

Universidad de las Ciencias Informáticas



Título: Metabuscador para los Sistemas de Gestión Documental de Imágenes Digitales Ecumene Pyxel V1.0

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

Autores:

Maridel Elena Almira Quiroga

Wilbert Jesús Galano Batista

Tutores:

Ing. Gilberto Lissabet Hernández

Ing. Yuneisy Barrios Pérez

La Habana, Junio 2013.

Declaramos ser los autores legítimos del trabajo titulado *Metabuscadore para los Sistemas de Gestión Documental de Imágenes Digitales Ecumene Pyxel V1.0* y delegamos al centro de Ideo-Infornática y a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo el presente a los ____ días del mes de ____ del año 2013.

Maridel Elena Almira Quiroga
Firma del autor

Wilbert Jesús Galano Batista
Firma del autor

Gilberto Lissabet Hernández
Firma del tutor

Yuneisy Barrios Pérez
Firma del tutor

Elena:

A las personas más importantes de mi vida.

A mis padres, por el simple hecho de existir, de estar a mi lado en cada paso de mi vida aunque sea en la distancia, por hacer de mí una mejor persona con su cariño y ejemplo, por encaminarme en la vida y darme la educación necesaria para mantenerme en ella.

A mi hermano, por hacerme feliz con tan solo su presencia, por ser un niño tan noble y desprender tanto amor y ternura en todos los que lo rodean. Por hacer de mí un ejemplo y defender mí persona ante todos.

A papi Quiroga, aunque no se encuentre entre nosotros, por ser una de las personas más relevantes en mi vida. Por creer en mí y haber idealizado desde pequeña este momento, por ser más que un abuelo, un padre, por su figura intachable, por saber educar con su palabra y por estar presente aun cuando no es posible.

A mami Nivia, por ser la abuela más linda del mundo, por estar siempre a mi lado desde que vine al mundo y mantenerse siempre tan atenta a mis necesidades. Por sus cuidados, consejos, por ser la más pequeña y a la vez más adulta de la familia.

A mis abuelos paternos, por ser tan gentiles y traer al mundo a un hombre tan honrado como mi padre.

A toda la familia en general, por apoyarme todos estos años.

A mi novio Howard, por ser tan lindo conmigo, por estar presente en los buenos momentos y en los malos. Por demostrarme que las personas pueden ser mejores si se lo proponen. Por brindarme tanto amor y cariño y hacerme reír cuando solo hay tristeza en mi corazón.

A mis tutores por su gran ayuda y apoyo durante todo el proceso de realización de este trabajo de diploma.

A mi compañero de tesis y de grupo durante 5 años, por su gran trabajo y esfuerzo para que el producto final fuese de gran calidad.

Wilbert:

A mis padres por su infinito amor, comprensión y apoyo incondicional en todas las decisiones y acciones que he tomado en mi vida.

A Noemí mi mamá por ser mi amiga, mi confidente, mi hermana en todo momento, jugando el papel de puntal para mi vida, siendo la persona a la que más amo en el mundo y por siempre hacer lo posible y lo imposible para que nunca me falte nada y esté siempre bien.

A Adrián mi papá por ser la figura de más respeto y amor que he tenido en mi vida, por confiar en mí y desear hasta el último aliento que su hijo fuese siempre un hombre de bien.

A mis hermanos Yamilet y Walmer por ser un ejemplo a seguir y nunca dudar de mí.

A mis tías Mima y Belkita, porque son como si fueran mis madres y estar siempre al tanto de mí.

A mis tíos Socorrito y Raúl porque son como si fueran mis padres y por darme la oportunidad de contar siempre con ellos.

A mí querida novia Edelsys (Vivi) por demostrarme que cuando hay amor se puede lograr todo en la vida, por cuidarme y quererme con tanto esmero y dedicación.

A toda mi familia en general, por apoyarme todos estos años.

A mis tutores y amigos por su gran ayuda y apoyo durante todo el proceso de realización de este trabajo de diploma y en mi vida universitaria.

A mi compañera de tesis y de grupo, por su gran trabajo y esfuerzo para que este sueño fuese realidad.

Elena:

Dedico este trabajo a mis padres, por darme la vida.

A mi hermano, por su amor y cariño.

Wilbert:

Dedico este trabajo a mis padres, por ser quienes merecen la realización de este sueño.

A mi mamá, por ser mi razón de ser y estar siempre a mi lado cuidándome.

A mi papá, por ser el motor impulsor de mi carrera, por confiar y desear hasta el último momento que este sueño se cumpliera.

Resumen

El almacenamiento, localización y recuperación de la información siempre ha requerido del desarrollo de mecanismos de acceso, ya sea, para uso de los profesionales que organizan y gestionan la información, o uso de los usuarios finales. En el centro de Ideo-Informática de la Universidad de las Ciencias Informáticas es desarrollado el Programa de Informatización de la Prensa, el cual está encaminado a alcanzar un periodismo digital participativo y colaborativo a nivel nacional. Como parte del Programa se encuentra en desarrollo el sistema de gestión documental de imágenes digitales Ecumene Pyxel V1.0, el cual permite recuperar las imágenes que se encuentran almacenadas en sus variantes web, sin embargo no garantiza la búsqueda centralizada de imágenes digitales una vez desplegadas en las entidades periodísticas. El objetivo del presente trabajo de diploma es el desarrollo de un sistema de recuperación de la información, específicamente un metabuscador, para la centralización de los recursos fotográficos disponibles en los medios de prensa que utilicen como sistema de gestión documental a Ecumene Pyxel V1.0. Para dar cumplimiento al objetivo propuesto se llevó a cabo un estudio de los sistemas de recuperación de la información según sus características y particularidades, y se escogieron herramientas y tecnologías encaminadas al desarrollo de una aplicación web. Como resultado se obtuvo un metabuscador, basado en estándares, para la búsqueda centralizada en los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0, capaz de socializar y compartir los recursos fotográficos disponibles entre las diferentes entidades que utilicen como núcleo dicho sistema.

Palabras claves: Centralización, imágenes digitales, metabuscador.

Índice

Introducción	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	6
1.1 Introducción.....	6
1.2 Visión general del Programa de Informatización de la Prensa	6
1.3 Sistemas de Recuperación de Información.....	8
1.3.1 Buscadores.....	9
1.3.2 Multibuscadores.....	11
1.3.3 Metabuscadores	11
1.4 Sistemas homólogos.....	14
1.5 Tecnologías	17
1.5.1 ECM	17
1.5.2 Framework.....	17
1.5.3 Sistema Gestor de Base de Datos (SGBD).....	19
1.6 Lenguajes utilizados	19
1.7 Herramientas, técnicas y estilo de arquitectura.....	22
1.8 Metodologías	25
1.9 Conclusiones	27
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA. ANÁLISIS Y DISEÑO.....	28
2.1 Introducción.....	28
2.2 Características del sistema	28
2.3 Modelado del Dominio	29
2.4 Requerimientos de software	30
2.4.1 Requisitos funcionales.....	30
2.4.2 Requisitos no funcionales	33

2.5	Casos de uso del sistema.....	36
2.6	Estilos arquitectónicos y patrones de diseño	39
2.7	Diagramas de Clases del Diseño	41
2.8	Diagrama de Interacción.....	43
2.9	Modelo de Despliegue	43
2.10	Conclusiones	44
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA		45
3.1	Introducción	45
3.2	Diagrama de componentes.....	45
3.3	Código fuente	47
3.3.1	Estándar de codificación.....	47
3.4	Interfaces principales de la aplicación.....	48
3.5	Validación del sistema	50
3.5.1	Pruebas funcionales	50
3.5.2	Pruebas de seguridad.....	54
3.5.3	Pruebas de carga y estrés	54
3.5.4	Pruebas de integración	55
3.6	Conclusiones	57
Conclusiones		58
Recomendaciones		59
Bibliografía Referenciada.....		60
Glosario de términos.....		65
Anexos.....		69

Índice de figuras

Figura No. 1: Mapa de aplicaciones del Programa de Informatización de la Prensa.	8
Figura No. 2: Arquitectura de un buscador.....	10
Figura No. 3: Estructura modular de un metabuscador.	13
Figura No. 4: Diagrama de Clases del Modelo del Dominio.	30
Figura No. 5: Diagrama de Casos de Uso del sistema.	37
Figura No. 6: Diagrama de Clases del Diseño del caso de uso Realizar búsqueda simple.	42
Figura No. 7: Diagrama de Secuencia para el caso de uso Realizar búsqueda simple.	43
Figura No. 8: Diagrama de Despliegue del sistema.	44
Figura No. 9: Diagrama de Componentes del sistema.	46
Figura No. 10: Código PHP de la clase TaxoController.php.....	48
Figura No. 11: Interfaz web Inicio.....	49
Figura No. 12: Interfaz web de Inicio para la búsqueda avanzada.	49
Figura No. 13: Diagrama de Clases del Diseño del caso de uso Autenticar usuario.	98
Figura No. 14: Diagrama de Clases del Diseño del caso de uso Realizar búsqueda avanzada.	99
Figura No. 15: Diagrama de Clases del Diseño del caso de uso Gestionar usuario.	100
Figura No. 16: Diagrama de Clases del Diseño del caso de uso Gestionar proveedor.....	101
Figura No. 17: Diagrama de Clases del Diseño del caso de uso Gestionar taxonomía.	102
Figura No. 18: Diagrama de Clases del Diseño del caso de uso Compartir contenidos.	103
Figura No. 19: Diagrama de Secuencia del caso de uso Autenticar usuario.	104
Figura No. 20: Diagrama de Secuencia del caso de uso Compartir contenidos.	104
Figura No. 21: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Adicionar usuario...	105
Figura No. 22: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Editar usuario.	105
Figura No. 23: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Eliminar usuario.....	106
Figura No. 24: Interfaz web Registrarse.....	107
Figura No. 25: Interfaz web Resultados.	108
Figura No. 26: Interfaz web Compartir contenidos.	109
Figura No. 27: Interfaz web Gestionar.....	110

Índice de tablas

Tabla No. 1: Requisitos funcionales del sistema.	33
Tabla No. 2: Especificación del caso de uso Realizar búsqueda simple.	39
Tabla No. 3: Caso de prueba para el requisito Autenticar usuario.	51
Tabla No. 4: Caso de prueba para el requisito Realizar búsqueda simple.	52
Tabla No. 5: Resultado de prueba funcionales según las no conformidades.	52
Tabla No. 6: No conformidades detectadas.	53
Tabla No. 7: Caso de prueba para los proveedores del sistema.	56
Tabla No. 8: Caso de prueba para la base de datos del sistema.	56
Tabla No. 9: Especificación del caso de uso Autenticar usuario.	71
Tabla No. 10: Especificación del caso de uso Gestionar usuario.	77
Tabla No. 11: Especificación del caso de uso Solicitar cuenta de usuario.	80
Tabla No. 12: Especificación del caso de uso Gestionar proveedores.	86
Tabla No. 13: Especificación del caso de uso Chequear proveedor.	88
Tabla No. 14: Especificación del caso de uso Salvar criterio de búsqueda.	89
Tabla No. 15: Especificación del caso de uso Gestionar taxonomía.	94
Tabla No. 16: Especificación del caso de uso Realizar búsqueda avanzada.	97

Introducción

A raíz de la evolución de las Tecnologías de la Información y las Comunicaciones (TIC), diversas organizaciones y entidades a nivel nacional e internacional han tenido que orientar sus esfuerzos en la identificación de nuevas y novedosas maneras de gestionar y socializar la información y el conocimiento. Los Medios de Comunicación Masiva (MCM), los cuales tienen por objetivo dar a conocer el acontecer de manera rápida y con la mayor calidad, han alcanzado un mayor desarrollo a partir de dicha evolución. En gran parte, ellos contribuyen a fijar las maneras de pensamiento de la sociedad, a crear la reputación de una organización, persona o grupo de personas, proporcionando información y elementos para que los individuos construyan, ponderen y formen sus opiniones [1].

Una de las necesidades de los MCM es la creación de mecanismos de integración, que les permita la socialización de la información y el conocimiento entre las distintas entidades que participan en el mundo mediático. En la prensa como medio de comunicación, la evolución soportada en tecnología es evidente. Como respaldo a tal afirmación, Canga Larequi¹ reconoce que *"las denominadas nuevas tecnologías de la comunicación/información inciden en la actualidad de forma determinante sobre la prensa escrita y llegan a condicionar el futuro de este medio"* [2].

Hoy las nuevas TIC hacen mayores aportes a la prensa, llevándola a lo que se conoce como una nueva etapa, la prensa digital, la cual se ve como un nuevo medio y no como una derivación o un periódico convencional en un nuevo soporte. Con el desarrollo proporcionado por el nacimiento de la prensa digital la obtención de la información es de forma más rápida, directa y precisa.

En Cuba se ha seguido muy de cerca el desarrollo de los MCM, particularmente en la prensa. Si bien este medio posee un desarrollo constante como parte del desarrollo de las TIC y cuenta con ejemplares como Granma, Trabajadores, Juventud Rebelde y muchos otros, no posee una integración que le permita socializar la información que tiene cada medio de una manera más fácil, íntegra y confiable. Es entonces que para suplir ese déficit en Mayo de 2005 surge oficialmente en la UCI² el Programa de Informatización de la Prensa, como transformador de la forma de hacer prensa en Cuba y un exponente de un periodismo

¹Jesús Canga Larequi Doctor en Ciencias de la Información. Catedrático de Tecnología de los Medios Impresos de la Universidad del País Vasco (UPV).

²Universidad de las Ciencias Informáticas.

digital que supone la integración de todos los medios de prensa cubanos en una red de información donde prime la colaboración y no la competencia [3].

Uno de los objetivos con mayor impacto en el Programa de Informatización de la Prensa es precisamente el lograr un periodismo digital participativo y colaborativo. En este sentido la fototeca de los diferentes medios de prensa juega un papel fundamental en la gestión y socialización de los recursos fotográficos, que para los MCM tiene un valor patrimonial e investigativo.

En las condiciones actuales dicha colaboración no es efectiva y en gran medida se debe a que:

- Se desarrollan diferentes aplicaciones para resolver un mismo problema, en los MCM no se comparten los recursos, no se socializa la información, ni la aplicación.
- No se respetan estándares en la representación y trasmisión de los recursos fotográficos.
- No se ejecuta una correcta gestión de la información y el conocimiento entre las diferentes entidades que participan en la industria mediática.
- Existe una débil integración entre las aplicaciones desarrolladas.

Las dificultades antes mencionadas traen como consecuencias:

- El patrimonio fotográfico de los medios de prensa no es accesible para las redacciones.
- Desvalorización de las fotografías, dado por las deficiencias en los procesos de identificación y documentación.
- No se cuenta con la capacidad de respuesta, inmediatez en las publicaciones y cobertura en tiempo real.

En este sentido en la UCI se ha desarrollado un sistema de gestión documental de imágenes digitales para la Prensa que permite identificar, describir, catalogar, recuperar, y publicar los recursos fotográficos, además de generar automáticamente diferentes perfiles a partir de la imagen digital adicionada originalmente al sistema. Esta solución resuelve una parte del problema pero no garantiza en su totalidad la integración entre varios sistemas una vez implantados en las diferentes entidades, dificultando así una búsqueda y recuperación centralizada de la información.

Teniendo en cuenta **la situación problemática** descrita anteriormente se enuncia el siguiente **problema a**

resolver:

¿Cómo socializar la información en los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0?

El **objeto de estudio** de la investigación lo constituyen los sistemas de búsqueda y recuperación de la información y el **campo de acción** se define como: los metabuscadores para la búsqueda y recuperación de la información de imágenes digitales.

Para darle solución al problema descrito, se ha planteado el siguiente **objetivo general**:

Desarrollar un metabuscador para la centralización de los recursos fotográficos disponibles en los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.

Del cual se desglosan los siguientes **objetivos específicos**:

- Caracterizar las tendencias de los buscadores, multibuscadores y metabuscadores.
- Diseñar las funcionalidades del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Implementar las funcionalidades del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Validar las funcionalidades del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.

Para dar cumplimiento a los objetivos específicos planteado con anterioridad, se definen las siguientes **tareas de la investigación**:

- Caracterización de las tendencias de los buscadores, multibuscadores y metabuscadores.
- Identificación de los requerimientos del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Selección de las tecnologías, herramientas, estándares, patrones y metodologías necesarias para el desarrollo del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.

- Elaboración de la arquitectura de información del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Confección del diseño del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Implementación de las funcionalidades del metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.
- Aplicación de las pruebas de funcionalidad, seguridad, carga y estrés e integridad al metabuscador para los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.

Idea a defender:

Desarrollar un metabuscador para la centralización de los recursos fotográficos, facilitará la búsqueda y recuperación de información disponible en los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0.

Con la realización del trabajo se esperan los siguientes **resultados**:

Una aplicación web, basada en estándares, para la búsqueda centralizada en los sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0, que permita socializar y compartir los recursos fotográficos disponibles entre las diferentes entidades que utilicen como núcleo dicho sistema.

Para el desarrollo de la investigación se utilizaron diferentes métodos científicos, tanto teóricos como empíricos, que se ajustan al objeto de estudio y al cumplimiento de los objetivos trazados. Entre los que se encuentran:

Métodos Teóricos

- **Analítico-Sintético:** Se utilizó en las consultas de diversas fuentes bibliográficas, lo que contribuyó a la conformación del estado del arte, haciendo énfasis en el proceso de desarrollo de los metabuscadores en los Sistemas de Gestión Documental.
- **Análisis Histórico-Lógico:** Este método facilitó la comprensión lógica del objeto de estudio llevándose a cabo un análisis del proceso evolutivo por el cual ha transitado el desarrollo de los metabuscadores.

Métodos Empíricos

- **Entrevista:** Se empleó este método con el propósito de obtener información, ideas, puntos de vista que contribuyan al desarrollo de la investigación y aporten conocimientos específicos del tema.

El presente trabajo de diploma consta de 3 capítulos estructurados de la siguiente manera:

Capítulo 1 Fundamentación Teórica: En este capítulo se abordan los elementos teóricos que sustentan la investigación y el desarrollo del tema propuesto, a través del estudio y análisis de soluciones existentes. Se tratan las principales definiciones relacionadas con los buscadores, multibuscadores y metabuscadores. Se describen las tecnologías, herramientas, lenguajes y metodologías de desarrollo utilizadas para el análisis, diseño e implementación del sistema.

Capítulo 2 Características del sistema. Análisis y Diseño: Este capítulo define las características del metabuscador realizando el levantamiento de requisitos, que describirán en detalle las propiedades y condiciones con las que debe cumplir, además se definen, describen y priorizan los casos de uso, se determina la arquitectura y se lleva a cabo la realización del diseño gráfico de las interfaces del metabuscador. Se hace referencia a los patrones utilizados para su construcción.

Capítulo 3 Implementación y Prueba: En este capítulo se describen los artefactos relacionados con la implementación y las pruebas realizadas al metabuscador para validar su correcto funcionamiento.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se precisan elementos teóricos que sustentan la investigación y el desarrollo del tema propuesto, a través del estudio y análisis de soluciones existentes. Se tratan las principales definiciones relacionadas con los buscadores, multibuscadores y metabuscadores. Se describen las tecnologías, herramientas, lenguajes y metodologías de desarrollo utilizadas para el análisis, diseño e implementación del sistema.

1.2 Visión general del Programa de Informatización de la Prensa

El Programa de Informatización de la Prensa surge bajo el pensamiento de crear un periodismo digital, para ello se debía transformar la forma de hacer prensa en Cuba y generar un sistema de gestión de contenidos que permitiese ganar en visibilidad, capacidad de respuesta, inmediatez de las publicaciones y las coberturas en tiempo real.

En mayo 2005 surge en la UCI, el Programa de Informatización de la Prensa como exponente de un periodismo digital participativo y colaborativo. Dicho programa se ha enmarcado en la solución de los problemas comunes existentes entre los MCM más que en la realización de sitios para la prensa, tales como la recuperación de imágenes, audiovisuales, textos y colecciones.

Objetivos

- Realizar una transformación organizacional progresiva en los medios de prensa de manera que se estandaricen las formas de hacer sin llegar a esquemas demasiado rígidos.
- Garantizar que se asignen e instalen los recursos tecnológicos y las soluciones de conectividad que demanda el Programa.
- Crear un grupo colaborativo para el desarrollo e introducción de herramientas informáticas en función de la gestión de los medios de prensa.
- Implantar y personalizar las soluciones informáticas para cada medio.

El objetivo principal del Programa es lograr un desarrollo de aplicaciones de carácter general y a su vez la transferencia tecnológica que necesitan los medios para el desarrollo de sus versiones digitales. Siendo así necesario el desarrollo de aplicaciones como [4]:

- **Archivos de imágenes:** El archivo de imágenes es una aplicación en línea que almacena y permite la recuperación de imágenes digitales y utiliza los metadatos IPTC³ que son almacenados dentro de las propias imágenes.
- **Archivo de audiovisuales:** El archivo de audiovisuales es una aplicación en línea que almacena y permite la transmisión de audio y vídeo.
- **Archivo de textos:** Este es un archivo en línea que permite almacenar, catalogar y recuperar textos, posiblemente con enlaces a otros documentos que estén en el mismo Archivo, en otro Archivo de textos, en la Web, o dentro de alguno de los otros archivos descritos en este documento.
- **Archivo de colecciones:** El archivo de colecciones es una aplicación en línea que permite el almacenamiento, catalogación y recuperación de colecciones de elementos textuales, imágenes y audiovisuales.
- **Plataforma para la publicación web integrada con los archivos:** Se supone la selección y extensión de alguna plataforma web existente.
- **Herramientas para la cibermetría:** Desarrollar un producto informático que procese inteligentemente los registros de servidores web, genere estadísticas a partir de ello y las visualice en una interfaz web para asistir en la toma de decisiones

La estructura del Programa en cuanto al mapa de aplicaciones en un principio era un poco inestable y estaba expuesto a ciertas fallas, existía una premisa muy evidente y era el constante cambio en los sistemas de prensa en Cuba. Siendo consecuente con ello se puso en práctica una nueva estructura para las aplicaciones del sistema, tal como se puede apreciar en la Figura No. 1.

³IPTC: son conocidos como "encabezados IPTC" y utilizados para fotografías y noticias. IPTC incluye varios vocabularios controlados, traducidos a varios idiomas, que se actualizan de forma continua. Está pensado para facilitar el intercambio de información entre agencias de noticias y ofrece metadatos sobre autor, título, descripción, palabras clave, urgencia, derechos, etc.

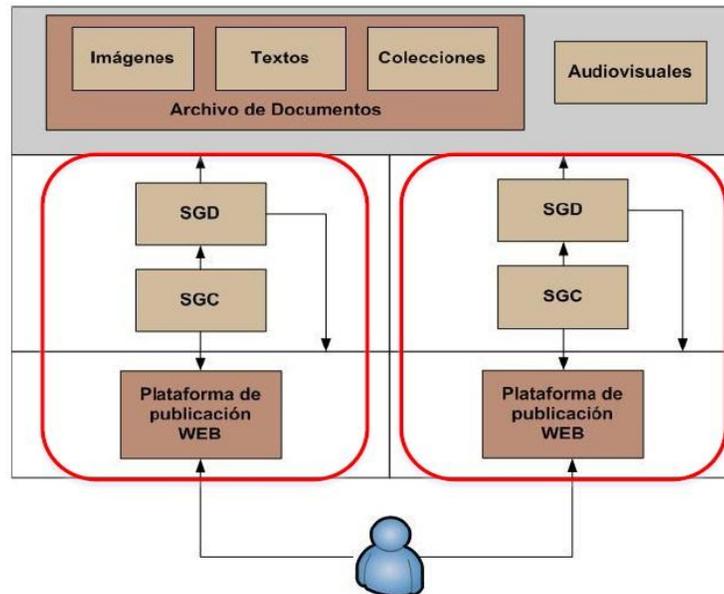


Figura No. 1: Mapa de aplicaciones del Programa de Informatización de la Prensa.

