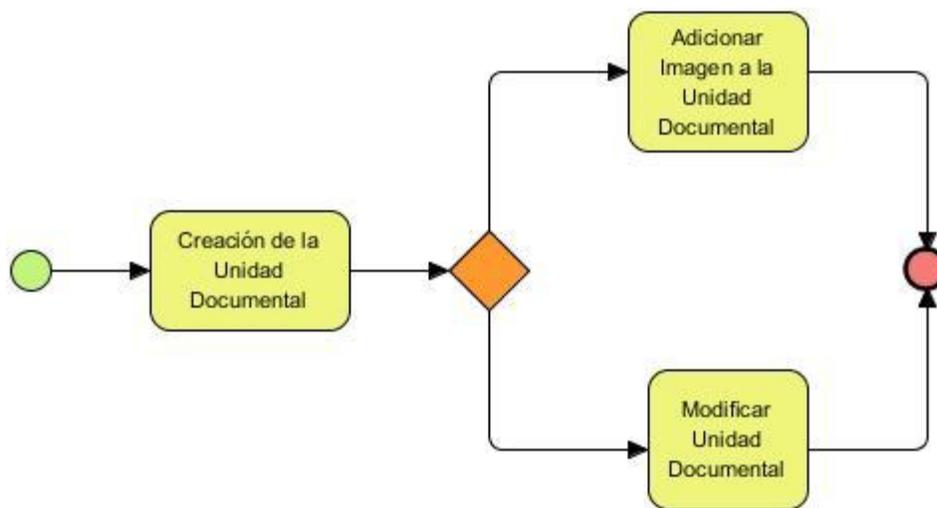


Figura 16. Ciclo de vida de la unidad documental

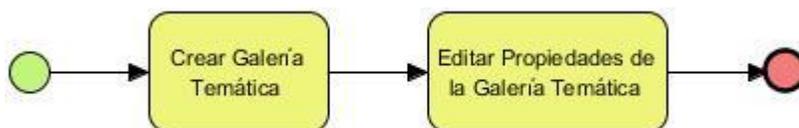


Figura 17. Ciclo de vida de la galería temática

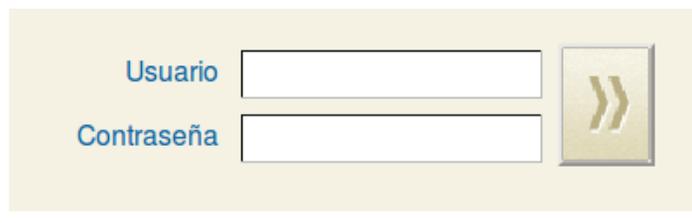
7.2. Anexo 2. Lista de riesgos

N ^a	Riesgo	Tipos de riesgos	Impacto	Descripción	Probabilidad	Efectos
1	Actividades políticas que surgen a última hora (marchas, tribunas).	Organización	Atraso en las actividades planificadas del proyecto.	Salida inmediata del lugar donde trabaja el proyecto por una jornada de trabajo completa o más.	Baja	Tolerables
2	Problemas familiares.	Personal	Retraso en las actividades que desarrolla esa persona en el proyecto.	No se cuenta con el personal calificado para realizar una determinada tarea por encontrarse fuera del proyecto por un tiempo indeterminado.	Media	Tolerables
3	Falta de preparación o conocimiento de las herramientas de trabajo.	Personal	Atraso en la fase lo cual provoca menos productividad.	Los miembros del equipo no presentan dominio de las herramientas.	Media	Tolerables
4	Estimación incorrecta de tiempo y recursos.	Estimación	Atraso en la entrega del producto final.	Por mal cálculo o atraso en el desarrollo, el equipo no tiene el producto final en el tiempo requerido.	Media	Serios
5	Ausencia de personal.	Personal	Retraso en la entrega del producto.	Ausencia de un programador o analista.	Media	Serios
6	Pérdida de Información.	Organización	Desde pequeños retrasos del trabajo hasta una pérdida total del trabajo realizado.	Pérdida de información por rotura de las máquinas o descuido de los integrantes del proyecto.	Media	Catastrófico
7	Rotura de Computadora.	Tecnológico	Retraso del trabajo lo cual incide negativamente en la terminación del producto.	Rotura de una máquina donde se encuentra información importante.	Baja	Serios