Como se puede apreciar en la figura anterior con la realización del Programa, se prevé la creación de cuatro archivos digitales en línea para el almacenamiento y recuperación de diferentes tipos de documentos como imágenes, noticias u otros géneros del periodismo, ficheros audiovisuales y también ediciones completas de publicaciones periódicas o eventuales. Ecumene Pyxel V1.0 es el nombre que se le ha dado al Archivo de Imágenes para el Programa de Informatización de la Prensa, el mismo permite recuperar las imágenes que están almacenadas en sus variantes web, miniatura y original, pero no garantiza la búsqueda centralizada de imágenes una vez desplegadas en los diferentes medios de prensa a nivel nacional, siendo necesario para ello el uso de un sistema de recuperación de información que facilite el acceso a los archivos fotográficos.

1.3 Sistemas de Recuperación de Información

La recuperación de información es una serie de procesos mediante los que un usuario localiza y accede a recursos de información relevantes. Para ello, se requiere la utilización de términos índice los cuales permiten indexar y recuperar documentos que tengan un alto grado de relevancia para el usuario.

En la actualidad se han creado sistemas de recuperación de información, que siendo automatizados, permiten el almacenamiento, procesamiento, recuperación y difusión de la información, sea esta de tipo texto, imágenes, audio, vídeo, multimedia, entre otros [5].

1.3.1 Buscadores

Los buscadores en el ámbito de la Informática son definidos como: *“un tipo de software que crea índices de bases de datos o de sitios web en función de los títulos de los ficheros, de palabras clave, o del texto completo de dichos ficheros. El usuario conecta con un buscador y especifica la palabra o las palabras clave del tema que desea buscar. El buscador devuelve una lista de resultados presentados en hipertexto, es decir que se pueden pulsar y acceder directamente al fichero correspondiente”* [6].

Por lo que se puede concluir que un buscador es un sistema o aplicación informática que permite la búsqueda de todo tipo de términos y palabras clave a partir del desarrollo de índices de archivos almacenados en servidores web.

Tipos de buscadores

Los buscadores se pueden diferenciar dado a la forma de obtener las direcciones que almacenan en su base de datos y como se muestra la información a los usuarios, dicha diferencia trae consigo que los buscadores se puedan clasificar de la siguiente manera [7]:

- Índice de búsqueda.
- Motor de búsqueda.
- Metabuscaadores.

El índice de búsqueda o directorios temáticos, como también se le conoce, fue el primer tipo de buscador que surgió y tiene la particularidad de que las direcciones almacenadas en su base de datos son construidas por un equipo humano y luego clasificadas por categorías según su contenido. En vista a un usuario sin un amplio conocimiento en el campo de la Informática es una herramienta viable, debido a que son muy fáciles de utilizar y permiten tener una idea general del contenido, sin embargo no logran cubrir gran parte de la red y debido a que su gestión se establece por acción humana su capacidad de respuesta es demasiado centrada.

Posteriormente surgieron los motores de búsqueda o “robots”, este tipo de buscador no requiere la búsqueda manual si no que el rastreo de la Web lo hace un programa, el cual va visitando las páginas y creando una base de datos con sus direcciones, de manera tal que provee un gran volumen de información facilitándole al usuario resultados incluso cuando no se define un criterio específico de búsqueda. Este tipo de buscador es muy efectivo e introduce una funcionalidad de búsqueda avanzada que posibilita acotar, dado filtros, el rango de búsqueda, no obstante tiene como deficiencia la posibilidad de presentar ambigüedades en sus respuestas.

Los metabuscadores son los últimos en esta clasificación y son ideados principalmente para centralizar las búsquedas, lo cual permite obtener un mayor número de resultados y realizar la búsqueda en diferentes buscadores.

Arquitectura

Como se puede apreciar en la Figura No. 2 la arquitectura de un buscador puede estar dada por distintos elementos [8]:

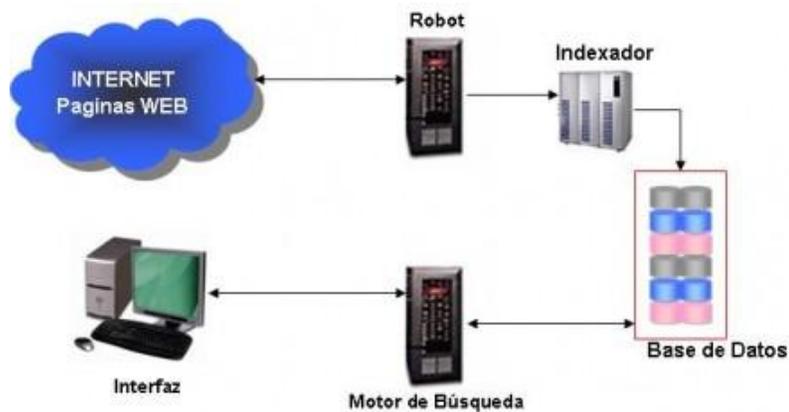


Figura No. 2: Arquitectura de un buscador.

- **Robot:** Las bases de datos de los buscadores se suelen construir utilizando robots (programas) que recorren la Web y recuperan los documentos de forma automática.

- **Indexador:** Se trata de un programa que recibe las páginas recuperadas por un robot (en ocasiones el robot y el indexador son el mismo programa), extrae una representación interna de la misma y la almacena en forma de índice en una base de datos.
- **Motor de búsqueda:** Programa que se encarga de analizar una consulta de usuario y buscar en el índice los documentos relacionados.
- **Interfaz:** La interfaz más utilizada es la basada en páginas web con formularios.

1.3.2 Multibuscadores

Un multibuscador es un servicio online que permite realizar búsquedas de un mismo término en diferentes buscadores. No dispone de índices propios, mediante API⁴ accede a las bases de datos de otros buscadores y genera los resultados en función de criterios propios.

El objetivo de un multibuscador es la de mejorar los resultados al no depender de un solo buscador, también ahorran tiempo al usuario al ofrecer una combinación de resultados de los distintos buscadores.

De igual manera varios autores afirman que un multibuscador es una página web que permite lanzar una consulta a diferentes motores de búsqueda, respetando el formato original de los buscadores. Una vez realizada la consulta ofrece las respuestas buscador por buscador. Teniendo el usuario que revisar los resultados de cada uno de forma separada [9].

1.3.3 Metabuscadores

Los metabuscadores son una herramienta de recuperación de la información que tiene como objetivo facilitar al máximo y hacer más efectiva la búsqueda de información en la red. Son llamados de diversas formas: metabuscador, megabuscador, *metaengine*, *parallel search engine*, *multiple search engine* y definidos como: sistemas que localizan información usando la base de datos virtual de otros buscadores, combinando las mejores páginas que devuelve cada uno en relación a un tema concreto. Se basan en la

⁴ API: Conjunto de convenciones internacionales que definen cómo debe invocarse una determinada función de un programa desde una aplicación.

premisa de que para obtener resultados confiables y recomendables es necesario buscar en varios motores de búsqueda [10].

Los autores del presente trabajo de diploma consideran que un metabuscador es el que permite lanzar varias búsquedas en varios robots. Realizan búsquedas en auténticos buscadores, analizan los resultados de las páginas, y presentan sus propios resultados, según un orden definido por el sistema estructural del metabuscador. En otras palabras se puede referir a un metabuscador como un buscador de buscadores.

Los metabuscadores son utilizados en distintos campos de la sociedad tales como: el metabuscador para la Universidad de Sevilla⁵ en la educación y el metabuscador para el Ministerio de Sanidad, Servicio Sociales e Igualdad⁶ en la salud, lo cual trae consigo que implementen funcionalidades distintas en dependencia del área en que se desarrollan, no obstante a ello todo metabuscador debe permitir [10]:

- Distribución de la búsqueda por áreas temáticas y por uno o varios recursos a la vez.
- Búsqueda simple y búsqueda avanzada.
- Determinar el número de resultados que se desea obtener por página y el número de resultados que se desea obtener por fuente, es decir, por recurso.
- Selección del número de registros obtenidos en la consulta.
- La ordenación de los resultados según varios criterios, incluido el recurso.
- Supresión de los duplicados.
- Filtrar los resultados para limitar aún más los resultados de la búsqueda.
- Exportar los resultados, imprimirlos o guardarlos.

Arquitectura

La estructura clásica de un metabuscador es claramente modular. Existen cinco módulos fijos y luego tantos módulos como motores de búsqueda estén siendo consultados. Cada uno de estos módulos se encarga de diversificar las consultas dependiendo de los distintos parámetros que los motores de búsqueda pueden aceptar.

⁵www.bib.us.es

⁶www.exelenciaclinica.net

La siguiente figura muestra los módulos fijos de un metabuscador [11]:

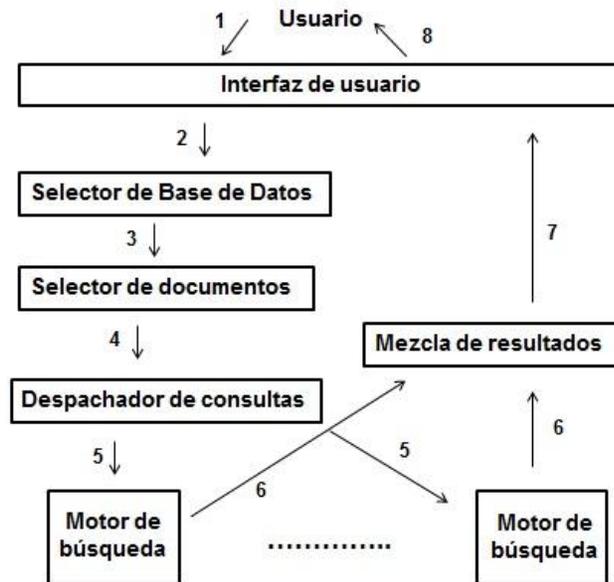


Figura No. 3: Estructura modular de un metabuscador.

- **Interfaz de usuario:** Es la interfaz de usuario, que debe ser de muy sencillo manejo y debe proporcionar una unificación de las distintas interfaces de los distintos motores de búsqueda.
- **Selector de Base de Datos:** Es un módulo clave para los metabuscadores que hacen una labor colaborativa con una cantidad alta o muy alta de motores de búsqueda. El selector permite evaluar la consulta y dependiendo del resultado de dicha consulta seleccionar a cuáles de los muchos motores de búsqueda se le envía. Esto permite que se produzca un ahorro en tiempo y que mejore el rendimiento ya que no se debe enviar la consulta a todos sino que sólo a una selección de éstos.
- **Selector de documentos:** Es la herramienta que vela por el correcto funcionamiento del programa. Sirve para hacer que los documentos que son devueltos sean lo más relevantes posible.
- **Despachador de consultas:** Es el que se ocupa de interactuar con los distintos módulos que tienen relación con los motores de búsqueda. Se encarga de mandar la consulta adecuada para cada motor de búsqueda y recoger los resultados devueltos.

- **Mezcla de resultados:** Es el encargado de poner en práctica los algoritmos desarrollados por el equipo del metabuscador que son los que permiten establecer una clasificación para ordenar los documentos devueltos.

El análisis de los sistemas de recuperación de la información, a partir de sus características y funcionamiento, permite concluir que los metabuscadores al cumplir con la filosofía de ser buscadores de buscadores, remitir las búsquedas a otros sistemas y mostrar los resultados obtenidos según una ponderación, posibilitan la integración necesitada por los medios de prensa.

1.4 Sistemas homólogos

Ixquick⁷:

Ixquick fue desarrollado y lanzado en 1998 por David Bodnick en Nueva York. En el año 2000, fue adquirido por Surfboard Holding B.V, una compañía privada holandesa. Desde sus comienzos en 1998, Ixquick ha crecido a un ritmo constante y ha ganado una serie de premios, entre los cuales se incluyen:

- Primer Sello Europeo de Privacidad otorgado por la autoridad de certificación Europrise.
- Premiado como mejor metabuscador por Search Engine Watch en 2000, 2002 y 2004.

Ixquick ha aumentado su popularidad de forma constante desde que se convirtió en el primer motor de búsqueda dedicado a la privacidad en 2006. Hasta enero de 2010, ha proporcionado 1.200 millones de búsquedas.

Ixquick es un metabuscador que navega simultáneamente múltiples motores de búsqueda y bases de datos en Internet para reunir y mostrar los resultados web más amplios y precisos. A diferencia de los motores de búsqueda simples como Google, Yahoo o Bing, la cobertura de Internet que ofrece Ixquick es mayor a la de cualquier motor de búsqueda. Al combinar los resultados de búsqueda, Ixquick puede ayudar a los usuarios a evitar la manipulación comercial de ciertos sitios conocidos por utilizar la técnica del encubrimiento “*cloaking*” para posicionarse artificialmente alto en los motores individuales [12].

⁷www.ixquick.com

METACRAWLER⁸

METACRAWLER es un sistema que provee servicios de búsqueda en paralelo y está disponible desde 1995 en la Universidad de Washington. Presenta a los usuarios una única interfaz de trabajo, los usuarios introducen la consulta y METACRAWLER la envía de manera paralela a varios servicios de búsqueda (motores). El sistema recoge los resultados y los clasifica en una sola lista en la cual se encuentra lo mejor de los motores de búsqueda utilizados.

En orden de realizar su objetivo el METACRAWLER tiene que adaptarse y realizar tareas. Este necesita emplear unas series de formato para cada motor de búsqueda y luego convertir los resultados al formato de salida propio. Una vez realizada la operación los añade de manera instantánea a los resultados encontrados por otros buscadores.

Finalmente debe analizar los resultados, eliminar los duplicados y realizar una revisión para asegurar la calidad [13].

KartOO⁹

KartOO S.A., fundada en el 2001, es una empresa francesa editora de programas informáticos especializados en los campos de la búsqueda de información, la gestión de conocimientos, la vigilancia y las interfaces visuales. El proyecto comenzó en 1997 con una entrevista con una editorial que quería desarrollar un sistema interactivo de navegación para las enciclopedias en CD-ROM. Plantearon este proyecto como la adaptación de la Teoría de las Redes Neuronales al área de la representación de la información. El 25 de Abril del 2002 se produce el lanzamiento de la nueva versión del buscador.

KartOO dispone de una tecnología de búsqueda de información y de representación visual y el metabuscador es solamente el escaparate de una serie de productos que se adaptan a todos los tipos de fuentes: portales, bases de datos, servidores de archivos, planificación de recursos empresariales (ERP), soluciones de gestión electrónica de documentos (GED) y de gestión de conocimientos (KM). Al realizar una búsqueda en el metabuscador KartOO los resultados son presentados en forma de mapa en el centro

⁸www.metacrawler.com

⁹www.kartoo.com

de la página, en la parte superior de la misma aparece un icono para acceder a la página principal de KartOO, la ayuda, borrar la búsqueda y el cuadro de búsqueda.

En el margen superior junto al cajetín muestra unos iconos que se utilizan para activar las siguientes funciones:

- Búsqueda únicamente en español.
- Búsqueda en todas las lenguas.
- Búsqueda simple.
- Búsqueda avanzada
- Historial: Muestra las búsquedas que el usuario ha realizado en KartOO.

En el margen derecho aparecen las banderas de todos los países para los que se ofrece un sitio KartOO: UK, USA, Brasil, Francia, España, Portugal, Alemania, Italia. También aparece una flecha para pasar al siguiente mapa.

Otras opciones que ofrece esta búsqueda avanzada es la de elegir los buscadores que se desea utilizar. Los buscadores con los que es posible realizar la búsqueda en KartOO son: Alltheweb, Altavista, HotBot, Yahoo, Wisenut, Teoma, Dmoz, MSN, Latinguia, Hispanista [14].

Teniendo en cuenta el estudio realizado de los sistemas homólogos, se puede apreciar que son aplicaciones de alto prestigio internacional y brindan funcionalidades que permiten el acceso a la información sin necesidad de un conocimiento previo del sistema. Realizan las búsquedas en la Web a partir de consultas ejecutadas a diversos motores de búsqueda, mostrando los resultados en dependencia del mecanismo de organización utilizado.

El análisis realizado constituye una base de conocimiento para el desarrollo de la solución, aunque el campo en que se realizan las búsquedas no es aplicable al producto esperado, su estudio representa un apoyo para su futura implementación. En adición, la investigación efectuada posibilita la elaboración de la arquitectura del sistema, haciendo énfasis en la representación de los recursos y funcionalidades en las distintas interfaces.

1.5 Tecnologías

1.5.1 ECM¹⁰

Alfresco 3.3

Es una alternativa libre y de código abierto que permite desarrollar proyectos de contenido empresarial. Cumple con la filosofía de software libre y cuenta con una creciente y fuerte comunidad de desarrolladores. Alfresco utiliza una arquitectura flexible para proveer gestión de documentos, gestión de contenido web y software colaborativo a más de 1.000 clientes empresariales a nivel mundial; además puede estar en muchas máquinas con distintas configuraciones funcionando como uno solo, y prestar servicios para muchos o pocos usuarios, seguridad y administración de usuarios, grupos de usuarios y roles [15].

Entre los múltiples beneficios del Alfresco se encuentra la colaboración de los usuarios, a través de módulos distribuidos y que no son muy complicados de implementar.

Tiene facilidad de uso, pues permite la conexión al gestor documental a través de unidades de red compartidas y presenta una búsqueda similar al buscador de Google o a Yahoo. Además tiene facilidades de administración como su instalación que es únicamente en el servidor, la gestión de los permisos es desde un único punto, la administración de manejo de versiones para los documentos.

1.5.2 Framework

Objetivos fundamentales de un *framework*:

- Acelerar el proceso de desarrollo.
- Reutilizar código ya existente.
- Promover buenas prácticas de programación como el uso de patrones.

¹⁰ECM: Gestión de Contenidos Empresariales.

Son las estrategias, métodos y herramientas utilizadas para capturar, gestionar, almacenar, preservar y entregar contenido y documentos relacionados con los procesos de la organización. Permiten la gestión de la información no estructurada de una organización, en la medida que la información existe.

Clasificación

Los *framework* pueden dividirse en tres grupos fundamentales:

- **Full Stack:** *Framework* completos que ofrecen todas las características y funcionalidades necesarias para construir aplicaciones web. Presentan un elevado acoplamiento entre sus componentes y se distribuyen como un todo. Ejemplo: CakePHP, Symfony2, etc.
- **Framework de componentes:** Sus componentes presentan un bajo acoplamiento por lo que pueden ser utilizados por partes, de acuerdo a la necesidad y el gusto del programador. Ejemplo: Zend Framework, Symfony2, etc.
- **Micro Framework:** Muy pequeños y ligeros. Ofrecen las funcionalidades básicas para desarrollar aplicaciones simples. Ejemplo: CodeIgniter, Symfony2, etc.

Se puede observar que Symfony2 se ajusta a los tres tipos de *framework* expuestos, lo cual constituye una de sus características: la flexibilidad.

Symfony 2.1.7: es un *framework* PHP basado en la arquitectura MVC (Modelo-Vista-Controlador). Fue escrito desde un origen para ser utilizado sobre la versión 5 de PHP ya que hace ampliamente uso de la orientación a objetos que caracteriza a esta versión y desde la versión 2 de Symfony se necesita mínimamente PHP 5.3.

Por más que Symfony puede ser utilizado para otros tipos de desarrollos no orientados a la Web, fue diseñado para optimizar el desarrollo de aplicaciones web, proporcionando herramientas para agilizar aplicaciones complejas y guiando al desarrollador a acostumbrarse al orden y buenas prácticas dentro del proyecto [16].

Este *framework* gracias a Doctrine¹¹, es capaz de soportar varios gestores de base de datos, algunos de estos son:

- MySQL
- PostgreSQL

¹¹ Es un ORM (Object Relation Mapper), técnica de programación que permite convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

- Oracle
- Microsoft SQL Server

Además incorpora soporte para MongoDB, el cual es un gestor de base de datos de tipo “NoSQL” [17].

1.5.3 Sistema Gestor de Base de Datos (SGBD)

PostgreSQL 9.1.8

Es un sistema integrado de alguna manera con parte del estándar SQL y permite la ejecución de consultas complejas así como claves externas.

Es extensible y multiplataforma. Soporta claves ajenas desde la versión 7.0 y soporta también transacciones. Posee características orientadas a objetos. Gran escalabilidad y rendimiento bajo grandes cargas de trabajo. Estabilidad y confiabilidad legendarias. Diseñado para ambientes de alto rendimiento [18].

A partir del estudio realizado se decide utilizar el ECM Alfresco en su versión 3.3, debido a que es un sistema de código abierto que permite la gestión de contenidos en una empresa y cumple las especificaciones del proyecto para el manejo y gestión de los recursos fotográficos. Además se escoge el *framework* de desarrollo de software Symfony en su versión 2.1.7, por ser una herramienta diseñada para optimizar el desarrollo de aplicaciones web que posibilita la reutilización de código, es ajustable a las características del producto y se encuentra en la base tecnológica del Centro de Ideo-Infomática. Como SGBD se selecciona PostgreSQL en su versión 9.1, debido a las características anteriormente expresadas y además es el sistema utilizado por el núcleo de Ecumene Pyxel V1.0

1.6 Lenguajes utilizados

XML

XML son las siglas del lenguaje de etiquetado extensible. Con la palabra "Extensible" se alude a la no limitación en el número de etiquetas, ya que permite crear aquellas que sean necesarias.

XML consiste de una serie de reglas, pautas o convenciones para planificar formatos de texto, de manera que produzcan archivos que sean fácilmente generados y leídos por un ordenador, que sean inequívocos y que eviten los problemas más comunes como la falta de extensibilidad, la falta de interoperabilidad entre plataformas o la falta de soporte para universalizar su tratamiento.

XML es un lenguaje que se ha creado para enriquecer la estructura de los documentos que pueden ser usados en la Web [19].

PHP 5.3.10

Es un lenguaje interpretado para el desarrollo de páginas web dinámicas del lado del servidor, cuyos fragmentos de código se intercalan fácilmente en páginas con extensión HTML, debido a esto, y a que es de *Open Source* (código abierto), es el más popular y extendido en la Web.

PHP es capaz de realizar determinadas acciones de una forma fácil y simple sin tener que generar programas implementados en un lenguaje distinto al HTML. Esto se debe a que ofrece un extenso conjunto de funciones para la explotación de bases de datos sin complicaciones. Es por esto, que levanta un mayor interés con respecto a los lenguajes pensados para los CGI¹² [20].

JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Técnicamente, es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con *JavaScript* se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

La ventaja básica de *JavaScript* consiste en su relativa sencillez, en tanto que el punto débil es que el código no es "seguro": estando intercalado en las mismas páginas, es perfectamente posible salvarlo y re-utilizarlo para otros fines [21].

¹²CGI: Common Gateway Interface, es un programa de interfaz que permite al servidor de Internet utilizar programas externos para realizar una función específica

CSS 3

Las hojas de estilo en cascada (en inglés, Cascading Style Sheets, por sus siglas CSS) constituyen el estándar para la inserción de estilo (tamaños, colores, tipografías, espacios, bordes) a documentos estructurados, como por ejemplo, documentos HTML o XML. El objetivo de la definición de este estándar del W3C¹³ es permitir la separación entre las normas de presentación y el propio contenido a mostrar.

Una de las características de la programación con hojas de estilo consiste en la posibilidad para definir los mencionados estilos de todo un sitio web. Esto es posible con la creación de un archivo donde se coloquen las declaraciones de estilos de la página y se enlacen todas las páginas del sitio. De este modo, todas las páginas comparten una misma declaración de estilos y, por tanto, si se realiza alguna modificación, cambiarán todas las páginas. Las hojas de estilo en cascada constituyen un estándar muy amplio, con grandes especificaciones y posibilidades [22].

HTML 5

El lenguaje de marcas de hipertextos (en inglés, HyperText Markup Language, por sus siglas HTML), es diseñado para estructurar textos y presentarlos en forma de hipertexto, siendo este el formato estándar de las páginas web.

Este lenguaje permite la introducción de referencias a otras páginas por medio de los enlaces de hipertexto y crear lenguajes de codificación descriptivos. Además define una estructura de documentos jerárquica, con elementos y componentes interconectados. No tiene un conjunto implícito de convenciones de señalización, por tanto, soporta un conjunto flexible de juegos de etiquetas [23].

Luego del estudio realizado se decide utilizar el lenguaje de programación PHP en su versión 5.3.10 para la implementación del código del lado del servidor, debido a que es un lenguaje interpretado, fácil de programar para un desarrollador y soportado en el *framework* escogido para la confección del código. Se escoge además el lenguaje JavaScript para las validaciones y la ejecución de mensajes del lado del cliente, pues brinda la posibilidad de reutilizar el código y mostrar efectos especiales en las páginas web.

¹³ www.w3c.es

Para la conformación de las páginas web se escoge el lenguaje HTML y para dar los estilos a las distintas interfaces del sistema se decide emplear el lenguaje de presentación CSS, debido a su uso en la presentación de documentos estructurados y escritos en HTML y a la posibilidad de incluir en un mismo archivo todos los estilos de una aplicación. Para la implementación de los servicios se escoge el lenguaje de etiquetado extensible, XML, debido a las características que posee y por ser el lenguaje utilizado para el desarrollo de otros servicios de Ecumene Pyxel V1.0.

1.7 Herramientas, técnicas y estilo de arquitectura

Visual Paradigm 8.0

Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML [24].

IDE Netbeans 7.2

Es una aplicación de código abierto diseñada para el desarrollo de aplicaciones fácilmente portables entre las distintas plataformas, haciendo uso de la tecnología Java. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, colaboración entre varias personas, creación de aplicaciones compatibles con teléfonos móviles, resaltado de sintaxis y además sus funcionalidades son ampliables mediante la instalación de paquetes.

Cuenta con módulos que hacen extensible su uso hacia otros lenguajes de programación como PHP, siendo compatible con la versión 5.3 de PHP. Tiene una creciente comunidad de usuarios, pues permite la depuración y ejecución de programas escritos [25].

Servidor Web Apache 2.2.2

Apache HTTP Server es un servidor web que se carga como un servicio más del sistema operativo. Cuando está activo, convierte la máquina en un servidor capaz de enviar contenido a cualquier navegador. Además es el servidor web gratuito más popular [26].

Varias fuentes aseguran que es un punto elemental para el desarrollo de aplicaciones web y en gran medida se debe a sus características y ventajas [27].

- Corre en una multitud de Sistemas Operativos.
- Apache es una tecnología gratuita de código fuente abierto.
- Es un servidor configurable de diseño modular.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurarlo para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de *logs*. Apache permite la creación de ficheros de *log*, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

Estilo de Arquitectura

REST

Es un estilo de arquitectura de software para sistemas hipermedios distribuidos como la Web. El término fue introducido en la tesis doctoral de Roy Fielding en 2000, quien es uno de los principales autores de la especificación de HTTP [28].

En realidad, REST se refiere estrictamente a una colección de principios para el diseño de arquitecturas en red. Estos principios resumen cómo los recursos son definidos y disecionados. El término “frecuentemente” es utilizado en el sentido de describir a cualquier interfaz que transmite datos específicos de un dominio sobre HTTP sin una capa adicional [28].

Cabe destacar que REST no es un estándar, ya que es tan solo un estilo de arquitectura, sin embargo, aunque no es un estándar, está basado en estándares:

- HTTP

- URL
- Representación de los recursos: XML/HTML/GIF/JPEG/...
- Tipos MIME: text/xml, text/html,...

Los objetivos de este estilo de arquitectura se listan a continuación [28]:

- **Escalabilidad de la interacción con los componentes:** La Web ha crecido exponencialmente sin degradar su rendimiento. Una prueba de ello es la variedad de clientes que pueden acceder a través de la Web: estaciones de trabajo, sistemas industriales, dispositivos móviles entre otros.
- **Generalidad de interfaces:** Gracias al protocolo HTTP, cualquier cliente puede interactuar con cualquier servidor HTTP sin ninguna configuración especial.
- **Puesta en funcionamiento independiente:** Los clientes y servidores pueden ser puestos en funcionamiento durante años. Por tanto, los servidores antiguos deben ser capaces de entenderse con clientes actuales y viceversa. Diseñar un protocolo que permita este tipo de características resulta muy complicado. HTTP permite la extensibilidad mediante el uso de las cabeceras, a través de las URLs, así como de la habilidad para crear nuevos métodos y tipos de contenido.
- **Compatibilidad con componentes intermedios:** Los más populares intermediarios son varios tipos de *proxys* para Web. Algunos de ellos, las *caches*, se utilizan para mejorar el rendimiento. Otros permiten reforzar las políticas de seguridad: *firewalls*. Y por último, otro tipo importante de intermediarios, *gateway*, permiten encapsular sistemas no propiamente Web. Por tanto, la compatibilidad con intermediarios permite reducir la latencia de interacción, reforzar la seguridad y encapsular otros sistemas.

A partir del estudio realizado se escoge la herramienta CASE Visual Paradigm en su versión 8.0 para el modelado del sistema, debido a que facilita la confección de los artefactos generados en las fases de Análisis, Diseño, Implementación y Despliegue; además cuenta con una exhaustiva documentación que prevé un uso efectivo del mismo. Para la implementación del código se selecciona el IDE Netbeans en su versión 7.2, al ser un entorno de desarrollo con funcionalidades que facilitan el trabajo y agilizan tiempo de los desarrolladores, además, forma parte de la base tecnológica del Centro de Ideo-Informática. Como

servidor web se selecciona Apache en su versión 2.2.2, pues es de código fuente abierto, soportado en cualquier sistema operativo y posibilita la personalización de los posibles errores que puedan ocurrir en el servidor. Finalmente, se selecciona el estilo de arquitectura REST para la comunicación entre los núcleos de Ecumene Pyxel V1.0 y el sistema, ya que ofrece una serie de principios en cuanto al diseño de arquitecturas en red aplicable al metabuscador.

1.8 Metodologías

RUP

El Proceso Unificado de Racional o Metodología RUP (en inglés, *Rational Unified Process*, por sus siglas RUP), es un producto comercial desarrollado y comercializado por Rational Software, una compañía de IBM. El proceso de software propuesto por RUP tiene tres características esenciales, es un proceso dirigido por casos de uso, centrado en la arquitectura y sigue una secuencia iterativa e incremental. RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto, y en las que se hace un mayor o menor hincapié en las distintas actividades.

Los aspectos que caracterizan a RUP se puede decir que cuenta con una forma disciplinada de asignar tareas y responsabilidades, que pretende implementar mejores prácticas, propone un desarrollo iterativo y el uso de la arquitectura basada en componentes, además de un riguroso control de cambios, se basa en el modelado visual y le da gran importancia a la verificación de la calidad del software. RUP se enfatiza en el uso exhaustivo de documentación durante todo el ciclo de vida del proyecto y es recomendada para los proyectos con grandes equipos de desarrollo [29].

OpenUp

Es un proceso modelo y extensible, dirigido a gestión y desarrollo de proyectos de software basados en desarrollo iterativo, ágil e incremental; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo. Permite un abordaje ágil al proceso de desarrollo de software, con sólo proveer un conjunto simplificado de contenidos, fundamentalmente relacionados con orientación, productos de trabajo, roles, y tareas [30].

Está caracterizado por cuatro principios básicos interrelacionados, a saber:

1. Colaboración para unificar intereses y compartir conocimientos.
2. Equilibrio de prioridades competentes a maximizar el valor de los involucrados con el resultado del proyecto.
3. Enfoque en la articulación de la arquitectura.
4. Desarrollo continuo para obtener realimentación y realizar las mejoras respectivas.

¿Por qué usar OpenUp?

- Metodología de desarrollo de software de código abierto diseñado para pequeños equipos organizados quienes quieren tomar una aproximación ágil del desarrollo.
- Proceso iterativo e incremental que es Mínimo, Completo y Extensible.
- Se valora la colaboración y el aporte de los *stakeholders* sobre los entregables y las formalidades innecesarias.
- Practicantes de desarrollo de software (desarrolladores, administradores de proyectos, analistas y probadores) trabajan juntos como un equipo de proyecto.
- No define un modelo de negocio ni de dominio necesario.
- Permite detectar errores tempranos a través de un ciclo iterativo.
- Evita la elaboración de documentación, diagramas e iteraciones innecesarios requeridos en la metodología RUP.
- Por ser una metodología ágil tiene un enfoque centrado al cliente y con iteraciones cortas [31].

Por lo expuesto anteriormente, se selecciona como guía para el desarrollo de la solución propuesta a OpenUp. Debido a que es una metodología aplicable a pequeños equipos de desarrollo, con iteraciones cortas que permiten una disminución en costo de tiempo y garantiza el éxito del producto sin un exceso de documentación.

1.9 Conclusiones

El estudio realizado de los sistemas de recuperación de la información permitió el conocimiento necesario para determinar sus características propias y diferencias notables, igualmente permitió la vinculación de las necesidades de los medios de prensa, en cuanto a la socialización de los recursos fotográficos, con las funcionalidades que brindan. Los sistemas homólogos analizados, aunque no representan una solución viable para el problema a resolver, sirvieron de modelo para la elaboración de la arquitectura del sistema.

Además se hizo un análisis de diferentes tecnologías, lenguajes y metodologías utilizadas mundialmente en un ciclo de desarrollo de software, lo cual permitió la selección de las más idóneas para el desarrollo del sistema.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA. ANÁLISIS Y DISEÑO

2.1 Introducción

En el presente capítulo se expresan las características principales del sistema a desarrollar. Se definen los requerimientos del sistema mostrándose los requisitos funcionales y no funcionales con que debe cumplir el mismo. Se enumeran los casos de uso de mayor prioridad junto a una breve descripción donde se expone una síntesis de lo que debe cumplir cada uno de ellos. Se describe el flujo de trabajo a partir del modelado del dominio mostrándose como artefacto: el diagrama del modelo del dominio del sistema. Además se precisan los elementos relacionados con el diseño del sistema realizándose una descripción de los estilos arquitectónicos y los patrones de diseño.

2.2 Características del sistema

Con el fin de dar respuesta al problema a resolver se propone un sistema de recuperación de la información, o sea un metabuscador que posibilite la búsqueda de imágenes digitales de manera centralizada en aquellas instituciones que adopten como Sistema de Gestión Documental de imágenes a Ecumene Pyxel.

En la actualidad el Archivo de Imágenes para el Programa de Informatización de la Prensa o sea Ecumene Pyxel V1.0 no se encuentra en la fase de despliegue; no obstante, una vez que dicho sistema sea desplegado y mantenga un correcto funcionamiento, el metabuscador debe permitir una comunicación entre la aplicación y los núcleos de Ecumene Pyxel V1.0 desplegados en las entidades periodísticas. Dicha comunicación se debe realizar utilizando el estilo de arquitectura REST, el cual propone un vocabulario de acciones fijo, que permite obtener recursos del servidor y una vez interpretados por el cliente, hacen que este cambie de estado.

Para la descripción de las imágenes se debe consumir el estándar IPTC utilizado en el sistema Ecumene Pyxel V1.0, o sea, cuando se realice una búsqueda dicho sistema debe proveer las imágenes con los metadatos incluidos, de forma tal que el metabuscador pueda brindar una descripción de sus resultados.

Se propone además que el metabuscador posea una estructura modular, permitiendo que los distintos módulos sean implementados por separado para posteriormente integrarlos en un todo.

Esta estructura en forma modular, posee ventajas como:

- Fácil integración con otros sistemas de aplicaciones favoreciendo la reutilización del código.
- Flexibilidad en las actualizaciones de los módulos. Una actualización crítica en un módulo, no debería tener profundos cambios en el funcionamiento del sistema como un todo.

2.3 Modelado del Dominio

La metodología de desarrollo de software, OpenUp, no define como artefacto necesario un modelo de negocio ni de dominio, sin embargo, en vista a que los procesos que intervienen en el sistema son engorrosos y no se definen con claridad, se decide realizar el modelo de dominio del sistema.

Un modelo del dominio captura los tipos más importantes de objetos en el contexto del sistema, los cuales representan los elementos que existen o los eventos que suceden en el entorno en el que trabaja dicho sistema. Muchos de los objetos o clases pueden obtenerse de una especificación de requisitos o mediante la entrevista con los expertos del dominio [32].

En el modelo del dominio para la solución propuesta se definen las siguientes clases:

- **Medio de Prensa X:** Sistema de gestión documental de un medio de prensa X.
- **Medio de Prensa Y:** Sistema de gestión documental de un medio de prensa Y.
- **Persona:** Individuo que accede a un sistema y descarga la información para luego publicarla en otro sistema.

Nota: X y Y son variables que representan a los medios de prensa nacional que utilicen como sistema de gestión documental a Ecumene Pyxel V1.0

En la actualidad, las entidades correspondientes a los medios de prensa poseen su propio sistema de gestión de información, en el cual se publican contenidos digitales, dichos contenidos pueden ser accedidos a través de su aplicación web y descargados por decisión de sus usuarios.

A continuación se muestra el diagrama de clases del modelo del dominio para el sistema, compuesto por clases y las relaciones entre ellas.

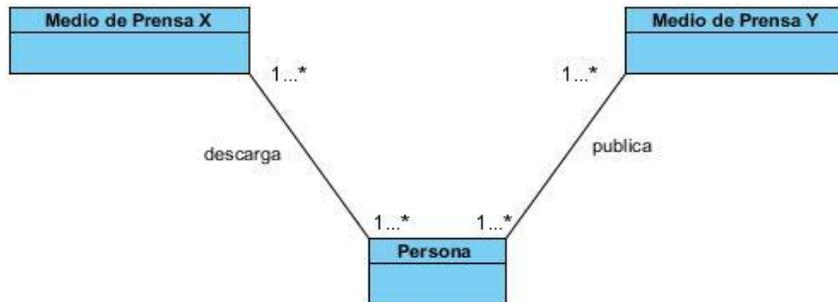


Figura No. 4: Diagrama de Clases del Modelo del Dominio.

2.4 Requerimientos de software

Los requerimientos son una descripción de las necesidades o deseos de un producto. La meta primaria de la fase de requerimientos es identificar y documentar lo que en realidad se necesita, en una forma que claramente se lo comunique al cliente y a los miembros del equipo de desarrollo. El reto consiste en definirlos de manera inequívoca, de modo que se detecten los riesgos y no se presenten sorpresas al momento de entregar el producto [33].

2.4.1 Requisitos funcionales

Los requisitos funcionales son las capacidades o condiciones que el sistema debe cumplir, ellos no alteran la funcionalidad del producto, o sea se mantienen invariables sin importarle con que propiedades o cualidades se relacionen.

A continuación se expresan los requisitos funcionales con que debe cumplir el sistema:

No.	Requisitos Funcionales	Descripción
RF_1	Autenticar usuario.	El sistema debe permitir que los usuarios se autenticuen en el metabuscador.
RF_2	Adicionar usuario.	El sistema debe brindar la posibilidad de adicionar una cuenta de usuario.
RF_3	Editar usuario.	La aplicación debe brindar la posibilidad de editar la cuenta de un usuario determinado.
RF_4	Eliminar usuario.	El sistema debe permitir eliminar una cuenta de usuario, lo cual sería eliminar permanentemente los datos de la persona de la base de datos.
RF_5	Deshabilitar usuario.	El sistema debe incluir una funcionalidad que permita deshabilitar una cuenta de usuario de manera tal que no tenga acceso a funcionalidades que requieran autenticación.
RF_6	Listar cuentas de usuarios.	El sistema debe permitir listar las cuentas de los usuarios que tienen acceso a él mismo.
RF_7	Solicitar cuentas de usuario.	El sistema debe brindar la opción de que un usuario pueda solicitar una cuenta de usuario.
RF_8	Mostrar proveedores.	Se deben mostrar los proveedores con que cuenta el sistema.
RF_9	Adicionar proveedor.	El sistema debe ser capaz de brindar la posibilidad de adicionar un nuevo proveedor.
RF_10	Editar proveedor.	Debe permitir que la información que se posea de los proveedores pueda editarse.
RF_11	Eliminar proveedor.	La aplicación debe contar con una funcionalidad que permita eliminar de forma permanente de la base de datos a un proveedor.

RF_12	Chequear proveedor.	El sistema debe brindar la posibilidad de chequear a los proveedores que posee registrados, lo cual debe ser idóneo para comprobar su correcto funcionamiento.
RF_13	Deshabilitar proveedor.	El sistema debe poseer una opción para deshabilitar a un proveedor determinado del sistema, de manera tal que aunque su acción sea bloqueada permanezca en la base de datos.
RF_14	Realizar búsqueda simple.	El sistema debe permitir que un usuario pueda realizar una búsqueda simple, o sea, que realice una búsqueda a partir de un criterio introducido por la persona.
RF_15	Realizar búsqueda avanzada.	El sistema debe brindar la posibilidad que un usuario pueda realizar la búsqueda de manera avanzada a partir de sentencias que acoten los resultados de su búsqueda.
RF_16	Salvar criterio de búsqueda.	El sistema debe ser consecuente con los caracteres introducidos por un usuario de manera tal que permita salvar los criterios de un usuario que ya ha obtenido resultados de una búsqueda.
RF_17	Adicionar taxonomías.	El sistema debe brindar la posibilidad de adicionar taxonomías.
RF_18	Editar taxonomías.	La aplicación debe contar con una opción que le posibilite a un usuario específico editar las taxonomías del sistema.
RF_19	Eliminar taxonomías.	El sistema debe permitir eliminar de manera permanente las taxonomías de la base de datos.
RF_20	Buscar taxonomías.	Debe permitir la funcionalidad de buscar una taxonomía.
RF_21	Visualizar página de mantenimiento.	El sistema debe incluir una interfaz que le permita a un usuario que intente acceder al mismo, conocer que no está disponible en ese instante.
RF_22	Visualizar contenidos (imágenes o unidades	Debe permitir visualizar los contenidos, los cuales deben mostrarse para todos los usuarios una vez que hayan accedido al sistema.

	documentales)	
RF_23	Compartir contenidos (imágenes o unidades documentales) con un amigo	Los usuarios deben contar con la posibilidad de compartir los contenidos con un amigo.
RF_24	Ayuda online.	El sistema debe incluir una interfaz de ayuda la cual le permita al usuario documentarse sobre las principales funcionalidades del sistema.
RF_25	Contáctanos.	Debe permitir que un usuario pueda contactar con los desarrolladores a partir de una interfaz que brinde esta opción.
RF_26	Visualizar ruta de navegación.	Visualizar la ruta de navegación es una funcionalidad que debe el sistema implementar en todas las vistas.

Tabla No. 1: Requisitos funcionales del sistema.

2.4.2 Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener, es decir las características que hacen a dicho producto más atractivo, usable, rápido o confiable. Son fundamentales en el éxito del producto, normalmente están vinculados a requisitos funcionales, o sea una vez se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser [33].

A continuación se enuncian los requisitos no funcionales que debe poseer el sistema según su clasificación:

Usabilidad

- A los administradores finales de la aplicación se les debe dar un adiestramiento básico en el uso de la aplicación. Estas personas deben tener un nivel de acceso amplio en la aplicación para poder darle respuesta a cada incidente ocurrido.

Eficiencia

- El sistema debe funcionar con un máximo rendimiento pero ajustado a bajas prestaciones de las computadoras debido que no todos los organismos poseen tecnología de punta.

Hardware

Hardware mínimo recomendado:

Del lado del cliente:

- 256 MB de RAM.
- Procesador Pentium IV de 3.0 GHz o superior.
- 40 GB de espacio de disco duro.

Del lado del servidor:

- 2 GB de RAM.
- Procesador Intel Dual Core o Core 2 Duo de 3.0 Ghz.
- 100 GB de espacio de disco duro.

Hardware recomendado:

Del lado del cliente:

- 1 GB de RAM o superior.
- Procesador Intel Dual Core de 3.0 GHz o superior.
- 40 GB de espacio de disco duro.

Del lado del servidor:

- 4 GB de RAM.
- Procesador Core i3 de 3.0 Ghz o superior.
- 250 GB de espacio de disco duro.

Restricciones de diseño

- Gestor de base de datos: PostgreSQL 9.1.8.
- Servidor de aplicaciones: Apache 2.2.2.
- Lenguaje de programación: PHP 5.3.10.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

- La ayuda y la documentación en línea deben estar en formato HTML, accesible a todos los usuarios del sistema.

Interfaz

- El sistema debe poseer una interfaz amigable al usuario, basada en el diseño web, brindando facilidades que permitan interactuar con el sistema de forma fácil y rápida.

Rendimiento

- La herramienta propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición.

Soporte

- Realizar Pruebas, Mantenimiento e Instalaciones necesarias para lograr el mejoramiento y evolución en el tiempo.

Software

- Sistema Operativo XP o superior, o un sistema GNU Linux.
- Gestor de base de datos: PostgreSQL 9.1.8.
- Servidor de aplicación: Apache 2.2.2.
- Herramienta CASE: Visual Paradigm 8.0.
- Framework de desarrollo: Symfony 2.1.7.

- Sistema de Gestión de Contenidos: Alfresco 3.3.

Seguridad

- La autenticación de un usuario en el sistema debe realizarse utilizando el certificado SSL.
- El acceso de un usuario al sistema debe estar restringido a una sesión única.
- El sistema debe desloguearse al cabo de 10 minutos inactivo.

2.5 Casos de uso del sistema

A partir de los requerimientos del sistema se generaron un conjunto de casos de uso los cuales representan las funcionalidades que deben ser desarrolladas en la solución propuesta.

Un caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso. Los casos de uso son historias o casos de utilización de un sistema; no son exactamente los requerimientos ni las especificaciones funcionales, sino que ejemplifican e incluyen tácticamente los requerimientos en las historias que narran [33].

En la siguiente figura se muestra el diagrama de casos de uso para el metabuscador.

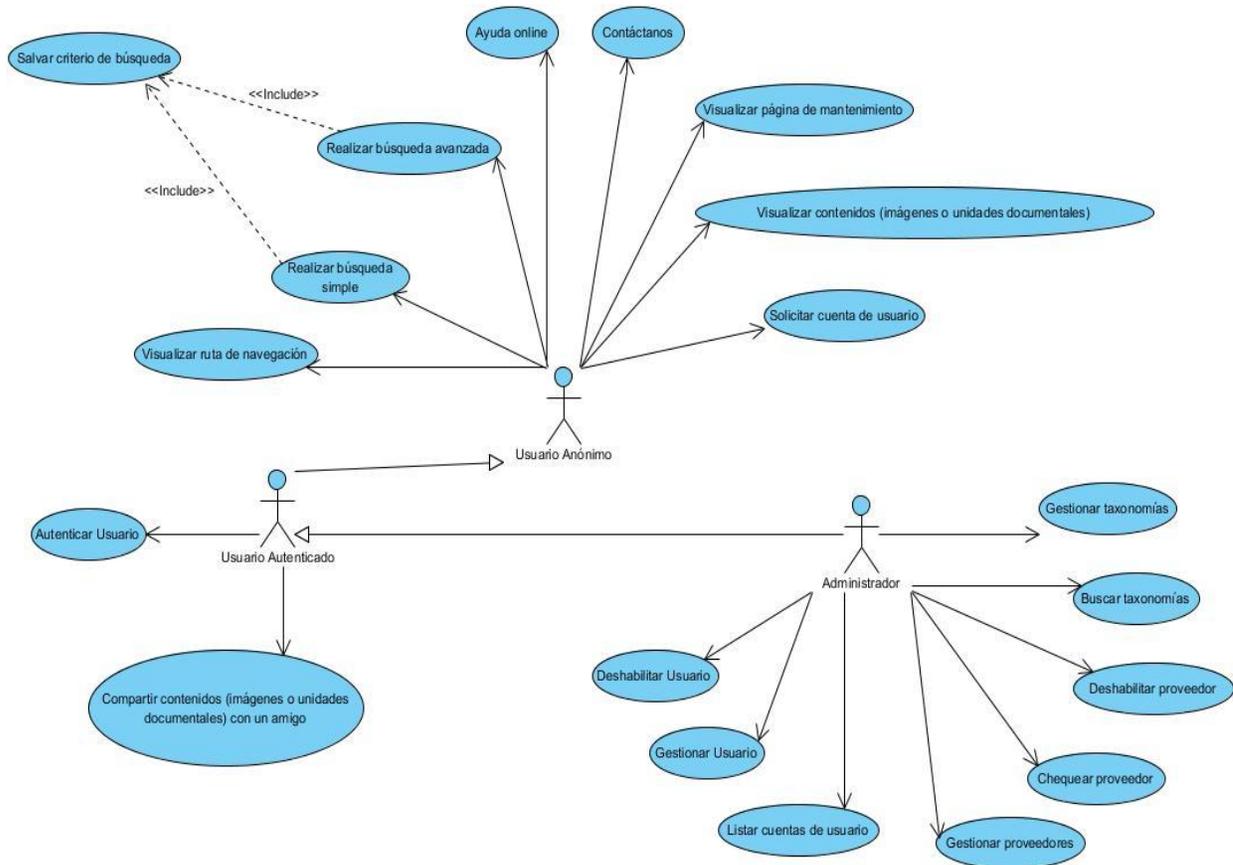


Figura No. 5: Diagrama de Casos de Uso del sistema.