8	Falta de fluido eléctrico.	Tecnológico	Desagrado por parte del equipo, se extiende el período de culminación.	Afectación del fluido eléctrico en el laboratorio de producción.	Media	Tolerables
9	Falta de acceso a Internet.	Tecnológico	Tiempo de trabajo.	Caída del servicio de Internet o terminación de las cuotas asignadas a los integrantes del proyecto.	Media	Tolerables
10	Inconformidad de la Arquitectura.	Requerimientos	Dudas en la calidad del trabajo, modificar el producto y volver a fases anteriores.	Inconformidad en la arquitectura y diseño del producto.	Media	Tolerables
11	No se pactan las tareas y requerimientos con los clientes.	Organización	Trabajar sin objetivos claros y no responder a los requerimientos del cliente.	No se acuerdan con claridad las tareas con los clientes ni se definen por escrito los requerimientos.	Alta	Catastrófico
12	El cliente no tiene bien descritos sus procesos.	Requerimientos	Desarrollo.	El cliente no tiene bien descritos sus procesos. Dificultad a la hora de informatizarlos.	Alta	Serios
13	Incumplimiento de las tareas del proyecto.	Organización	Atraso en la entrega de las liberaciones.	Los estudiantes no cumplen con las tareas en tiempo establecido.	Alta	Serios
14	Insatisfacción del cliente.	Organización	Pérdida de reputación.	Insatisfacción de los clientes por baja calidad.	Media	Catastróficos

Tabla 29. Lista de riesgos

7.3. Anexo 3. Prototipos no funcionales



Este prototipo muestra un formulario de autenticación con un fondo beige. Incluye dos campos de entrada de texto: 'Usuario' y 'Contraseña', ambos con un borde gris. A la derecha de los campos hay un botón con un icono de flechas dobles (>>) que apunta a la derecha.

Figura 18. Prototipo autenticar usuario



Este prototipo muestra una ventana de diálogo con el título 'Adicionar Imagen' y un icono de cerrar (x) en la esquina superior derecha. Dentro de la ventana hay un campo de entrada de texto y un botón 'Examinar...' con un borde rojo. En la parte inferior de la ventana hay dos botones: 'Adicionar' y 'Cancelar', y un icono de cerrar (//) en la esquina inferior derecha.

Figura 19. Prototipo adicionar imagen

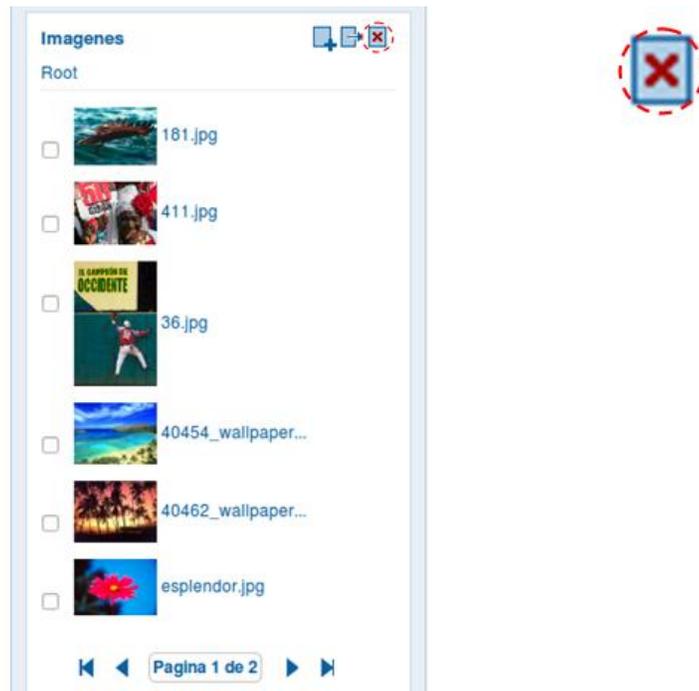


Figura 20. Prototipo eliminar imagen

7.4. Anexo 4. Diagramas de clases

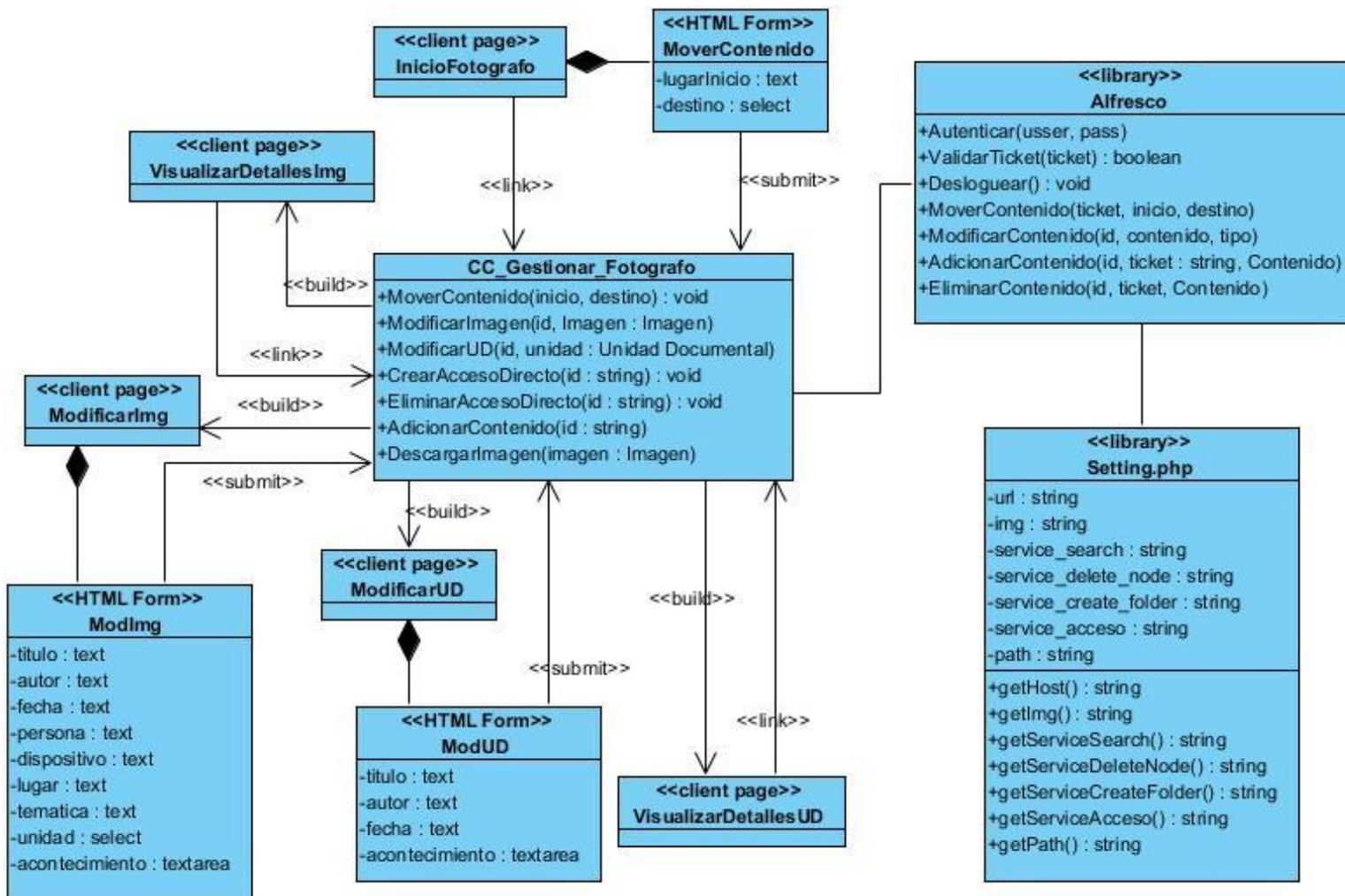


Figura 21. Diagrama de clases Fotógrafo

7.5. Anexo 5. Diagramas de componentes

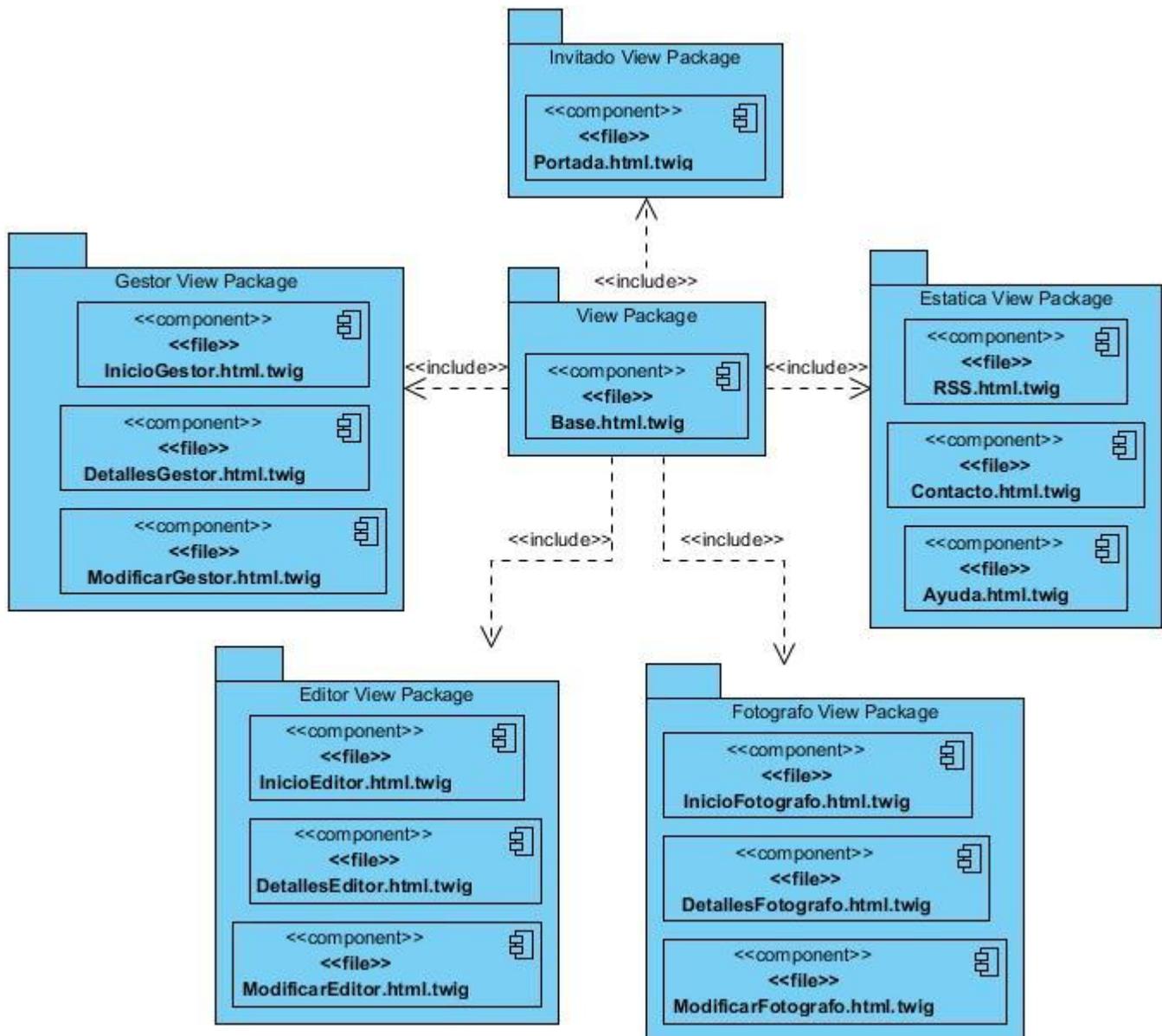


Figura 22. Diagrama de componentes Vistas

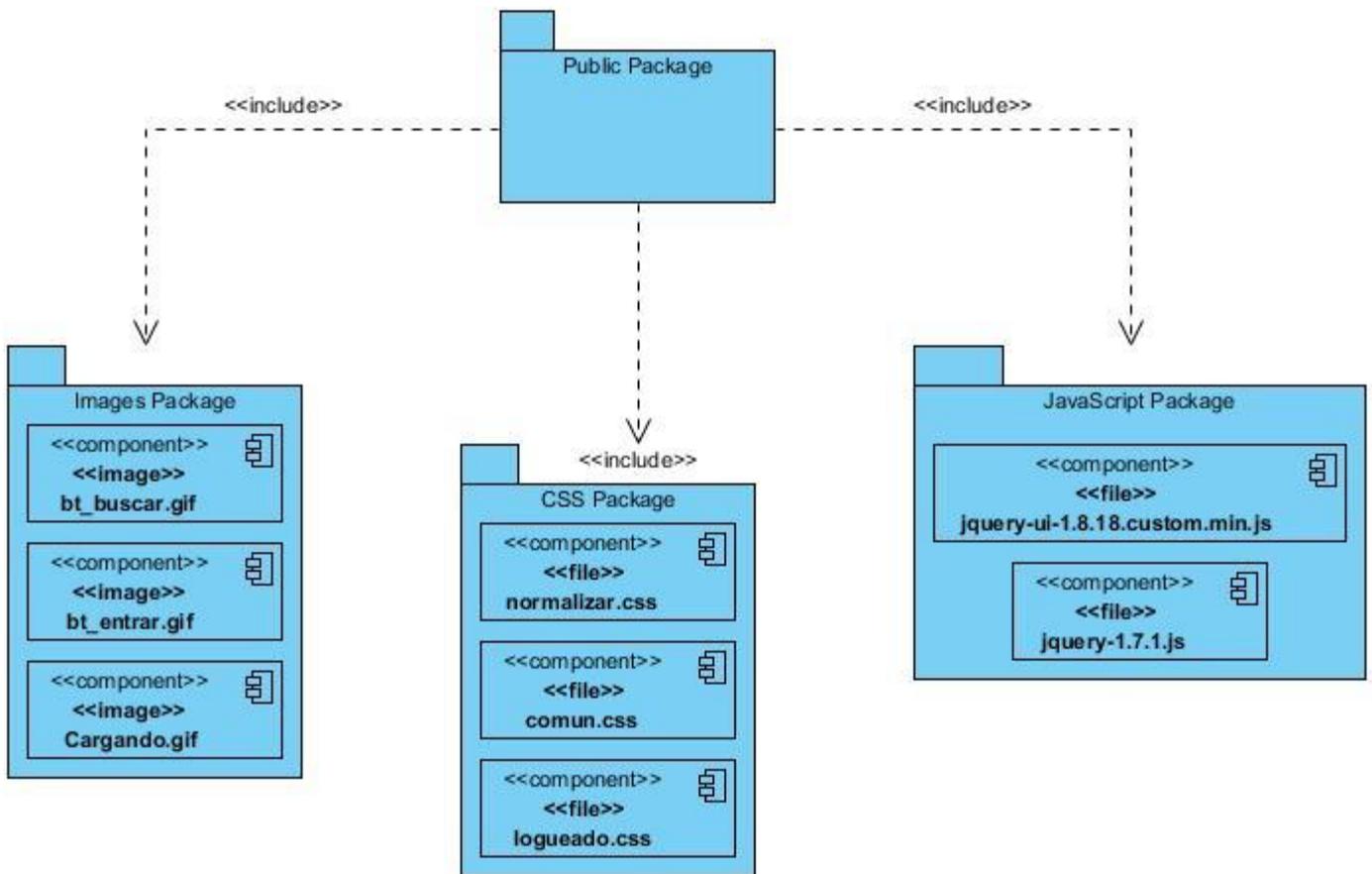


Figura 23. Diagrama de componentes Público

8. Glosario de términos

Término	Definición
API	(Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta librería para ser utilizado por otro software como una capa de abstracción.
CMYK	Se basa en la cualidad de absorber y rechazar luz de los objetos. Si un objeto es rojo esto significa que el mismo absorbe todas las componentes de la luz a excepción de la componente roja.
ECM	Gestor de Contenido Empresarial, el cual identifica a los sistemas informáticos que manejan la captura, almacenamiento, seguridad, control de versiones, recuperación, distribución, conservación y destrucción de documentos y contenido a nivel empresarial.