A continuación se describe el caso de uso Realizar búsqueda simple:

CU No. 1: Realizar búsqueda simple.

Objetivo	Realizar la búsqueda de un contenido en el sistema.
Actores	Usuario Anónimo: (Inicia) Realiza la búsqueda simple. Usuario Autenticado: (Inicia) Realiza la búsqueda simple. Administrador: (Inicia) Realiza la búsqueda simple.

Resumen	El CU permite realizar una búsqueda en el sistema a partir de un criterio introducido por el usuario.	
Complejidad	Alta	
Prioridad	Alta	
Precondiciones	El usuario introduce un criterio de búsqueda.	
Postcondiciones	El usuario realizó la búsqueda.	
Flujo de eventos		
Flujo básico Realizar búsqueda simple		
	Actor	Sistema
1.	Abre su navegador y escribe la dirección URL del Metabusador.	Muestra la interfaz de inicio de la aplicación, en la cual aparece en la parte central un formulario con los siguientes campos: - Criterio de búsqueda - Tipo de contenido
2.	Introduce el criterio de búsqueda, selecciona el tipo de contenido y presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.	Valida que los datos sean correctos. Realiza la búsqueda a partir del criterio introducido por el usuario. Muestra los resultados.
3.		Termina el caso de uso.
Flujos alternos		
Nº 1 “Resultados no encontrados”		
	Actor	Sistema

2.1.	Introduce el criterio de búsqueda, selecciona el tipo de contenido y presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.	Muestra el siguiente mensaje: "No hay resultados disponibles"
Nº 2 "Criterio no introducido"		
	Actor	Sistema
2.1	Selecciona el tipo de contenido y presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.	Muestra el mensaje de error: "El campo "Criterio de Búsqueda" no puede estar vacío y debe tener como mínimo 3 caracteres para poder realizar la búsqueda."
Nº 3 "Tipo de contenido no seleccionado"		
	Actor	Sistema
2.1	Introduce el criterio de búsqueda y presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.	Muestra el mensaje de error: "Debe de elegir al menos un tipo de contenido para poder realizar la búsqueda."
Relaciones	CU Incluidos	Salvar criterios. Ver CU: Salvar criterio de búsqueda.
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 2: Especificación del caso de uso Realizar búsqueda simple.

2.6 Estilos arquitectónicos y patrones de diseño

Symfony en su versión 2.1.7 es el *framework* a utilizar durante el desarrollo del sistema, el cual está basado en la arquitectura MVC. Esta arquitectura permite dividir la aplicación en tres grandes capas [16]:

- **Vista:** Todo lo que se refiera a la visualización de la información, el diseño, colores, estilos y la estructura visual en sí de las páginas.
- **Modelo:** Es el responsable de la conexión a la base de datos y la manipulación de los datos almacenados en ella.
- **Controlador:** Es la capa encargada de manejar y responder las solicitudes del usuario, procesando la información necesaria y modificando el Modelo en caso de ser preciso.

A continuación se muestran algunas ventajas del uso de este estilo arquitectónico:

- **Soporte de múltiples vistas:** Dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Por ejemplo, múltiples páginas de una aplicación web pueden utilizar el mismo modelo de objetos mostrado de maneras diferentes.
- **Adaptación al cambio:** Los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Los usuarios pueden preferir distintas opciones de representación, o requerir soporte para nuevos dispositivos como teléfonos celulares. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo.

Symfony además utiliza patrones de diseño, los cuales son soluciones en forma de código a problemas comunes. Soluciones que se pueden reutilizar siempre que se utilice para el mismo problema y luego de haber comprobado su efectividad.

Entre los patrones utilizados en dicho *framework* se encuentran los siguientes [33]:

- **Experto:** Este patrón se utiliza con el fin de asignar una responsabilidad al experto en la información, es decir, la clase que cuenta con la información necesaria para cumplir la responsabilidad. Es un patrón que se usa más que cualquier otro al asignar responsabilidades; siendo un principio básico que a utilizar en el diseño orientado a objetos.

Es uno de los patrones que más se utiliza cuando se trabaja con Symfony, con la inclusión de la librería *Propel* para mapear la Base de Datos. Symfony utiliza esta librería para realizar su capa de

abstracción en el modelo, encapsular toda la lógica de los datos y generar las clases con todas las funcionalidades comunes de las entidades.

- **Creador:** El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento, por lo que indica que la clase incluyente del contenedor o registro es idónea para asumir la responsabilidad de crear.

En la clase *Actions* se encuentran las acciones definidas para el sistema, en dichas acciones se crean los objetos de las clases que representan a las entidades, lo que evidencia que la clase *Actions* es creador de dichas entidades.

- **Patrón Decorador:** Este patrón permite añadir responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades.

En Symfony 2.1.7 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. Su uso aporta una mayor flexibilidad que la herencia estática, permitiendo, entre otras cosas, añadir una funcionalidad dos o más veces.

2.7 Diagramas de Clases del Diseño

Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como las relaciones entre los mismos. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Además son la base para un par de diagramas relacionados conocidos como: diagramas de componentes y diagramas de despliegue. Dichos diagramas son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa [34].

Contienen normalmente los siguientes elementos:

- Clases.
- Interfaces.
- Colaboraciones.
- Relaciones de dependencia, generalización y asociación.

A continuación se muestra el diagrama de clases del diseño con estereotipos web para el caso de uso Realizar búsqueda simple.

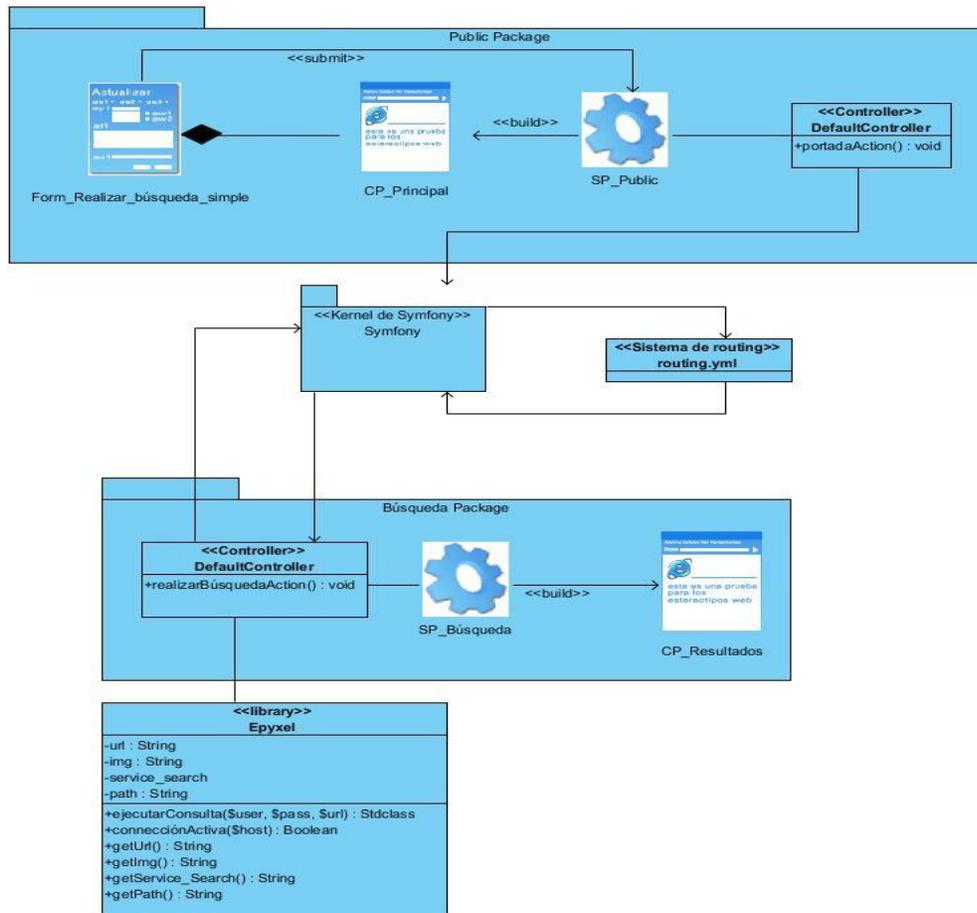


Figura No. 6: Diagrama de Clases del Diseño del caso de uso Realizar búsqueda simple.

2.8 Diagrama de Interacción

Un diagrama de interacción explica gráficamente las interacciones existentes entre las instancias y las clases del modelo de estas. El punto de partida de las interacciones es el cumplimiento de las condiciones de los contratos de operación.

El UML define dos tipos de estos diagramas; ambos sirven para expresar interacciones semejantes o idénticas de mensaje:

- diagramas de colaboración
- diagramas de secuencia

Los diagramas de colaboración describen las interacciones entre los objetos en un formato de grafo o red y los de secuencia describen las interacciones en una especie de formato de cerca o muro [33].

En la Figura No. 7 se muestra el diagrama de secuencia para el caso de uso Realizar búsqueda simple. En dicha imagen se muestra el flujo de acciones que se realiza un usuario al realizar una búsqueda en el sistema a partir de un criterio y un tipo de contenido.

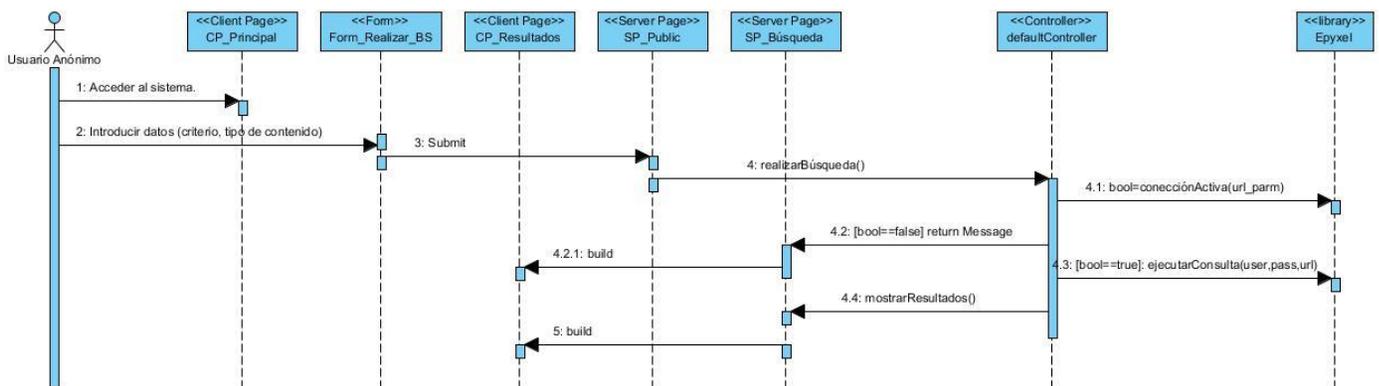


Figura No. 7: Diagrama de Secuencia para el caso de uso Realizar búsqueda simple.

2.9 Modelo de Despliegue

En la Figura No. 8 se muestra el diagrama de despliegue del sistema el cual representa las configuraciones físicas de funcionamiento de la aplicación. Está integrado por 4 nodos, 2 dispositivos y 2 servidores. Los dispositivos representan los ordenadores de los cuales se debe acceder al metabuscador,

uno hace referencia a un usuario que puede ser Usuario Anónimo o Usuario Autenticado y el otro se refiere al administrador del sistema, el cual debe estar en contacto directo con la aplicación. El servidor de aplicaciones es donde la aplicación se debe encontrar permitiendo que los ordenadores puedan acceder a la información que es almacenada en él, así como permitir la realización de modificaciones y actualizaciones sobre su estructura. Por último el servidor de base de datos es donde se debe albergar toda la información que perdurará en el tiempo, este servidor es una computadora conectada a una red que ejecuta un sistema de base de datos donde se deben realizar peticiones respecto al contenido que este contiene.

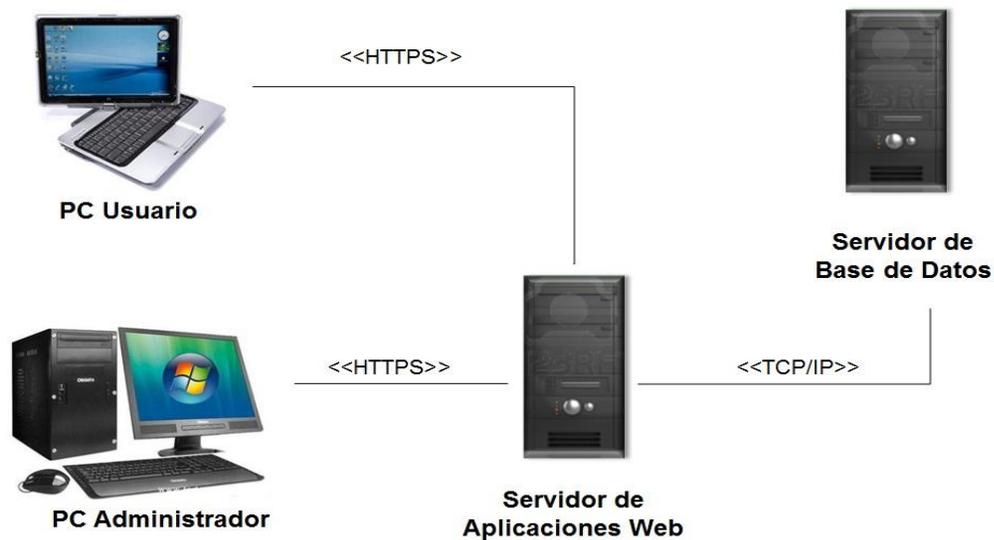


Figura No. 8: Diagrama de Despliegue del sistema.

2.10 Conclusiones

La caracterización del sistema sirvió para tener un mayor entendimiento de las funcionalidades del mismo. Los requerimientos definidos por el cliente posibilitaron un entendimiento de las expectativas a alcanzar con el desarrollo de la aplicación, de qué debe cumplirse y bajo qué circunstancias. El modelado del sistema durante la fase de diseño proporcionó las bases para el comienzo de la fase de implementación.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA

3.1 Introducción

En el presente capítulo se describen los elementos que componen el diagrama de componentes, el cual permite modelar, de manera estructurada, los elementos del sistema. Además se describe el estándar de codificación utilizado durante la implementación y se muestran las interfaces de Inicio para la ejecución de las búsquedas. Conjuntamente se muestran las pruebas realizadas así como los resultados obtenidos luego de su ejecución.

3.2 Diagrama de componentes

Un diagrama de componente es, como su nombre lo indica, un esquema o diagrama que muestra las interacciones y relaciones de los componentes de un modelo. Entendiéndose como componente a una clase de uso específico, que puede ser implementada desde un entorno de desarrollo, ya sea de código binario, fuente o ejecutable; dichos componentes poseen tipo, que indican si pueden ser útiles en tiempo de compilación, enlace y ejecución [35].

Este tipo de diagrama se representa a partir de componentes unidos mediante relaciones de dependencia, generalmente de compilación. A continuación se presenta el diagrama de componentes del sistema:

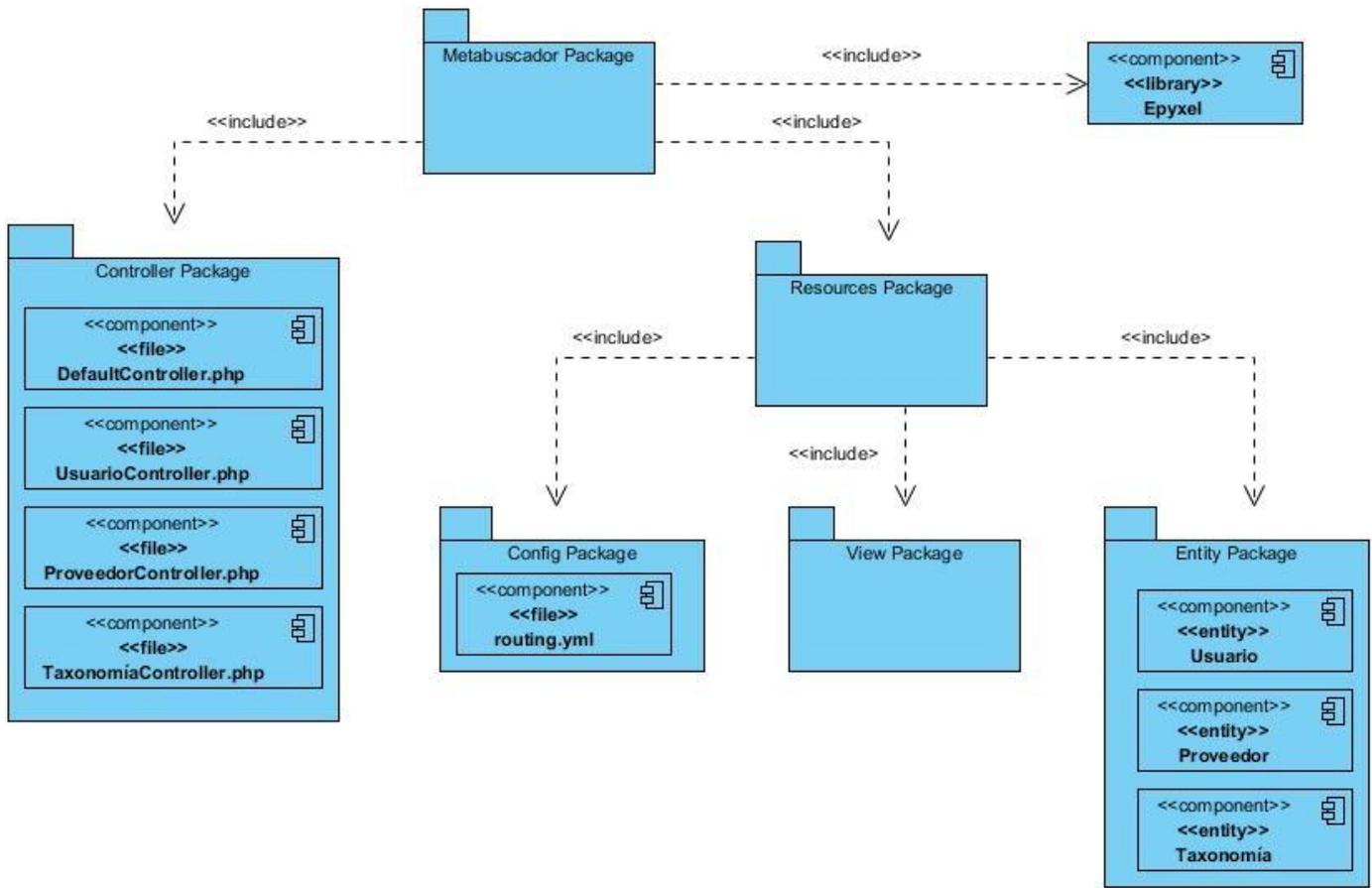


Figura No. 9: Diagrama de Componentes del sistema.

Descripción del diagrama de componentes

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

A continuación se describen los elementos que integran el diagrama de componentes del sistema:

- **Metabuscador Package:** Es el paquete que integra todos los componentes del sistema.
- **Controller Package:** Es donde se encuentran las clases controladoras del metabuscador.
- **Resource Package:** Es el que integra los paquetes Config, Entity y View.

- **Entity Package:** Es el paquete que agrupa las entidades de la aplicación y permite el acceso a la base de datos.
- **Config Package:** Es donde se almacenan los archivos de configuración.
- **View Package:** Agrupa las plantillas HTML del sistema, las cuales deben permitir la visualización de la información por parte de los usuarios.
- **Epyxel:** Es una librería que contiene las funcionalidades para la conexión con los núcleos de Ecumene Pyxel V1.0.

3.3 Código fuente

El código fuente es un conjunto de líneas que conforman un bloque de texto, escrito según las reglas sintácticas de cierto lenguaje de programación, destinado a ser legible por humanos. En su forma original, tal y como fue escrito por el programador, es un programa que no puede ser ejecutable directamente por el computador, por lo cual debe convertirse en lenguaje de máquina mediante compiladores, ensambladores o intérpretes. El metabuscador siendo una aplicación informática, posee un código fuente legible y comprensible por cualquier desarrollador [36].

3.3.1 Estándar de codificación

Con el fin de garantizar una homogeneidad en el código de la aplicación se utilizó el estándar de codificación basado en Pautas de Codificación PHP de Zend Framework.

Dicho estándar presenta temas relacionados con la estructura y confección del código, en un sistema bajo el uso de lenguaje PHP:

Convenciones de nombre

Los nombres de las clases contienen solo caracteres alfanuméricos, en su gran mayoría se encuentran compuestos por más de una palabra por lo que la primera letra de cada una aparece en mayúscula. Las barras bajas son utilizadas como sustitución de los espacios en blancos, en los nombres que poseen las características anteriormente expresadas.

Las funcionalidades implementadas poseen nombres que sugieren el propósito y comportamiento de la acción a realizar. Particularmente en los métodos de acceso para las instancias o variables estáticas, los caracteres que comienzan sus nombres son “*get*” o “*set*”, indistintamente por la acción que se vaya a realizar.

Todos los archivos que contienen código PHP terminan con la extensión “.php” tal y como se muestra a continuación:

- ProveedorController.php

Estilos de código

El código PHP se encuentra delimitado por la forma completa de las etiquetas PHP estándar, como se muestra en la figura:

```
<?php
namespace Metabuscador\TaxoBundle\Controller;
use Symfony\Bundle\FrameworkBundle\Controller\Controller;
class DefaultController extends Controller
{
    public function indexAction($name)
    {
        return $this->render
('TaxoBundle:Default:index.html.twig', array('name' => $name));
    }
}
?>
```

Figura No. 10: Código PHP de la clase TaxoController.php

3.4 Interfaces principales de la aplicación

El metabuscador cuenta con una interfaz web amigable y sencilla, fácil de utilizar y con cierto parecido a los sistemas desarrollados para el Programa de Informatización de la Prensa. A continuación se muestra la interfaz de inicio con las funcionalidades: Realizar búsqueda simple y Realizar búsqueda avanzada.

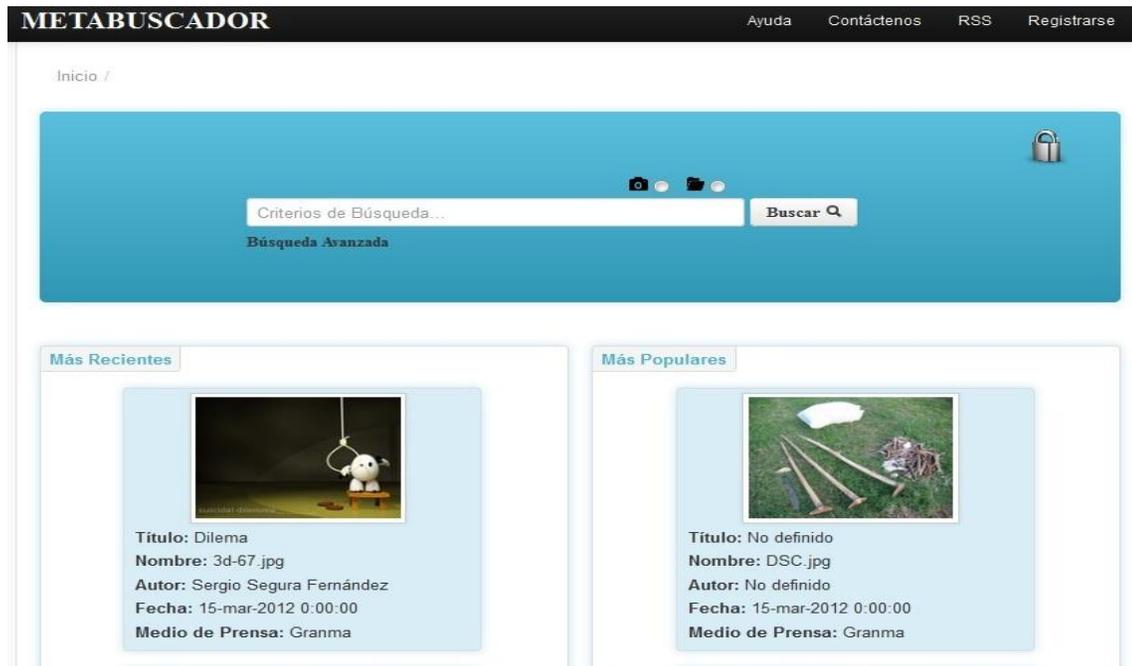


Figura No. 11: Interfaz web Inicio.



Figura No. 12: Interfaz web Inicio para la búsqueda avanzada.

3.5 Validación del sistema

La validación del software o, de forma más general, la verificación y validación se utilizan para mostrar que el sistema se ajusta a su especificación y que cumple las expectativas del usuario final. Implica procesos de comprobación, como las inspecciones y revisiones, en cada etapa del proceso de software desde la definición de requerimientos hasta en desarrollo del programa. Sin embargo, la mayoría de los costos de validación aparecen después de la implementación, cuando se prueba el funcionamiento del sistema [35].

3.5.1 Pruebas funcionales

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 o pruebas de caja negra, ya que los analistas de pruebas, no enfocan su atención a como se generan las respuestas 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 [37].

A continuación se muestran los casos de prueba funcionales a realizar sobre las interfaces Autenticar usuario y Realizar búsqueda simple.

Caso de Prueba
Nombre de la sección: Autenticar usuario
Condiciones de ejecución: El usuario debe acceder a través del navegador al sistema.
Pasos de ejecución: Se deben introducir los caracteres correspondientes a los siguientes campos: <ul style="list-style-type: none">• Nombre de usuario• Contraseña Luego presionar el botón Acceder.

Escenarios	Usuario	Contraseña	Respuesta del sistema
Entrada de los datos de autenticación correctos.	Maridel Elena	elena123	Muestra la interfaz correspondiente al usuario autenticado.
Los datos introducidos en el campo Nombre de usuario son incorrectos.	ele	elena123	Muestra el mensaje de error: "El nombre de usuario y/o contraseña son incorrectos".
Los datos introducidos en el campo Contraseña son incorrectos.	Maridel Elena	Almira_852	Muestra el mensaje de error: "El nombre de usuario y/o contraseña son incorrectos".
El campo Nombre de usuario se encuentra vacío.	" "	elena123	Muestra el mensaje de error: "Por favor rellene este campo".
El campo Contraseña se encuentra vacío.	Maridel Elena	" "	Muestra el mensaje de error: "Por favor rellene este campo".

Tabla No. 3: Caso de prueba para el requisito Autenticar usuario.

Caso de Prueba
Nombre de la sección: Realizar búsqueda simple.
Condiciones de ejecución: El usuario debe acceder a través del navegador al sistema.
Pasos de ejecución: <ol style="list-style-type: none"> 1. Introducir el criterio de búsqueda. 2. Seleccionar el tipo de contenido. 3. Presionar el botón Buscar.

Escenarios	Criterio de búsqueda	Tipo de contenido	Respuesta del sistema
Búsqueda realizada correctamente.	José Martí	Unidad documental	Realiza la búsqueda y muestra los resultados en una interfaz.
Campo vacío.	“ ”	Fotografía	Muestra el mensaje de error: “Por favor rellene este campo”.
Tipo de contenido no seleccionado.	Abel Santamaría	“ ”	Muestra un mensaje de error indicando que debe seleccionar un tipo de contenido.
Resultado no encontrado.	Débora Noriega	Unidad documental	Muestra un mensaje indicando que no se encontraron resultados.

Tabla No. 4: Caso de prueba para el requisito Realizar búsqueda simple.

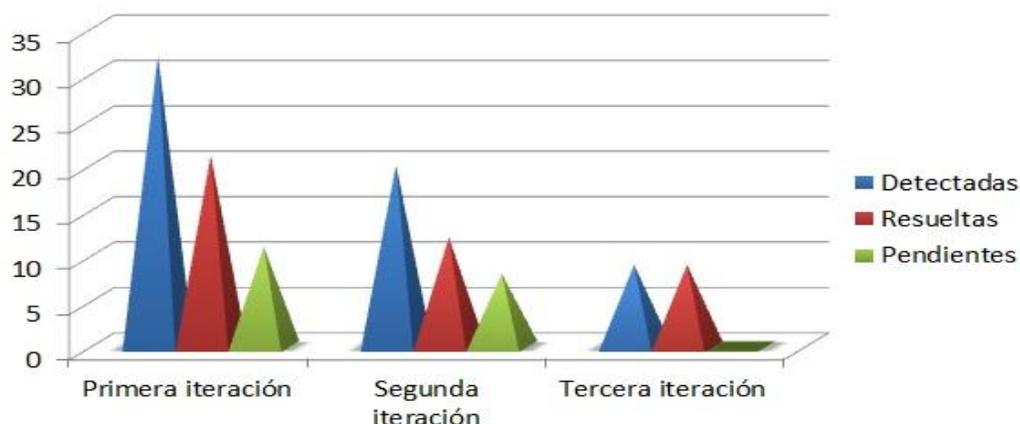
Resultados de las pruebas funcionales

A partir del diseño de los casos de prueba se procedió a su ejecución, validando así los requisitos funcionales del sistema, para ello se realizaron tres iteraciones de pruebas. A continuación se muestran los resultados obtenidos en cada una de las iteraciones realizadas.

No conformidades	Primera iteración	Segunda iteración	Tercera iteración
Detectadas	32	20	9
Resueltas	21	12	9
Pendientes	11	8	0

Tabla No. 5: Resultado de prueba funcionales según las no conformidades.

La siguiente gráfica muestra los estados de comportamiento de las no conformidades en cada una de las iteraciones de pruebas realizadas.



Gráfica No. 1: Comportamiento de las no conformidades en las pruebas funcionales.

A continuación se exponen una serie de no conformidades detectadas durante la realización de las pruebas a la aplicación.

Iteración	No conformidades detectadas
1 ^{ra}	El entorno de trabajo de un Administrador es expuesto ante un Usuario Anónimo luego de autenticarse.
1 ^{ra}	Admite caracteres extraños a la hora de adicionar un usuario al sistema.
2 ^{da}	No es posible adicionar una foto a un proveedor del sistema.
2 ^{da}	No se muestra la página de mantenimiento del sistema.
3 ^{ra}	No permite ejecutar la funcionalidad: Comprobar proveedor.
3 ^{ra}	No se muestra un mensaje de error cuando un usuario deshabilitado intenta autenticarse en el sistema.

Tabla No. 6: No conformidades detectadas.

3.5.2 Pruebas de seguridad

Las pruebas de seguridad son realizadas con el fin de comprobar los mecanismos de protección en el sistema. Para ello se mide el nivel de seguridad en cuanto a:

- **La aplicación:** al verificar que un actor solo pueda acceder a las funcionalidades y datos que su usuario tiene permitido.
- **El sistema:** al verificar que solo los actores con acceso al sistema y a la aplicación están habilitados para accederla.

Para la ejecución de la pruebas se utilizó como guía una lista de chequeo confeccionada por especialistas del Centro de Calidad de Software (Calisoft), la cual posibilitó la evaluación de la seguridad del metabuscador en un primer nivel.

Además se tuvieron en cuenta los roles representados en la aplicación: Usuario Anónimo, Usuario Autenticado y Administrador. Se comprobaron las funcionalidades establecidas según los privilegios de cada actor y la información presentada a los mismos. Se llevaron a cabo acciones para lograr incurrir en la seguridad, en cuanto a la autenticación en el sistema, tratando de acceder al mismo siendo un Usuario Anónimo.

Para la evaluación del software en un segundo nivel se utilizó la herramienta Acunetix WVS Reporter, la cual detectó un total de 11 vulnerabilidades, de ellas 5 de información, 3 de nivel bajo y 3 de nivel medio.

Luego de finalizar las pruebas se puede concluir que a pesar de poseer vulnerabilidades el metabuscador es una aplicación web que implementa mecanismos de seguridad, capaces de velar por el acceso al sistema y a las funcionalidades brindadas.

3.5.3 Pruebas de carga y estrés

Para la realización de las pruebas de carga y estrés se escogió la herramienta Jmeter, la cual es una aplicación java que permite definir comportamientos para casos de pruebas y medir su rendimiento. Dicha herramienta posibilita diagnosticar las respuestas del sistema, realizar un análisis de la capacidad de carga y rendimiento de la aplicación así como definir el número de usuarios que acceden simultáneamente al mismo.

Las pruebas se realizaron para un total de 120 usuarios concurrentes en un intervalo de 1 segundo, reportando un 1.09% de error para un rendimiento del servidor de 7.1 peticiones por segundo.

El tiempo de respuesta mínimo fue de 0.001 segundos y el máximo de 4 segundos para una media de 9409 páginas cargadas satisfactoriamente.

Al concluir las pruebas se pudo determinar que el sistema para dicha carga responde correctamente, llegando a un estado de estrés si supera el número de los usuarios estimados.

3.5.4 Pruebas de integración

Las pruebas de integración tienen como objetivo verificar si los componentes o subsistemas interactúan correctamente a través de sus interfaces, tanto internas como externas. Además comprobar si los mismos cubren la funcionalidad establecida y se ajustan a los requisitos especificados en las verificaciones correspondientes.

La necesidad de realizar dichas pruebas viene dada por el hecho de que los módulos que forman un programa suelen fallar cuando trabajan de forma conjunta, aunque previamente se haya demostrado que funcionan correctamente de manera individual. Por ello se realiza este tipo de pruebas, asegurando que los módulos que están relacionados se ejecuten correctamente [38].

Para realizar la integración del sistema se dispone de un servidor de base de datos y dos sistemas de gestión documental de imágenes digitales Ecumene Pyxel V1.0, a los cuales el metabuscador debe integrarse y responder correctamente.

A continuación se muestran los casos de prueba previstos para realizar la validación del sistema.

Caso de Prueba
Nombre de la sección: Ecumene Pyxel V1.0
Condiciones de ejecución: <ul style="list-style-type: none">• El usuario debe acceder a través del navegador al sistema.• El usuario debe tener privilegios de administración.

<ul style="list-style-type: none"> El sistema debe tener al menos un proveedor en su base de datos. <p>Pasos de ejecución:</p> <ol style="list-style-type: none"> Escoger la opción Comprobar Proveedor. 	
Escenarios	Respuesta del sistema
Proveedor disponible.	Muestra el mensaje siguiente: "El proveedor solicitado está disponible".
Proveedor no disponible.	Cambia el estado del proveedor a deshabilitado y muestra el mensaje siguiente: "El proveedor solicitado no está disponible".

Tabla No. 7: Caso de prueba para los proveedores del sistema.

Caso de Prueba	
Nombre de la sección: Servidor de BD	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> El usuario debe acceder a través del navegador al sistema. El usuario debe tener privilegios de administración. <p>Pasos de ejecución:</p> <ol style="list-style-type: none"> Introducir un criterio de búsqueda. 	
Escenarios	Respuesta del sistema
Base de datos disponible.	Muestra el resultado de la búsqueda en caso satisfactorio, por lo contrario muestra un mensaje indicando que no se han encontrado resultados.
Base de datos no disponible.	Muestra una interfaz con un mensaje de error.

Tabla No. 8: Caso de prueba para la base de datos del sistema.

Resultados de las pruebas de integración.

Luego de ejecutar los casos de prueba previamente mostrados se puede concluir que en caso hipotético de que el sistema se encuentre afectado por alguna avería o fallo de dicha índole, el metabuscador es capaz de mantener su funcionamiento.

3.6 Conclusiones

El uso de patrones arquitectónicos y de diseño a la hora de la implementación del sistema, permitió la presentación de una aplicación enfocada a las especificidades del cliente y encaminada al cumplimiento de su objetivo. En el desarrollo de la solución se generaron un grupo de artefactos entre ellos el diagrama de componentes, el cual posibilitó una mejor comprensión de la estructura del sistema, y se enfatizó en la legibilidad y homogeneidad del código, con el uso del estándar de codificación de Zend Framework. Tras la ejecución de pruebas al sistema se pudo comprobar el correcto funcionamiento del mismo y el cumplimiento de las expectativas del usuario final.

Conclusiones

Durante la realización del presente trabajo de diploma se llevó a cabo un ciclo completo de Ingeniería de Software, utilizando como guía una metodología ágil de desarrollo con iteraciones cortas y centradas en el cliente.

Se cumplieron los objetivos propuestos y se arribaron a las siguientes conclusiones:

- A partir del estudio realizado en el estado del arte, sobre los sistemas de recuperación de la información, se concluyó que un metabuscador es la herramienta necesaria para garantizar la búsqueda centralizada de imágenes digitales en los sistemas de gestión documental Ecumene Pyxel V1.0.
- Con el diseño del sistema se logró un mejor entendimiento y organización de los elementos integrados al metabuscador, sirviendo así de guía para la implementación del producto final.
- El metabuscador desarrollado permite la búsqueda de imágenes digitales, de manera rápida y sencilla, posibilitando a los usuarios una interfaz que integre las imágenes disponibles en los medios de prensa que utilicen como sistema de gestión documental a Ecumene Pyxel V1.0.
- Las validaciones realizadas al sistema demostraron resultados satisfactorios, siendo las no conformidades detectadas, tratadas y resueltas, lo cual propició un aseguramiento de la calidad del software.

Recomendaciones

Una vez cumplidos los objetivos del presente trabajo de diploma se recomienda:

- Implementar nuevas funcionalidades que faciliten y permitan la búsqueda y recuperación de la información en los medios de prensa, tales como:
 - Realizar búsqueda de galerías temáticas, audio, vídeo y archivos de texto.
 - Permitir la configuración de las interfaces de administración.
- Evaluar la incorporación de una capa de servicio web que posibilite el acceso al sistema por terceros.

Bibliografía Referenciada

- 1- CASTILLO BARRAGAN, Carmen. *Medios masivos de comunicación y su influencia en la educación*. [Disponible en: <http://www.odiseo.com.mx/bitacora-educativa/medios-masivos-comunicacion-su-influencia-educacion>]
- 2- NÚÑEZ, Dr. C. Israel. NÚÑEZ, Ing. Yiny. *Propuesta de clasificación de las herramientas - software para la gestión del conocimiento*. S.I.: © ECIMED, 2005.
- 3- VÁZQUEZ ACOSTA, Lic. Manuel. *Programa de Informatización de la Prensa*. Cuba: s.n., 2009.
- 4- *Programa de Informatización de la Prensa*. UCI. 2009. 1, 6, 7, 8,9, 10, 11.
- 5- Recuperación de Información en la Web, Universidad de La Salle - Programa de Sistemas de Información y Documentación. [Disponible en: <http://sistemasrecuperaciondeinformacion.blogspot.com/2010/09/recuperacion-de-informacion-en-la-web.html>]
- 6- JORQUERA, Marcos. *Administración de servicios de Internet, publicaciones Universidad de Alicante*. Alicante, Publicaciones Universidad de Alicante, 2008. ISBN: 978-84-7908-989-4.
- 7- DÍAZ CECA, Helena M^a. ZAPATA MORALES, Fátima M^a. *Buscadores de Internet*. [Disponible en: <http://www.slideshare.net/helenadiaz/buscadores-en-internet-2718947>]
- 8- *Arquitectura de un buscador*. 2007. [Disponible en: http://buscadores.fullblog.com.ar/arquitectura_de_un_buscadore_531191953898.html]
- 9- *Metabuscadore y multibuscadore*. Buscador.us. [Disponible en: http://www.buscador.us/internet/paginas/web/metabuscadore_y_multibuscadore/]
- 10- SOBRINO, Angel. *Metabuscadore: concepto de uso, ventajas e inconveniente*. [Disponible en: <http://iso-go.es/2010/09/metabuscadore-concepto-de-uso-ventajas-e-inconveniente-2/>]
- 11- GARRUDO, Álvaro Manuel. *Metabuscadore O-QE*. Universidad Pontificia Comillas. Madrid, 2006. 146 p.
- 12- *Centro de prensa de Ixquick*. [Disponible en: <https://www.ixquick.com/esp/press-center-company-background.html>]

- 13- SELBERG, Erick. *The MetaCrawler Architecture for Resource Aggregation on the Web*. Department of Computer Science and Engineering University of Washington. [Disponible en: <http://www.cs.washington.edu/research/metacrawler/>]
- 14- EDO DEL MORA, I Isabel. FERNÁNDEZ LLORIA, Susana. MORAGA BARRERO, Patricia. MOYA FERNÁNDEZ, Gloria. PÉREZ VÁZQUEZ, Carmen. *Estudio del metabuscador KartOO*. Licenciatura en Documentación Facultad de Informática. 2004.
- 15- *Alfresco. Sistema de gestión de contenido (CMS) empresarial de código abierto desarrollado por Alfresco*. 2010. [Disponible en: <http://www.alfresco.com/es/>]
- 16- *Symfony Framework PHP orientado a objetos*. [Disponible en: <http://www.maestrosdelweb.com/editorial/curso-symfony2-introduccion-instalacion/>]
- 17- *Introducción a los framework de desarrollo web*. Centro de Ideoinformática. Ciudad Habana: Universidad de las Ciencias Informáticas. 2012.
- 18- *Características de PostgreSQL*. 2010. [Disponible en: <http://www.postgresql-es.org/>]
- 19- LAMARCA LAPUENTE, María Jesús. *XML*. [Disponible en: <http://www.hipertexto.info/documentos/xml.htm>]
- 20- *Introducción, definición y evolución de PHP*. [Disponible en: <http://php.ciberaula.com/articulo/introduccion.php>]
- 21- PÉREZ EGUÍLUZ, Javier. *Introducción a JavaScript*. [Disponible en: <http://www.librosweb.es/javascript/>]
- 22- MARTIN, Pedro Rufo. *ASPtutor*. 2010. [Disponible en: <http://www.asptutor.com/zip/curcss.zip>]
- 23- LEÓN, Eduardo. *Tutorial Visual Paradigm for UML*. [Disponible en: <http://es.scribd.com/doc/36636137/Tutorial-Visual-Paradigm>]
- 24- *NetBeans*. Netbeans Corporation. 2010. [Disponible en: <http://netbeans.org/>]
- 25- FERRI-BENEDETTI. *El servidor web gratuito más popular*. [Disponible en: <http://apache-http-server.softonic.com/>]
- 26- REYES ROMELI, Walter Iván. *Optimización de la Web del I.T.M, Aplicación Servidor Web*. 2007.

- [Disponible en: <http://es.scribd.com/doc/52208534/29/CARACTERISTICAS-Y-VENTAJAS-DEL-APACHE>]
- 27- NAVARRO MARSET, Rafael. *Rest vs Servicios Web*. ELP-DSIC-UPV Modelado, Diseño e Implementación de Servicios Web. 2006-07. 9 p.
- 28- MARTIN, Pedro Rufo. *ASPtutor*. 2011. [Disponible en: <http://www.asptutor.com/zip/cbhtml.pdf>]
- 29- *Computación, Departamento de Sistemas Informáticos y Rational Unified Process (RUP)*. Universidad Politécnica de Valencia. : © P.Letelier.
- 30- GUZMÁN JIMÉNEZ, Jesús Ricardo. *Metodologías de Desarrollo Ágil*. [Disponible en: http://www.academia.edu/1740166/Metodologias_de_desarrollo_agil]
- 31- HERNÁNDEZ, Yulainne. *Configuración de la Metodología OpenUP*. Ciudad Habana: Centro Ideoinformática, 2012.
- 32- SOSA LÓPEZ, Dailyn. *Sistema para el control del uso de software educativo*. Biblioteca virtual de Derecho, Economía y Ciencias Sociales. [Disponible en: <http://www.eumed.net/libros-gratis/2009c/585/Descripcion%20del%20modelo%20del%20dominio.htm>]
- 33- LARMAN, Craig. *UML y Patrones*. Introducción al análisis y diseño orientado a objetos.
- 34- Asignatura Ingeniería de Software II. Tema 1. Conferencia No. 1: Patrones del Diseño. Ciudad de La Habana. Cuba. Curso 2006-2007.
- 35- SOMMERVILLE, Ian. *Ingeniería de Software*. Séptima edición. Madrid. 2005
- 36- CASAÑAS, María Elena. *¿Qué es software libre? Algunos conceptos técnicos*. [Disponible en: <http://www.casanas.com.ar>]
- 37- BLANK, Isabel. *Pruebas funcionales*. Asignatura Software II. Mayo 2005.
- 38- VISCONTI, Marcello. *Ingeniería de Software Avanzada*. Universidad Técnica Federico Santa María. Departamento de Informática.

Bibliografía consultada

ABBRUZZESE, Tec. Claudio Guillermo. *La fotografía como documento de archivo*. Instituto Superior de Formación Docente y Técnica No. 8. Ciudad de La Plata, Argentina.

ACOSTA, Lic. Manuel. *Desarrollo de una arquitectura rectora para el Programa de Informatización de la Prensa*. Cuba: s.n., 2008.

ÁVILA, Yusniel. LLANES, Néstor. *Sistema Automatizado para la gestión de información en rehabilitación*. 2008

Base Tecnológica. Universidad de las Ciencias Informáticas. Centro CIDI. 2012

COMAS, Anabell. *METADATOS Y SITIOS WEB*. 10, s.l.: © Coordinación de Publicaciones Digitales. DGSCA-UNAM, 2006, Vol. 7. ISSN: 1067-6079.

FERNÁNDEZ DE BOMBADILLA, Luis. *Metabusador REQEX*. Universidad Pontificia Comillas. Madrid. 2009. 138 p.

GRANOLLERS, T. LORÉS, J. PERDRIX, F. *Modelo de proceso de la Ingeniería de la Usabilidad. Integración de la Ingeniería de Software y la de la Usabilidad*. Universidad de Lleida. Madrid.

HERNÁNDEZ MURILLO, Ronny. SANES PÉREZ, Iván. *Diseño e implementación de un metabuscador con acceso vocal para personas discapacitadas con funcionamiento del habla*. Corporación universitaria Rafael Nuñez. Facultad de ingeniería de sistemas. Cartagena de Indias. 2004

JOSKOWICZ, Ing. José. *Reglas y Prácticas en extreme programing*. España: s.n., 2008.

LAMARCA LAPUENTE, María Jesús. *Metadatos para imágenes*. [Disponible en: http://www.hipertexto.info/documentos/metad_imag.htm]

MARTÍNEZ FAJARDO, Alexander. MÉNDEZ PÉREZ, Javier. *Sistema de gestión de materiales audiovisuales*. Universidad de las Ciencias Informáticas. Santiago de Cuba. 2010.

MENA MUGICA, Mayra. *Gestión Documental y organización de archivos*. La Habana, Editorial Félix Varela, 2005. 51

MERLINO SANTESTEBAN, Cristian. *Acceso y recuperación de información en la World Wide Web*. Universidad Nacional de Mar del Plata. Mar del Plata, Argentina. 2001.

REASCOS, Ing. Irving. *Estudio de la arquitectura de Software*. Universidad Técnica del Norte. Ecuador. 2010.

SÁNCHEZ VIGIL, Juan Miguel. *La fotografía como documento en el siglo XXI*. Facultad de Ciencias de la Información. 2001.

Web2.0 en la Gestión de la Información Turística. Facultad de Turismo Universidad Nacional del Comahue. [Disponible en: <http://web20gi.wordpress.com/2011/09/01/%C2%BFque-son-los-metabuscadore/>]

Glosario de términos

A

Alfresco: Gestor de Contenido Empresarial, alternativa libre y de código abierto que permite desarrollar proyectos de contenido empresarial.

Artefacto: Pieza de información tangible que es creada, modificada y usada por los trabajadores al realizar actividades; representa un área de responsabilidad, y es candidata a ser tenida en cuenta para el control de la configuración. Un artefacto puede ser un modelo, un elemento de un modelo, o un documento.

C

CASE: Acrónimo en inglés de Computer Aided Software Engineering (Ingeniería de Software Asistida por Computadora).

CSS: Acrónimo en inglés de Cascading Style Sheets (Hojas de Estilo en Cascada). Es un conjunto de instrucciones escritas en HTML, que definen las apariencias de una página web con el objetivo de que sus estilos aparezcan.

E

ECM: Acrónimo en inglés de Enterprise Content Management (Gestión de Contenido Empresarial).

Ecumene Pyxel: Archivo de Imágenes para el Programa de Informatización de la Prensa.