ECM Alfresco	Es la alternativa de Código Abierto para la gestión de contenido empresarial (ECM), este proporciona gestión documental, colaboración, gestión de registros, gestión de información, gestión del contenido web e imágenes.
Framework	En el desarrollo de software es una estructura conceptual y tecnológica de soporte definida, normalmente, con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado.
GUI	Interfaz Gráfica de Usuario, es un paradigma visual que permite al usuario comunicarse con un programa de forma intuitiva.
HU	Secuencias de acciones que el sistema puede llevar a cabo para interactuar con sus actores, incluyendo alternativas dentro de las secuencias.
Interfaz	La capa de presentación del sistema.
Internet	Sistema global de interconexión de redes informáticas, para servir a miles de millones de usuarios en todo el mundo.
Interoperabilidad	La condición mediante la cual sistemas heterogéneos pueden intercambiar procesos o

	datos.
ISO	Organización Internacional de Normalización. Proporciona la normalización y las cuestiones conexas a todos los interesados en alguna información.
PHP	Lenguaje de código abierto muy popular especialmente adecuado para el desarrollo Web y que puede ser incrustado en HTML.
Script	Secuencia de comandos de comportamientos.
Symfony 2	Es un Framework para programar aplicaciones Web en PHP.
UML	Lenguaje Unificado de Modelado por sus siglas en inglés <i>Unified Modeling Language</i> es un conjunto de herramientas, que permite modelar (analizar y diseñar) sistemas orientados a objetos.
URL	Localizador de Recurso Uniforme por sus siglas en inglés <i>Uniform Resource Locator</i> , la dirección global de documentos y de otros recursos en la Web.
W3C	<i>World Wide Web Consortium</i> abreviado W3C, es un consorcio internacional que produce recomendaciones para la World Wide Web.
XHTML	Lenguaje extensible de marcado de hipertexto, por su acrónimo en inglés <i>eXtensible Hypertext Markup Language</i> , es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas Web.

Tabla 30. Glosario de términos