F

Framework: Es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado.

G

GIF: Acrónimo en inglés de Graphics Interchange Format (Formato de Intercambio de Gráficos). Es una extensión y formato de imagen.

H

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

HTTP: Acrónimo en inglés de HyperText Transfer Protocol, protocolo usado para acceder a la Web. Se encarga de procesar y dar respuestas a las peticiones, para visualizar una página web.

I

IBM: Acrónimo en inglés de International Business Machines.

IDE: Acrónimo en inglés de Integrated Development Environment (Entorno de Desarrollo Integrado), es una aplicación compuesta por un conjunto de herramientas útiles para un programador.

IPTC: Son conocidos como "encabezados IPTC" y utilizados para fotografías y noticias.

J

JPEG: Acrónimo en inglés de Joint Photographic Experts Group, nombre del comité que desarrolló el estándar JPEG para la compresión de imágenes. Es un algoritmo diseñado para comprimir imágenes fijas con 24 bits de profundidad o en escala de grises.

M

MCM: Medios de Comunicación Masiva.

MIME: Acrónimo en inglés de Multipurpose Internet Mail Extensions (Extensiones de Correo Internet Multipropósito). Serie de especificaciones dirigidas al intercambio transparente de todo tipo de archivos a través de Internet.

Módulo: En programación, es un software que agrupa un conjunto de subprogramas y estructura de datos. Unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea.

N

No conformidades: Problema detectado en un artefacto según:

- Error con respecto a lo definido en artefactos anteriores y/o en lo pactado con el cliente.
- No concordancia con normas internacionales que deben ser cumplidas por el artefacto.
- Insatisfacción del cliente con el resultado final de un elemento de configuración según lo pactado con anterioridad en el proyecto. La resolución de una no conformidad siempre genera una orden de trabajo para el proyecto.

P

PHP: Acrónimo en inglés de Hypertext Preprocessor, lenguaje interpretado para el desarrollo de páginas web dinámicas del lado del servidor.

R

RUP: Acrónimo en inglés de Rational Unified Process (Proceso Unificado de Rational o Metodología RUP), es una metodología de desarrollo de software.

Requisitos funcionales: Son las capacidades o condiciones que el sistema debe cumplir.

Requisitos no funcionales: Son las propiedades o cualidades que el producto debe tener.

S

Servidor: Es una computadora central en un sistema de red que provee servicios a otras computadoras.

SGBD: Sistema Gestor de Bases de Datos.

SGC: Sistema de Gestión de Contenidos.

SGD: Sistema de Gestión Documental.

T

TCP/IP: Acrónimo en inglés de Transfer Control Protocol/ Internet Protocol. Es un protocolo que utiliza internet para comunicarse.

TIC: Tecnologías de la Información y las Comunicaciones.

U

UCI: Universidad de las Ciencias Informáticas.

UML: Acrónimo en inglés de Unified Modeling Language (Lenguaje Unificado de Modelado).

URL: Acrónimo en inglés de Uniform Resource Locator (Localizador Uniforme de Recursos). Es la forma de organizar la información en la Web.

W

W3C: Es el Consorcio World Wide Web, una comunidad internacional donde las organizaciones miembro, personal a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares web.

X

XML: Acrónimo en inglés de Extensible Markup Language (Lenguaje de Etiquetado Extensible).

Z

Zend Framework: Es una librería de componentes escritos en PHP5, para facilitar el desarrollo de aplicaciones web.

Anexos

Anexo No. 1: Entrevista realizada al cliente.

La presente entrevista persigue realizar el levantamiento de requisitos y lograr un mayor entendimiento de las características que debe poseer la herramienta a implementar. De esta manera se prevé un mejor acercamiento con el cliente y una correcta visión de la lógica del negocio.

Entrevistado: _____

Cargo: _____

Fecha_____

Descripción del Sistema

1. ¿Qué objetos del mundo real se deben gestionar en la aplicación?
2. ¿Cómo se va a desarrollar el flujo de eventos en el sistema?
3. ¿Cuáles son los tipos de contenidos que se deben mostrar en la aplicación?
4. ¿Cómo se relacionan las clases del sistema?

Roles y funcionalidades.

1. ¿Qué roles intervienen y qué función desempeñan en el sistema?
2. Procedimientos a seguir para realizar las búsquedas.
3. Mecanismo a utilizar para la organización de los resultados de las búsquedas.

Anexo No. 2: Especificación del caso de uso Autenticar usuario.

Objetivo	Autenticarse en el sistema.	
Actores	Usuario Autenticado: (Inicia) Se autentica en el sistema. Administrador: (Inicia) Se autentica en el sistema.	
Resumen	El CU permite autenticarse en el sistema a partir de un nombre de usuario y una contraseña.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones		
Postcondiciones	El usuario se autenticó en el sistema.	
Flujo de eventos		
Flujo básico Autenticar usuario		
	Actor	Sistema
1.	Abre su navegador y escribe la dirección URL del Metabusador.	Muestra la interfaz de inicio de la aplicación
2.	Selecciona el icono que representa la funcionalidad de autenticación.	Muestra la interfaz de autenticación, en la parte central muestra un formulario con los siguientes campos: - Nombre de usuario - Contraseña
3.	Introduce el nombre de usuario y la contraseña. Presiona el botón Acceder.	Valida que los datos sean correctos. Muestra la interfaz de inicio según los privilegios del usuario.
4.		Termina el caso de uso.

Flujos alternos		
Nº 1 Los datos son incorrectos.		
	Actor	Sistema
3.1	<p>Introduce un nombre de usuario o/y una contraseña que no se encuentra registrado en el sistema.</p> <p>Presiona el botón Acceder.</p>	<p>Muestra el siguiente mensaje:</p> <p>“El nombre de usuario y/o contraseña son incorrectos.”</p>
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 9: Especificación del caso de uso Autenticar usuario.

Anexo No. 3: Especificación del caso de uso Gestionar usuario.

Objetivo	Gestionar las cuentas de los usuarios.	
Actores	Administrador: (Inicia) Adiciona, modifica y elimina la cuenta de un usuario en el sistema.	
Resumen	El CU permite gestionar una cuenta de usuario a través de las siguientes acciones: adicionar, editar y eliminar cuenta de usuario.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha sido autenticado. El usuario tiene permisos de administración.	
Postcondiciones	El usuario adicionó un nuevo usuario. El usuario editó un usuario. El usuario eliminó un usuario.	
Flujo de eventos		
Flujo básico Gestionar usuario		
	Actor	Sistema
1.	Autenticado en el sistema y ubicado en su espacio de trabajo selecciona la opción Gestionar.	Muestra la interfaz de administración.
2.	Selecciona el icono que representa la funcionalidad Gestionar usuario.	Muestra la interfaz Gestionar usuario. Muestra las siguientes opciones: <ul style="list-style-type: none"> - Adicionar un usuario. Ver Sección 1: “Adicionar usuario”. - Editar un usuario. Ver Sección 2: “Editar usuario”. - Eliminar un usuario. Ver Sección 3:

		"Eliminar usuario".
3.		Termina el caso de uso.
Sección 1: "Adicionar usuario"		
Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Adicionar usuario encontrada en la parte superior derecha de la interfaz.	<p>Muestra la interfaz Adicionar usuario, en la parte central muestra un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - Apellidos - Correo - Carnet de identidad - Ciudad - País - Clave - Confirmar clave - Sexo - Icono - Estado
4.	<p>Introduce los datos en los campos del formulario.</p> <p>Presiona el botón Adicionar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p> <p>Muestra una interfaz con la cuenta de usuario</p>

		y el siguiente mensaje: "Usuario creado con éxito."
5.		Termina el caso de uso.
Flujos alternos		
Nº 1 Datos incorrectos.		
	Actor	Sistema
4.1	<p>Introduce cadenas no compatibles con el tipo de datos o con la longitud predefinida.</p> <p>Presiona el botón Adicionar.</p>	<p>Muestra un mensaje en dependencia del error.</p> <p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se incumple con la estructura de una dirección de correo electrónico, muestra un mensaje indicando que la dirección introducida es incorrecta.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.</p>
Nº 2 Campos vacíos.		
	Actor	Sistema
4.1	<p>Ha dejado campos vacíos a la hora de adicionar una cuenta de usuario.</p> <p>Presiona el botón Adicionar.</p>	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.</p>

Sección 2: "Editar usuario"		
Flujo básico		
	Actor	Sistema
3.	<p>Selecciona la opción Editar-Eliminar ubicada en la parte derecha del usuario a modificar.</p>	<p>Muestra la interfaz Editar-Eliminar usuario, en la parte central visualiza un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - Apellidos - Correo - Carnet de identidad - Ciudad - País - Clave - Confirmar clave - Sexo - Icono - Estado
4.	<p>Introduce los datos en los campos que se deseen editar.</p> <p>Presiona el botón Editar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p> <p>Muestra una interfaz con la cuenta de usuario y el siguiente mensaje:</p> <p>"Usuario editado con éxito."</p>

5.		Termina el caso de uso.
Flujos alternos		
Nº 1 Datos incorrectos.		
	Actor	Sistema
4.1	<p>Introduce cadenas no compatibles con el tipo de datos o con la longitud predefinida.</p> <p>Presiona el botón Editar.</p>	<p>Muestra un mensaje en dependencia del error.</p> <p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se incumple con la estructura de una dirección de correo electrónico, muestra un mensaje indicando que la dirección introducida es incorrecta.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.</p>
Nº 2 Campos vacíos.		
	Actor	Sistema
4.1	<p>Ha dejado vacío los campos obligatorios.</p> <p>Presiona el botón Editar.</p>	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.</p>
Sección 3: "Eliminar usuario"		
Flujo básico		

	Actor	Sistema
3.	Selecciona la opción Editar-Eliminar ubicada en la parte derecha del proveedor a modificar.	Muestra la interfaz Editar-Eliminar usuario.
4.	Presiona el botón Eliminar.	Muestra una ventana de confirmación.
5.	Selecciona la opción Sí.	Elimina la cuenta de usuario de la base de datos.
6.		Termina el caso de uso.
Flujos alternos		
Nº 1 Selección contraria.		
	Actor	Sistema
5.1	Selecciona la opción No.	Muestra la interfaz de gestión de usuario.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 10: Especificación del caso de uso Gestionar usuario.

Anexo No. 4: Especificación del caso de uso Solicitar cuenta de usuario.

Objetivo	Solicitar una cuenta de usuario.	
Actores	Usuario Anónimo: (Inicia) Solicita una cuenta de usuario.	
Resumen	El CU permite que una persona pueda solicitar una cuenta de usuario, a partir del envío de sus datos al personal encargado del funcionamiento del sistema.	
Complejidad	Media	
Prioridad	Media	
Precondiciones		
Postcondiciones	El usuario solicitó una cuenta de usuario.	
Flujo de eventos		
Flujo básico Solicitar una cuenta de usuario		
	Actor	Sistema
1.	Abre su navegador y escribe la dirección URL del Metabusador.	Muestra la interfaz de inicio de la aplicación, en la parte superior izquierda muestra la opción Registrarse.

2.	<p>Selecciona la opción Registrarse.</p>	<p>Muestra la interfaz correspondiente a la opción Registrarse, en el centro de la página muestra un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - Apellidos - Correo - Carnet de identidad - Ciudad - País - Clave - Confirmar clave - Sexo
3.	<p>Introduce los datos correspondientes para solicitar una cuenta de usuario en el sistema.</p> <p>Presiona el botón Registrarse.</p>	<p>Valida los datos sean correctos.</p> <p>Envía un correo al encargado del funcionamiento del sistema con los datos introducidos.</p>
4.		<p>Termina el caso de uso.</p>
<p>Flujos alternos</p>		
<p>Nº 1 "Datos incorrectos"</p>		
	<p>Actor</p>	<p>Sistema</p>
3.1	<p>Introduce cadenas no compatibles con el tipo de datos o con la longitud predefinida.</p>	<p>Muestra un mensaje en dependencia del error.</p>

	Presiona el botón Registrarse.	<p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se incumple con la estructura de una dirección de correo electrónico, muestra un mensaje indicando que la dirección introducida es incorrecta.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.</p>
Nº 2 Campos vacíos.		
	Actor	Sistema
3.1	Ha dejado campos vacíos a la hora de introducir los datos.	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.</p>
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 11: Especificación del caso de uso Solicitar cuenta de usuario.

Anexo No. 5: Especificación del caso de uso Gestionar proveedores.

Objetivo	Gestionar los proveedores del sistema.	
Actores	Administrador: (Inicia) Adiciona, modifica, elimina y muestra los proveedores en el sistema.	
Resumen	El CU permite gestionar los proveedores del sistema a través de las siguientes acciones: mostrar, adicionar, editar y eliminar proveedor.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha sido autenticado. El usuario tiene permisos de administración.	
Postcondiciones	El usuario adicionó un proveedor. El usuario editó un proveedor. El usuario eliminó un proveedor. El usuario mostró un proveedor.	
Flujo de eventos		
Flujo básico Gestionar proveedores		
	Actor	Sistema
1.	Autenticado en el sistema y ubicado en su espacio de trabajo selecciona la opción Gestionar.	Muestra la interfaz de administración
2.	Selecciona el icono que representa la funcionalidad Gestionar proveedores.	Muestra la interfaz Gestionar proveedores. Muestra las siguientes opciones: - Adicionar proveedor. Ver Sección 1:

		<p>“Adicionar proveedor”.</p> <ul style="list-style-type: none"> - Editar proveedor. Ver Sección 2: “Editar proveedor”. - Eliminar proveedor. Ver Sección 3: “Eliminar proveedor”. - Mostrar proveedor. Ver Sección 4: “Mostrar proveedor”.
3.		Termina el caso de uso.
Sección 1: “Adicionar proveedor”		
Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Adicionar proveedor encontrada en la parte superior derecha de la interfaz.	<p>Muestra la interfaz Adicionar proveedor, en la parte central visualiza un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - URL - Usuario - Clave - Avatar - Estado - Path - Relevancia - Sexo

		- Icono
4.	<p>Introduce los datos correspondientes para adicionar un proveedor al sistema.</p> <p>Presiona el botón Adicionar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p> <p>Muestra una interfaz con el proveedor añadido.</p>
5.		Termina el caso de uso.
Flujos alternos		
Nº 1 “Datos incorrectos”		
	Actor	Sistema
4.1	<p>Introduce cadenas no compatibles con el tipo de datos.</p> <p>Presiona el botón Adicionar.</p>	<p>Muestra un mensaje en dependencia del error.</p> <p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo</p>
Nº 2 Campos vacíos.		
	Actor	Sistema
4.1	Ha dejado campos vacíos a la hora de adicionar un proveedor.	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer</p>

		vacío.
Sección 2: "Editar proveedor"		
Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Editar-Eliminar proveedor encontrada en la parte superior derecha de la interfaz.	<p>Muestra la interfaz Editar-Eliminar proveedor, en la parte central visualiza un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - URL - Usuario - Clave - Avatar - Estado - Path - Relevancia - Sexo - Icono
4.	<p>Introduce los datos correspondientes al proveedor que se desee editar.</p> <p>Presiona el botón Editar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p> <p>Muestra una interfaz con el proveedor añadido.</p>
5.		Termina el caso de uso.

Flujos alternos		
Nº 1 “Datos incorrectos”		
	Actor	Sistema
4.1	<p>Introduce cadenas no compatibles con el tipo de datos.</p> <p>Presiona el botón Editar.</p>	<p>Muestra un mensaje en dependencia del error.</p> <p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.</p>
Nº 2 Campos vacíos.		
	Actor	Sistema
4.1	Ha dejado vacío algún campo obligatorio.	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.</p>
Sección 3: “Eliminar proveedor”		
Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Editar-Eliminar proveedor.	Muestra la interfaz Editar-Eliminar proveedor, en la parte inferior derecha muestra el botón

		Eliminar.
4.	Presiona el botón Eliminar.	Muestra una ventana de confirmación.
5.	Selecciona la opción Sí.	Elimina el proveedor de la base de datos del sistema. Muestra la interfaz de gestión de proveedores.
6.		Termina el caso de uso.
Flujos alternos		
Nº 1 No selección.		
	Actor	Sistema
5.1	Selecciona la opción No.	Muestra la interfaz de gestión de proveedores.
Sección 4: "Mostrar proveedor"		
Flujo básico		
	Actor	Sistema
3.	Selecciona el proveedor a mostrar.	Muestra una interfaz con los datos relacionados al proveedor escogido.
4.		Termina el caso de uso.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 12: Especificación del caso de uso Gestionar proveedores.

Anexo No. 6: Especificación del caso de uso Chequear proveedor.

Objetivo	Chequear los proveedores del sistema.	
Actores	Administrador: (Inicia) Chequea un proveedor.	
Resumen	El CU permite conocer el estado de un proveedor (habilitado/no habilitado).	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	<p>El usuario ha sido autenticado.</p> <p>El usuario tiene permisos de administración.</p> <p>El proveedor a chequear está en el sistema.</p>	
Postcondiciones	El proveedor fue chequeado.	
Flujo de eventos		
Flujo básico Chequear proveedor		
	Actor	Sistema
1.	Autenticado en el sistema y ubicado en su espacio de trabajo selecciona la opción Gestionar.	Muestra la interfaz de administración.
2.	Selecciona el icono que representa la funcionalidad Gestionar proveedores.	Muestra la interfaz Gestionar proveedores, en la parte derecha de cada proveedor muestra la opción Comprobar
3.	Selecciona la opción Comprobar correspondiente al proveedor requerido.	<p>Si se encuentra el proveedor disponible muestra el siguiente mensaje:</p> <p>“El proveedor solicitado está disponible.”</p> <p>En caso contrario:</p> <p>“El proveedor solicitado no está disponible”.</p>
4.		Termina el caso de uso.

Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 13: Especificación del caso de uso Chequear proveedor.

Anexo No. 7: Especificación del caso de uso Salvar criterio de búsqueda.

Objetivo	Salvar los resultados obtenidos de la búsqueda en el sistema.	
Actores	Usuario Autenticado: (Inicia) Salva el resultado de la búsqueda.	
Resumen	El CU permite salvar un criterio introducido por un usuario luego de realizar una búsqueda.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	El usuario ha sido autenticado. El usuario realiza una búsqueda.	
Postcondiciones	El usuario salvó el criterio de búsqueda.	
Flujo de eventos		
Flujo básico Salvar criterio de búsqueda		
	Actor	Sistema
1.	Escoge la opción Salvar criterio de búsqueda, ubicada en la parte derecha de la interfaz.	Salva el criterio de búsqueda.
2.		Termina el caso de uso.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 14: Especificación del caso de uso Salvar criterio de búsqueda.

Anexo No. 8: Especificación del caso de uso Gestionar taxonomía.

Objetivo	Gestionar las taxonomías en el sistema.	
Actores	Administrador: (Inicia) Adiciona, modifica y elimina las taxonomías en el sistema.	
Resumen	El CU permite gestionar las taxonomías del sistema a partir de las siguientes acciones: adicionar, editar y eliminar taxonomía.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha sido autenticado. El usuario tiene permisos de administración.	
Postcondiciones	El usuario adicionó una taxonomía. El usuario editó una taxonomía. El usuario eliminó una taxonomía.	
Flujo de eventos		
Flujo básico Gestionar taxonomía		
	Actor	Sistema
1.	Autenticado en el sistema y ubicado en su espacio de trabajo selecciona la opción Gestionar.	Muestra la interfaz de administración.
2.	Selecciona el icono que representa la funcionalidad Gestionar taxonomía.	Muestra la interfaz Gestionar taxonomía. Muestra las siguientes opciones: - Adicionar taxonomía. Ver Sección 1: “Adicionar taxonomía”. - Editar taxonomía. Ver Sección 2:

		<p>“Editar taxonomía”.</p> <ul style="list-style-type: none"> - Eliminar taxonomía. Ver Sección 3: “Eliminar taxonomía”.
3.		Termina el caso de uso.
Sección 1: “Adicionar taxonomía”		
Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Adicionar taxonomía encontrada en la parte superior derecha de la interfaz.	<p>Muestra la interfaz Adicionar taxonomía, en la parte central visualiza un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - Tipo
4.	<p>Introduce los datos correspondientes a la taxonomía a adicionar.</p> <p>Presiona el botón Adicionar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p> <p>Muestra una interfaz con la taxonomía añadida.</p>
5.		Termina el caso de uso.
Flujos alternos		
Nº 1 Campos vacíos		
	Actor	Sistema
4.1	<p>No introduce todos los datos.</p> <p>Presiona el botón Adicionar.</p>	<p>Muestra un mensaje de error.</p> <p>Si se encuentra algún campo vacío, se</p>

		muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.
Nº 1 “Datos incorrectos”		
	Actor	Sistema
4.1	<p>Introduce cadenas no compatibles con el tipo de datos.</p> <p>Presiona el botón Adicionar.</p>	<p>Muestra un mensaje en dependencia del error.</p> <p>Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo.</p> <p>Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.</p>
Sección 2: “Editar taxonomía”		
Flujo básico		
	Actor	Sistema
3.	<p>Selecciona la opción Editar-Eliminar ubicada en la parte izquierda de la taxonomía.</p>	<p>Muestra la interfaz Editar-Eliminar, en la parte central muestra un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Nombre - Tipo
4.	<p>Introduce los nuevos datos en los campos que desea editar.</p>	<p>Valida que los datos sean correctos.</p> <p>Guarda los datos.</p>

	Presiona el botón Editar.	Muestra el siguiente mensaje “La taxonomía se ha editado satisfactoriamente”.
5.		Termina el caso de uso.
Flujos alternos		
Nº 1 Campos vacíos		
	Actor	Sistema
4.1	No introduce todos los datos. Presiona el botón Editar.	Muestra un mensaje de error. Si se encuentra algún campo vacío, se muestra un mensaje de error indicando al usuario que el campo no puede permanecer vacío.
Nº 1 “Datos incorrectos”		
	Actor	Sistema
4.1	Introduce cadenas no compatibles con el tipo de datos. Presiona el botón Editar.	Muestra un mensaje en dependencia del error. Si se introdujo caracteres extraños, muestra un mensaje indicando que no se admiten caracteres extraños en ese campo. Si se introducen caracteres no compatibles con el tipo de datos especificado, muestra un mensaje indicando que solo debe introducir el tipo de datos requerido para ese campo.
Sección 3: “Eliminar taxonomía”		

Flujo básico		
	Actor	Sistema
3.	Selecciona la opción Editar-Eliminar ubicada en la parte izquierda de la taxonomía.	Muestra la interfaz Editar-Eliminar, en la parte inferior derecha muestra el botón Eliminar.
4.	Presiona el botón Eliminar.	Muestra en una ventana un mensaje de confirmación.
5.	Selecciona la opción Sí.	Muestra el siguiente mensaje: “La taxonomía se ha eliminado satisfactoriamente”
6.		Termina el caso de uso.
Flujos alternos		
Nº 1 “No selección”		
	Actor	Sistema
5.1	Selecciona la opción No.	Muestra la interfaz Gestionar Taxonomía.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 15: Especificación del caso de uso Gestionar taxonomía.

Anexo No. 9: Especificación del caso de uso Realizar búsqueda avanzada.

Objetivo	Realizar la búsqueda avanzada en el sistema.	
Actores	Usuario Anónimo: (Inicia) Realiza la búsqueda avanzada. Usuario Autenticado: (Inicia) Realiza la búsqueda avanzada. Administrador: (Inicia) Realiza la búsqueda avanzada.	
Resumen	El CU permite realizar una búsqueda en el sistema a partir de filtros para acotar los resultados.	
Complejidad	Alta	
Prioridad	Alta	
Precondiciones		
Postcondiciones	El usuario realizó la búsqueda avanzada.	
Flujo de eventos		
Flujo básico Realizar búsqueda avanzada		
	Actor	Sistema
1.	Abre su navegador y escribe la dirección URL del Metabusador.	Muestra la interfaz de inicio de la aplicación, en la que aparece en la parte inferior izquierda del campo de búsqueda, la opción Búsqueda Avanzada.
2.	Selecciona la opción Búsqueda Avanzada.	En la parte inferior de la opción Búsqueda Avanzada muestra los siguientes campos: - Autor - Lugar

		<ul style="list-style-type: none"> - Medio de prensa - Persona - Fecha de inicio - Fecha de fin - Temática - Palabra clave - Colecciones - Frase exacta
3.	<p>Introduce el criterio de búsqueda, selecciona el tipo de contenido y los filtros por los que desea acotar la búsqueda e introduce los datos.</p> <p>Presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.</p>	<p>Valida que los datos sean correctos.</p> <p>Realiza la búsqueda a partir de lo introducido por el usuario.</p> <p>Muestra los resultados.</p>
4.		Termina el caso de uso.
Flujos alternos		
Nº 1 “Resultados no encontrados”		
	Actor	Sistema
3.1	<p>Introduce el criterio de búsqueda, selecciona el tipo de contenido y los filtros por los que desea acotar la búsqueda e introduce los datos.</p> <p>Presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.</p>	<p>Muestra el siguiente mensaje:</p> <p>“No hay resultados disponibles”</p>
Nº 2 “Criterio no introducido”		

	Actor	Sistema
3.1	<p>Selecciona el tipo de contenido y los filtros por los que desea acotar la búsqueda e introduce los datos.</p> <p>Presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.</p>	<p>Muestra el mensaje de error:</p> <p>“El campo "Criterio de Búsqueda" no puede estar vacío y debe tener como mínimo 3 caracteres, para poder realizar la búsqueda.”</p>
Nº 3 “Tipo de contenido no seleccionado”		
	Actor	Sistema
3.1	<p>Introduce el criterio de búsqueda, selecciona los filtros por los que desea acotar la búsqueda e introduce los datos.</p> <p>Presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.</p>	<p>Muestra el mensaje de error:</p> <p>“Debe de elegir al menos un tipo de contenido para poder realizar la búsqueda.”</p>
Nº 4 “Filtros no utilizados”		
	Actor	Sistema
3.1	<p>Introduce el criterio de búsqueda y selecciona el tipo de contenido.</p> <p>Presiona el botón que se encuentra a la derecha del formulario para enviar los datos al sistema.</p>	<p>Valida que los datos sean correctos.</p> <p>Realiza la búsqueda a partir de lo introducido por el usuario.</p> <p>Muestra los resultados.</p>
Relaciones	CU Incluidos	Salvar criterios. Ver CU: Salvar criterio de búsqueda.
	CU Extendidos	
Requisitos no funcionales	Todos los requisitos no funcionales del sistema están presentes en este caso de uso.	

Tabla No. 16: Especificación del caso de uso Realizar búsqueda avanzada.

Anexo No. 10: Diagrama de Clases del Diseño del caso de uso Autenticar usuario.

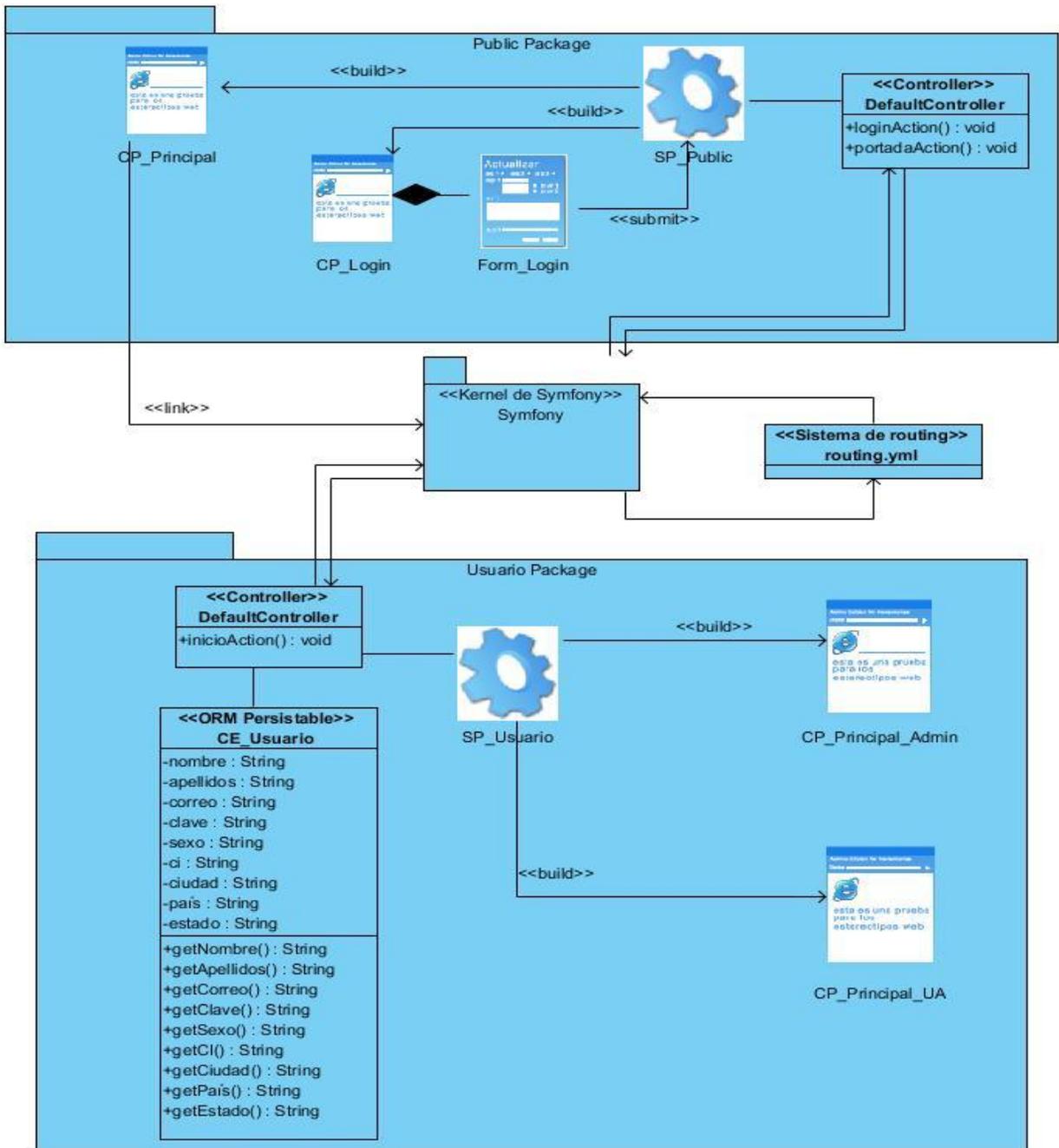


Figura No. 13: Diagrama de Clases del Diseño del caso de uso Autenticar usuario.

Anexo No. 11: Diagrama de Clases del Diseño del caso de uso Realizar búsqueda avanzada.

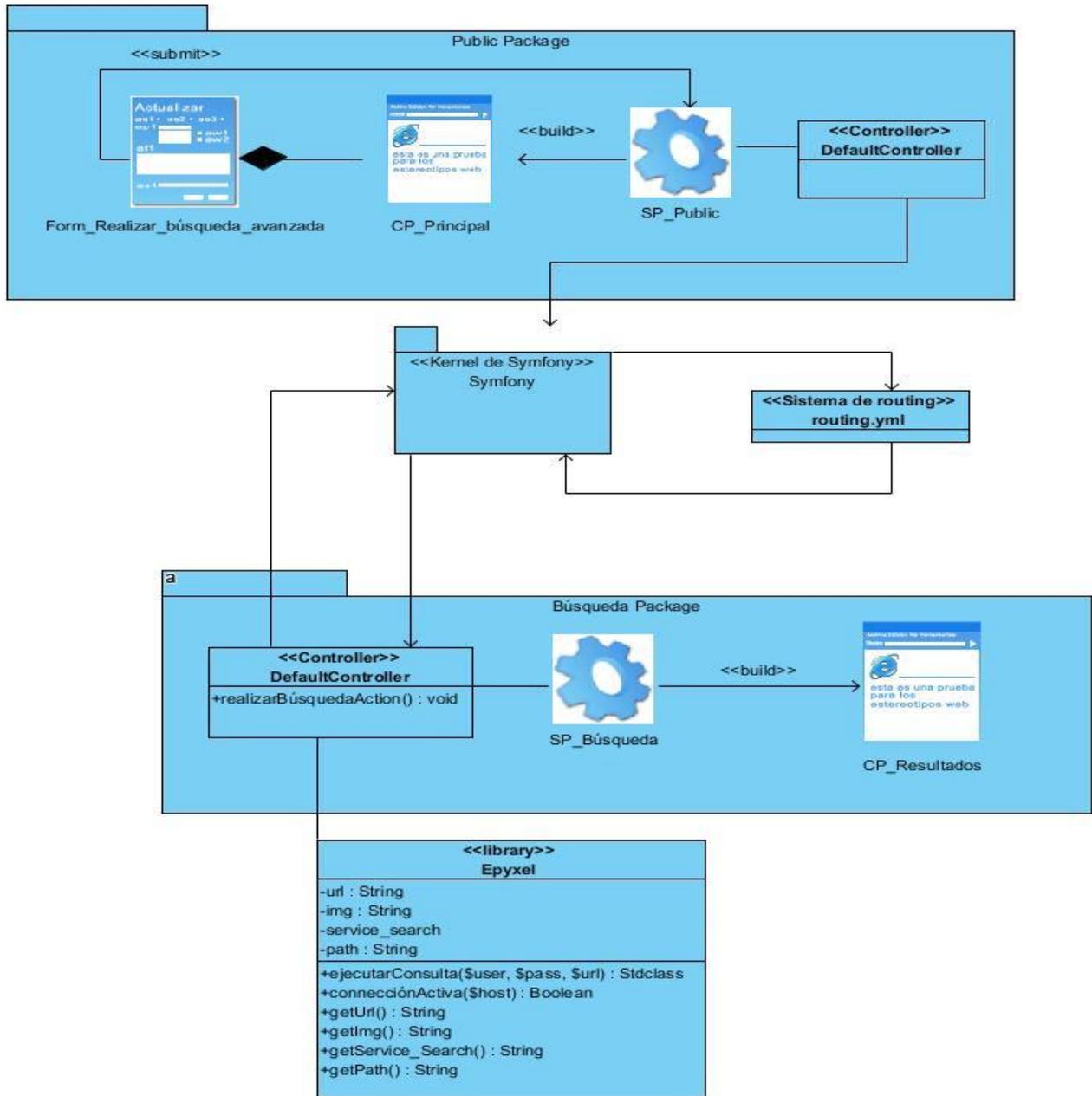


Figura No. 14: Diagrama de Clases del Diseño del caso de uso Realizar búsqueda avanzada.

Anexo No. 12: Diagrama de Clases del Diseño del caso de uso Gestionar usuario.

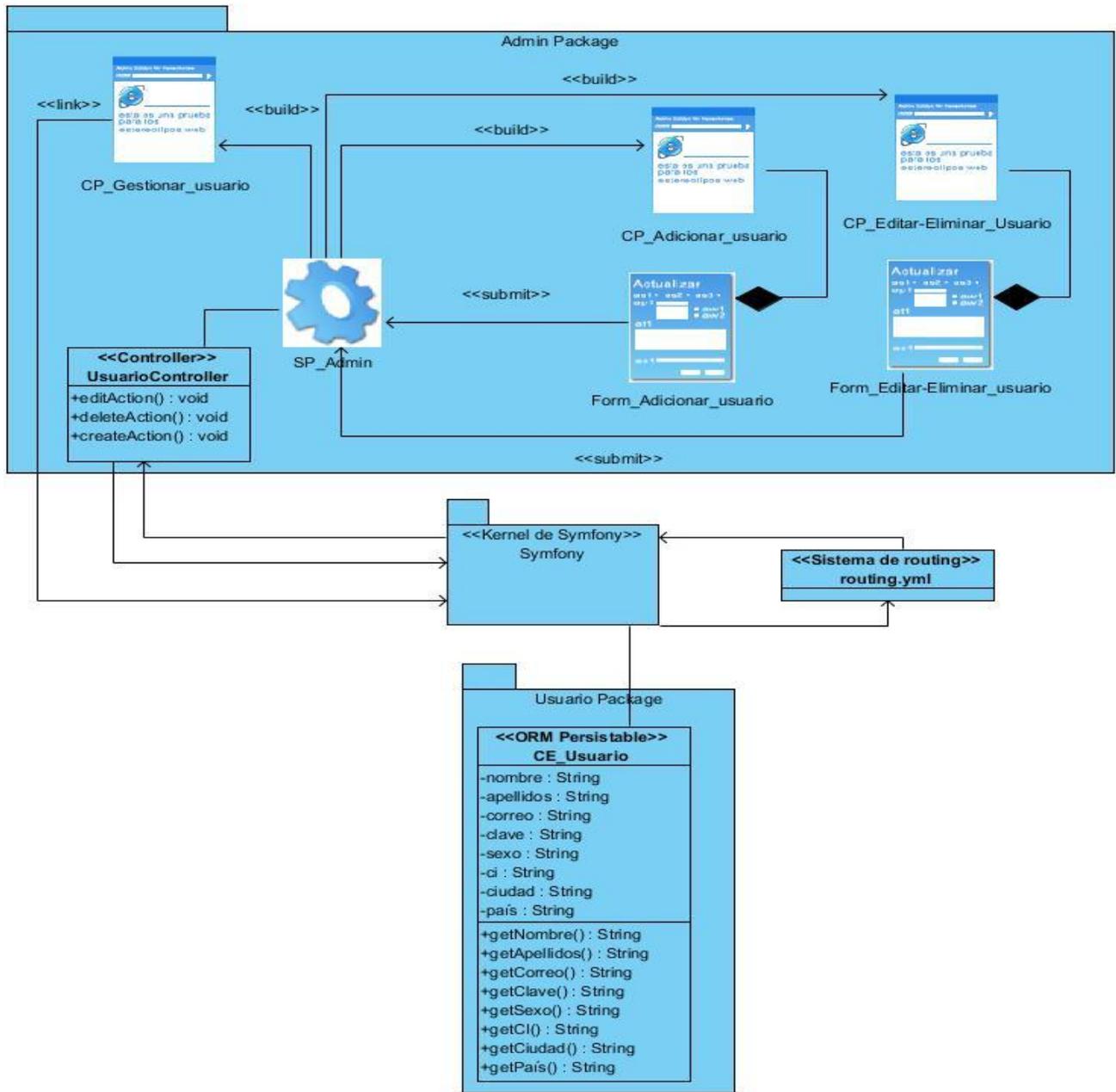


Figura No. 15: Diagrama de Clases del Diseño del caso de uso Gestionar usuario.

Anexo No. 13: Diagrama de Clases del Diseño del caso de uso Gestionar proveedor.

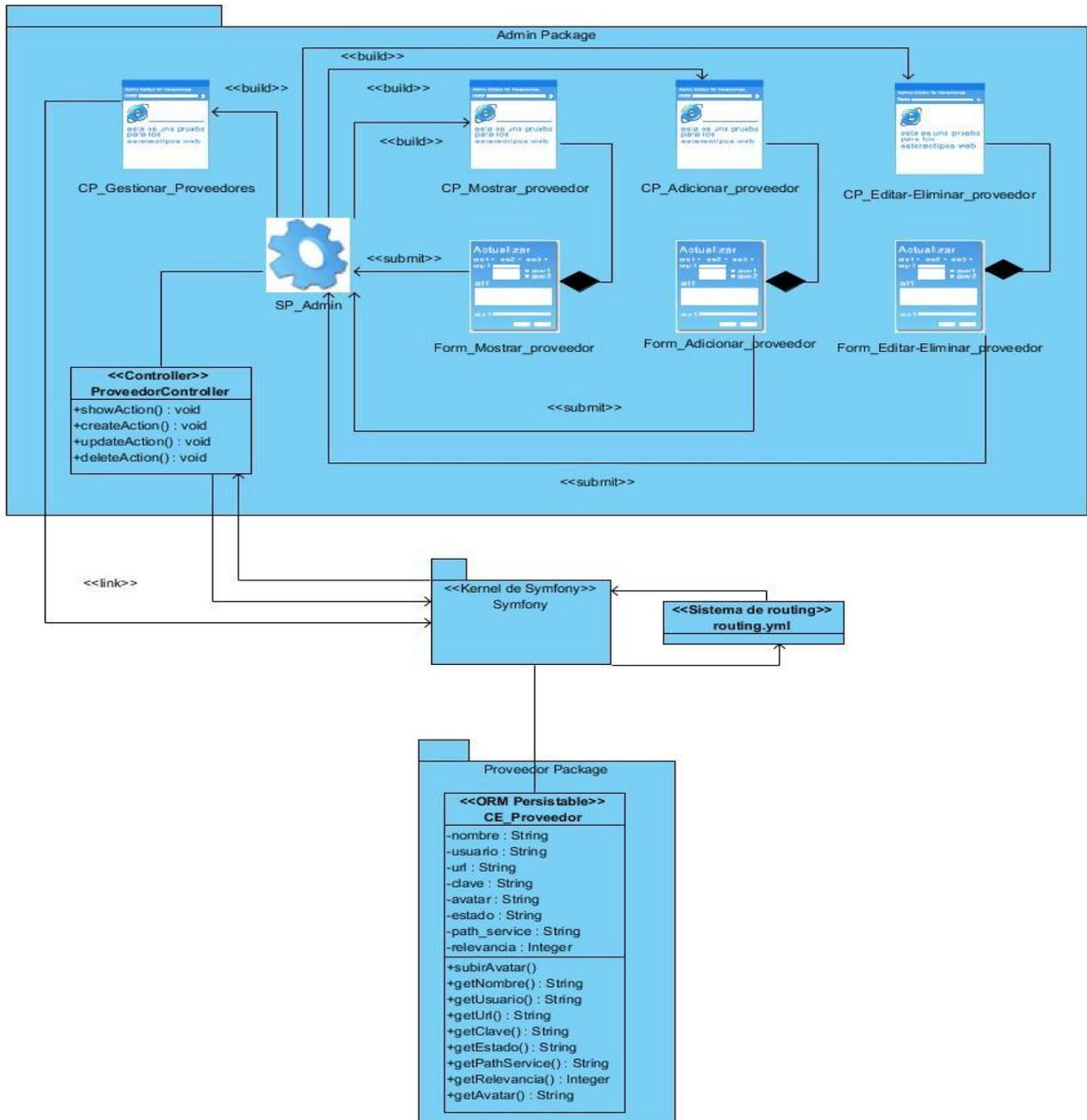


Figura No. 16: Diagrama de Clases del Diseño del caso de uso Gestionar proveedor.

Anexo No. 14: Diagrama de Clases del Diseño del caso de uso Gestionar taxonomía.

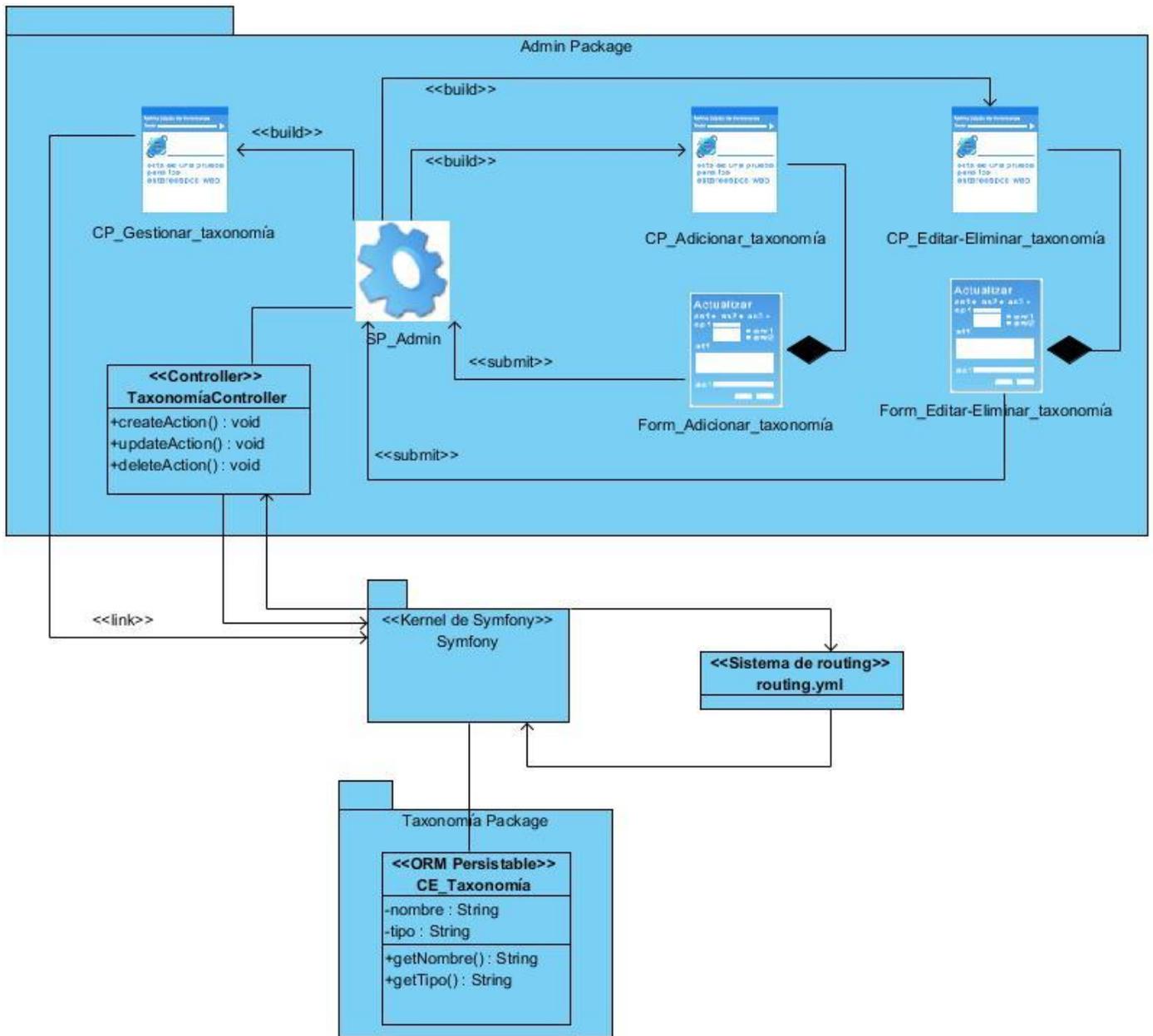


Figura No. 17: Diagrama de Clases del Diseño del caso de uso Gestionar taxonomía.

Anexo No. 15: Diagrama de Clases del Diseño del caso de uso Compartir contenidos.

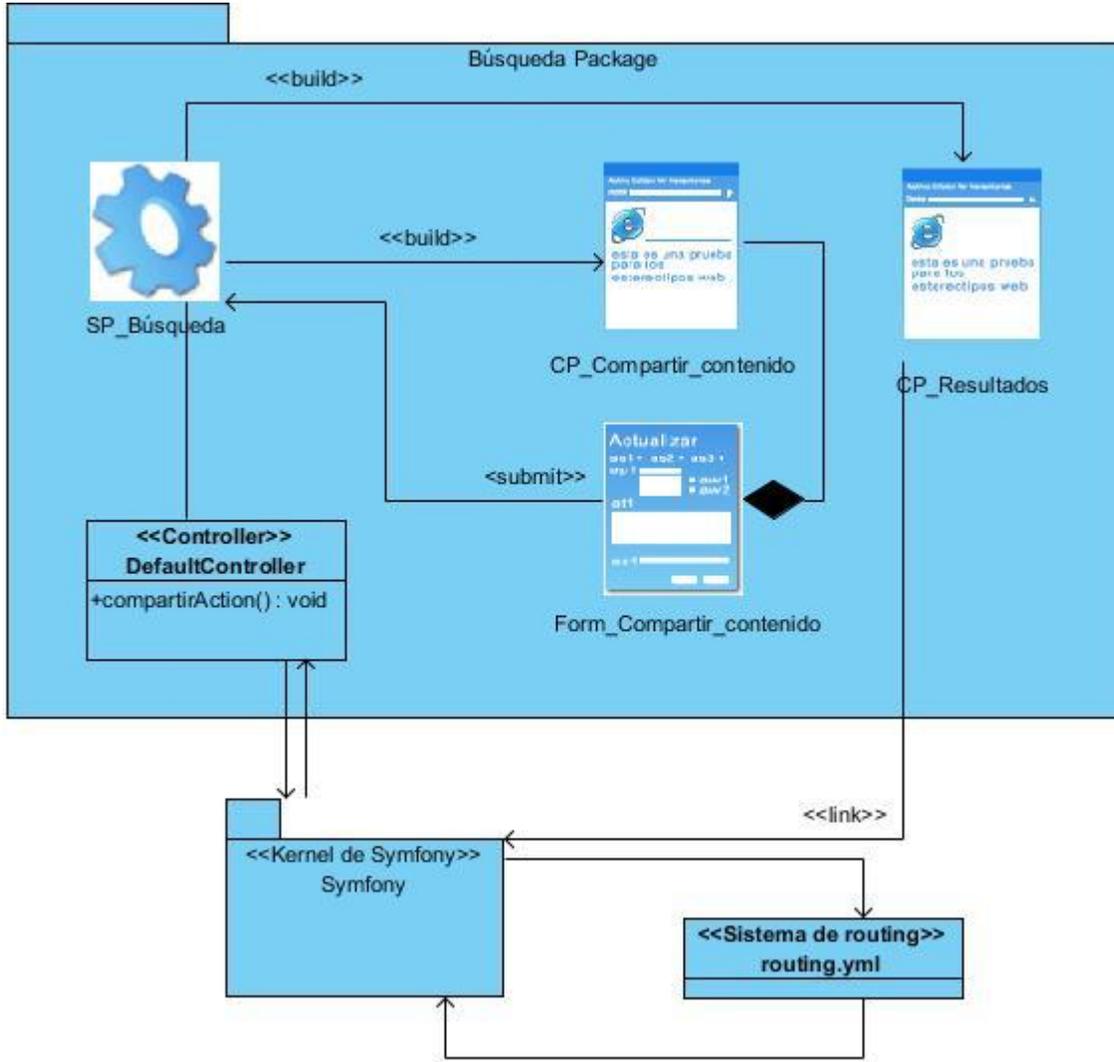


Figura No. 18: Diagrama de Clases del Diseño del caso de uso Compartir contenidos.

Anexo No. 16: Diagrama de Secuencia del caso de uso Autenticar usuario.

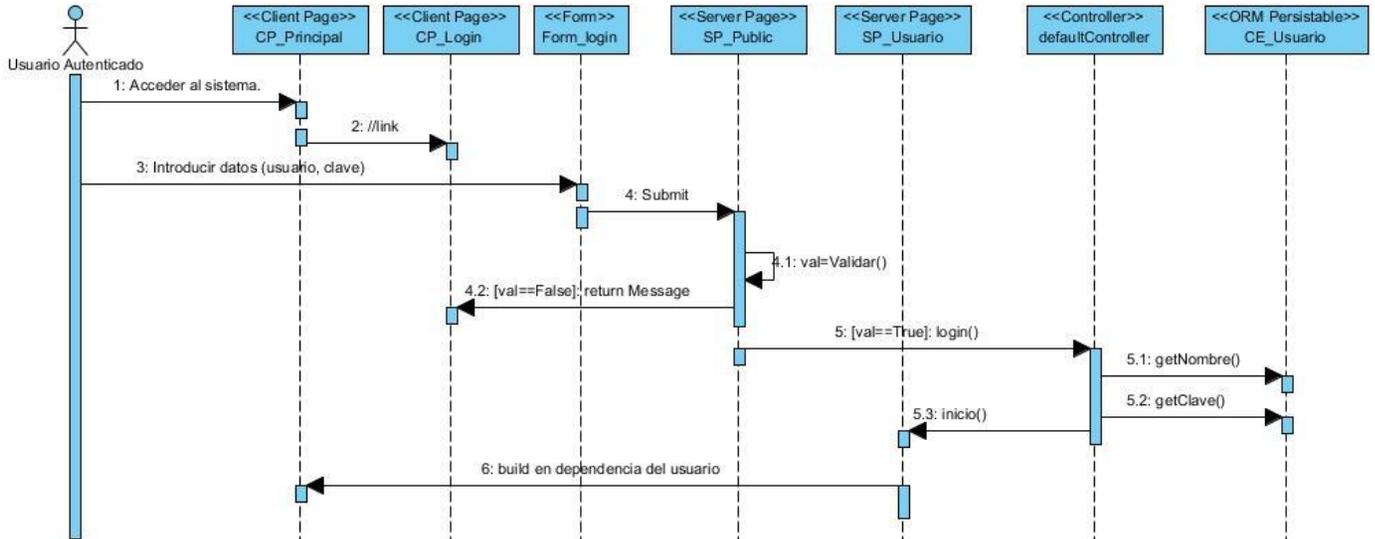


Figura No. 19: Diagrama de Secuencia del caso de uso Autenticar usuario.

Anexo No. 17: Diagrama de Secuencia del caso de uso Compartir contenidos.

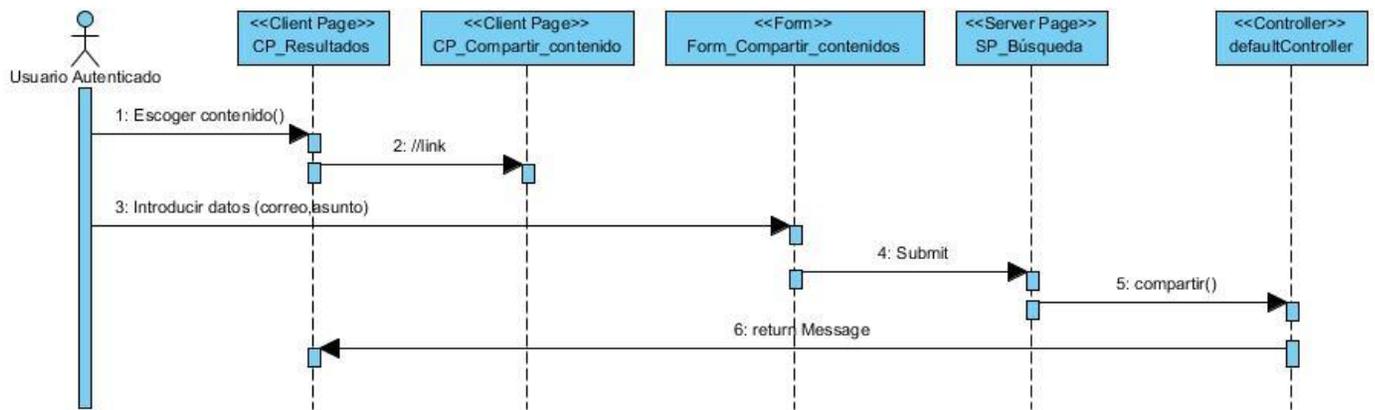


Figura No. 20: Diagrama de Secuencia del caso de uso Compartir contenidos.

Anexo No. 18: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Adicionar usuario.

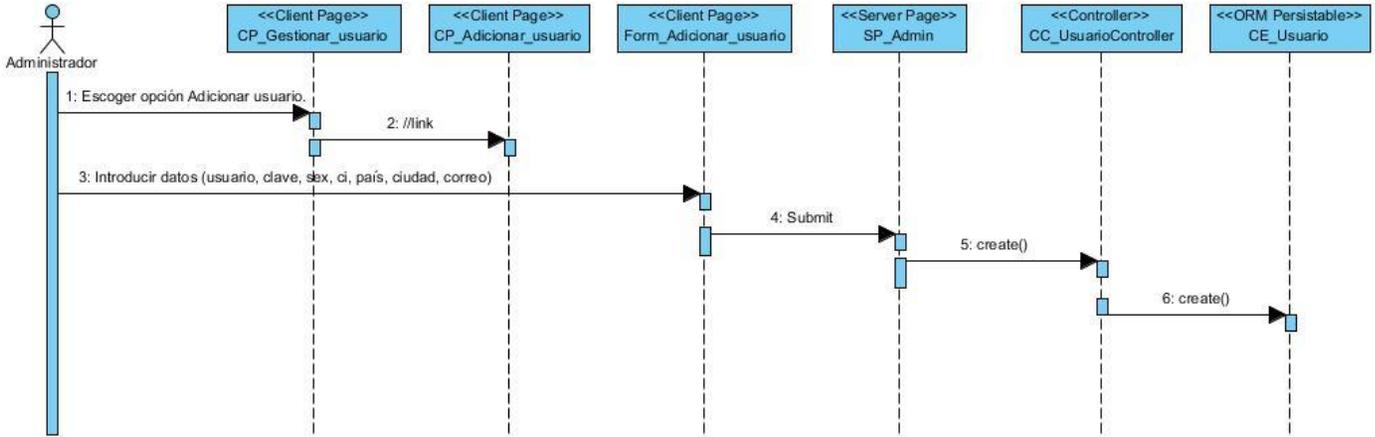


Figura No. 21: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Adicionar usuario.

Anexo No. 19: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Editar usuario.

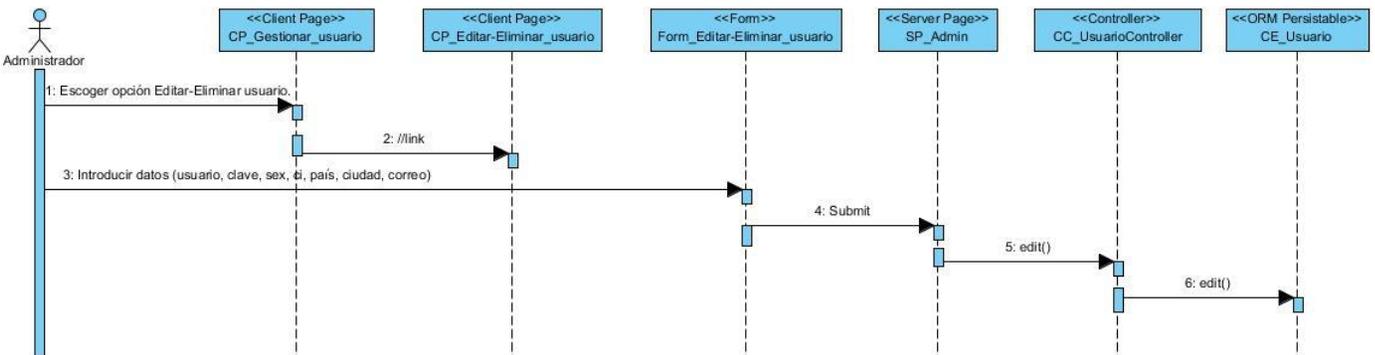


Figura No. 22: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Editar usuario.

Anexo No. 20: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Eliminar usuario.

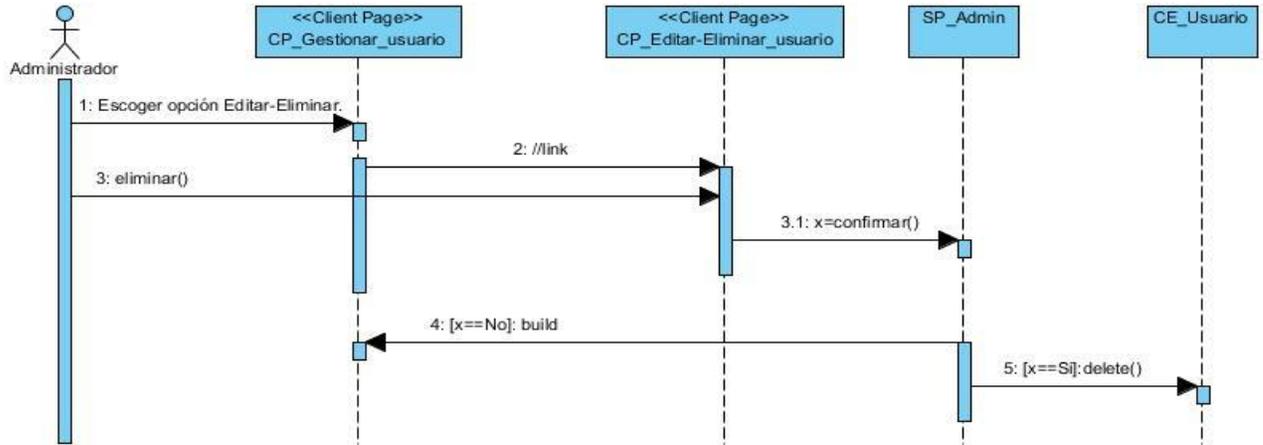


Figura No. 23: Diagrama de Secuencia del caso de uso Gestionar usuario, sección Eliminar usuario.

Anexo No. 21: Interfaz web Registrarse.



METABUSCADOR Ayuda Contáctenos RSS Registrarse

Inicio / Registrarse /

Datos personales

Nombre *

Apellidos *

Correo *

Carnet de Identidad *

Ciudad *

País *

Clave *

Sexo * ▼

Confirmar clave *

Figura No. 24: Interfaz web Registrarse.

Anexo No. 22: Interfaz web Resultados.

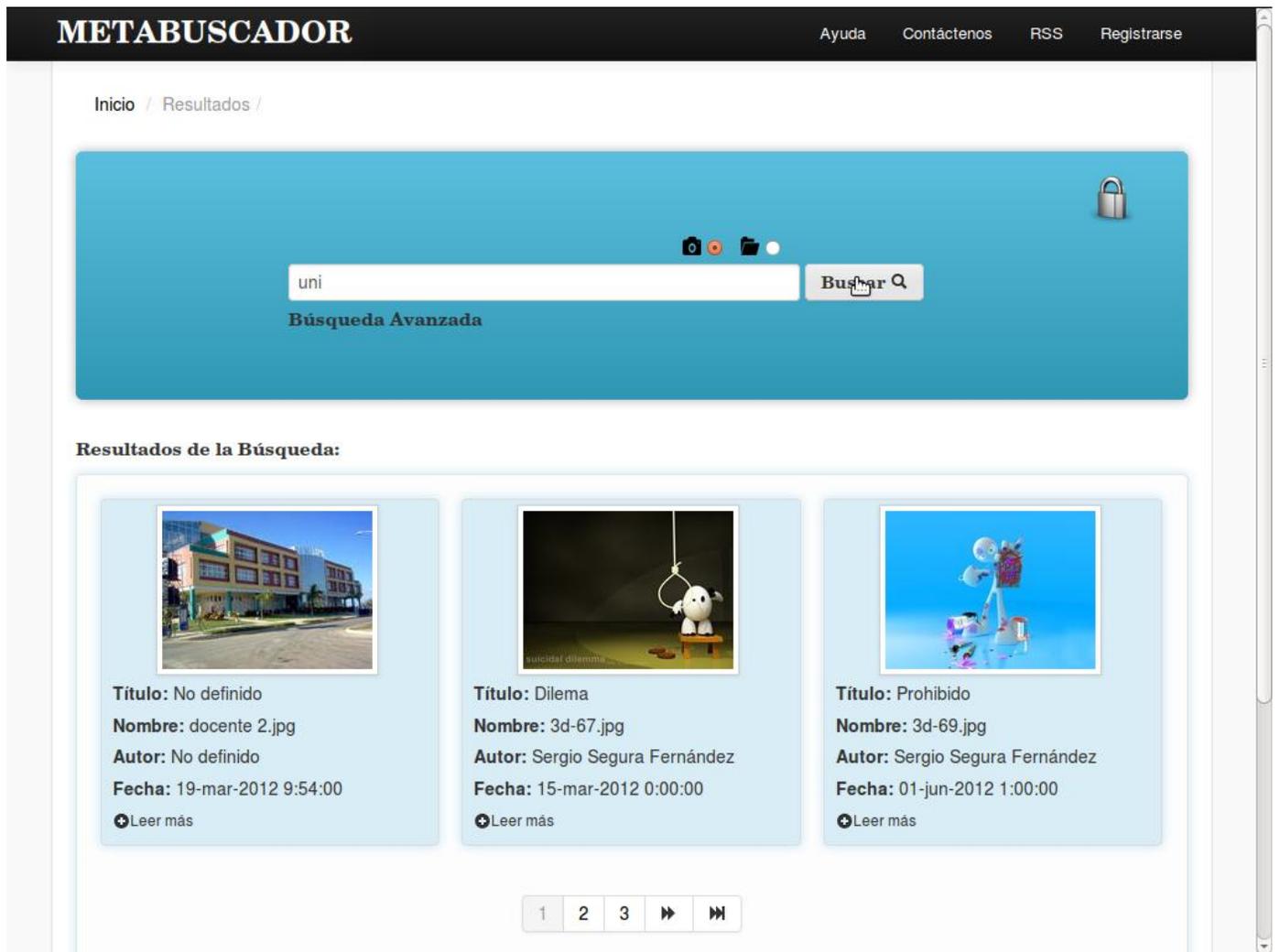


Figura No. 25: Interfaz web Resultados.

Anexo No. 23: Interfaz web Compartir contenidos.

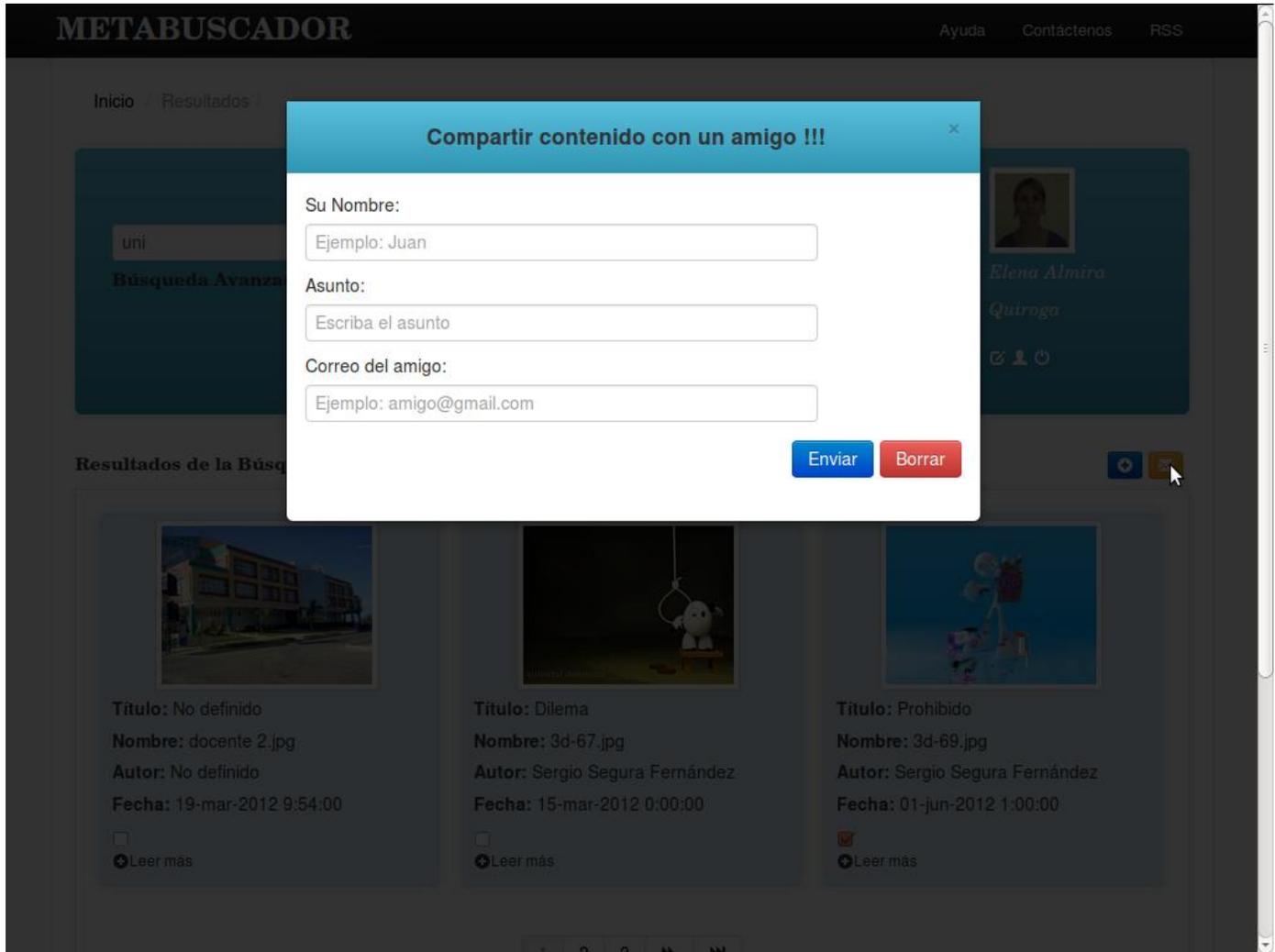


Figura No. 26: Interfaz web Compartir contenidos.

Anexo No. 23: Interfaz web Gestionar.

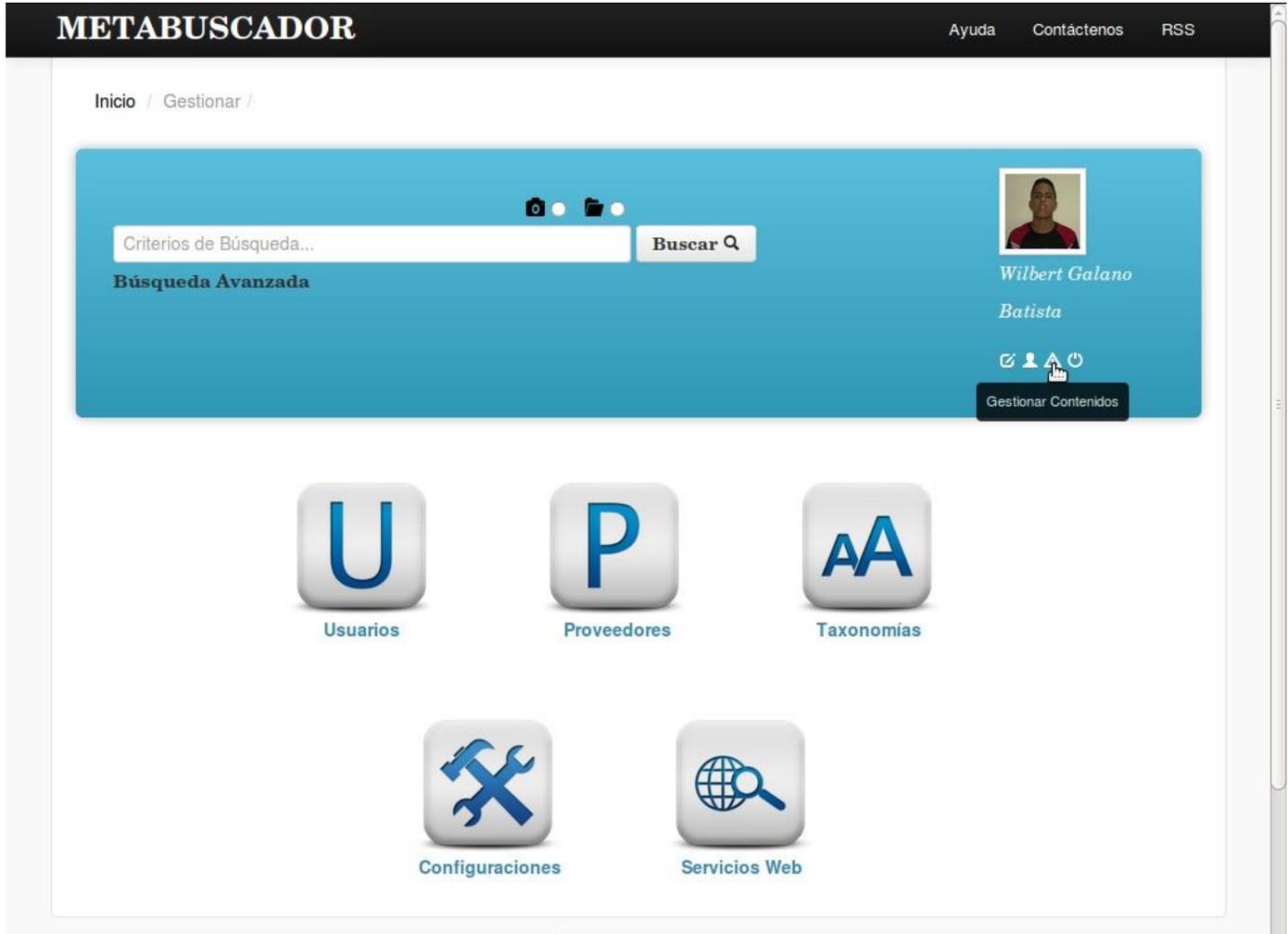


Figura No. 27: Interfaz web Gestionar.