

ARCHIVO DIGITAL PARA FILÓSOFO COLOMBIANO

YOLMAN LENIN CARRILLO GARAVITO

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2014

ARCHIVO DIGITAL PARA FILÓSOFO COLOMBIANO

YOLMAN LENIN CARRILLO GARAVITO

**TRABAJO DE GRADO PRESENTADO COMO REQUISITO PARA OPTAR AL
TÍTULO DE INGENIERO DE SISTEMAS E INFORMÁTICA**

DIRECTORA

Ph.D. SONIA CRISTINA GAMBOA SARMIENTO

CODIRECTOR

M.Sc. CARLOS HUMBERTO CARREÑO DÍAZ

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA**

2014

DEDICATORIA

A mis padres, Yolman Carrillo que con su ejemplo, apoyo y cariño me ha enseñado el camino correcto, el valor de la familia y el poder de la reflexión, a Luz Dary Garavito mi madre que con su amor incondicional, confianza y apoyo siempre me alentó a seguir adelante.

A mis hermanas Eliana e Isabella que con sus ocurrencias y sonrisas alegran mi vida y me impulsan a salir adelante.

A mis abuelos Jaime Garavito que con su apoyo estuvo presente durante esta etapa, Elina Abril quien se convirtió en una segunda mamá y estuvo pendiente de mí en todos los aspectos de mi vida desde el momento en que deje mi casa para recorrer este camino, Blanca de Carrillo que con su amor, consejo y oraciones hizo de mí una mejor persona.

A mi novia Cindy Guarnizo que con su amor y compañía cambio días oscuros con su alegría y con sus palabras cambio defectos por virtudes.

A mis familiares por ser cada uno parte de la persona que hoy en día soy.

AGRADECIMIENTOS

Este proyecto fue desarrollado e implementado satisfactoriamente gracias al apoyo de mi directora de trabajo de grado Sonia Cristina Gamboa Sarmiento quien me oriento en el rumbo del proyecto y moldeo los aspectos representativos del mismo, a mi codirector Carlos Humberto Carreño Díaz quien fue el mentor en la parte tecnológica y guio el rumbo correcto del desarrollo.

Al profesor Germán Vargas Guillén por el apoyo recibido, enseñanza y colaboración en este proceso de formación.

Al maestro Daniel Herrera Restrepo por permitirme hacer parte del proyecto que unificará su trayectoria de vida.

Mis familiares, amigos y compañeros que me apoyaron en este proceso para culminar con satisfacción este ciclo de formación profesional.

CONTENIDO

	pág.
INTRODUCCIÓN	15
1. OBJETIVOS	18
1.1. OBJETIVO GENERAL	18
1.2. OBJETIVOS ESPECÍFICOS	18
2. MARCO TEÓRICO	19
2.1. ARCHIVO	19
2.2. REPOSITORIO DIGITAL	20
2.3. HERRAMIENTAS TECNOLÓGICAS	21
2.3.1. Java	21
2.3.2. Java y JDK (Java Development Kit)	22
2.3.3. EJB	23
2.3.4. JSF	25
2.3.5. Primefaces	25
2.3.6. HTML Y XHTML	25
2.3.7. Java Script	26
2.3.8. MySQL	26
3. METODOLOGÍA	28
3.1. METODOLOGÍA SELECCIONADA	28
4. INICIO DEL PROYECTO	31

4.1.	DIGITALIZACIÓN DE OBJETOS	34
5.	DISEÑO DE LA HERRAMIENTA	37
5.1.	ESPECIFICACIÓN DE REQUERIMIENTOS DEL SISTEMA	37
5.2.	MÓDULOS A IMPLEMENTAR	37
5.3.	DISEÑO DE CASOS DE USO	38
5.3.2.	Casos de Uso Identificados:	41
5.3.3.	Escenarios organizados por autores	43
5.4.	DISEÑO DE BASE DE DATOS	58
5.4.1.	Datos para todos los documentos	59
5.4.2.	Datos de Resumen Analítico de Filosofía (RAF)	62
5.4.3.	Datos Específicos	64
5.4.4.	Traducción	66
5.5.	PATRONES DE SOFTWARE A UTILIZAR	70
5.5.1.	DAO	70
5.5.2.	Facade	71
5.5.3.	MVC	72
6.	DESARROLLO	74
6.1.	EJECUCIÓN DE LA PLANEACIÓN DEL CODIFICACIÓN DE SOFTWARE	74
6.2.	IMPLEMENTACIÓN DE DISEÑO DE CLASES	75
7.	PRUEBAS	83
7.1.	PRUEBAS DE FUNCIONALIDAD	83
7.1.1.	Guía del procedimiento:	83
	BIBLIOGRAFÍA	90

TABLA DE FIGURAS

Figura 1.	Proceso de compilación y ejecución de aplicaciones java	22
Figura 2.	Arquitectura EJB.	24
Figura 3.	Open UP	29
Figura 4.	Readiris Pro 11, Impresión en Pantalla del software	35
Figura 5.	Actores del Sistema.	41
Figura 6.	Diagrama de Usuario y Casos de Uso del Sistema	43
Figura 7.	Diagrama modelo caso de uso - Actor: Usuario Visitante	44
Figura 8.	Diagrama modelo caso de uso - Actor: Usuario Registrado	44
Figura 9.	Diagrama modelo caso de uso - Actor: Usuario Editor	45
Figura 10.	Diagrama modelo caso de uso - Actor: Usuario Administrador	45
Figura 11.	Diagrama de caso de uso - Editar Perfil - Usuario Registrado	46
Figura 12.	Diagrama de caso de uso - Realizar Comentario - Usuario Registrado	46
Figura 13.	Diagrama de caso de uso - Buscar Información - Usuario Editor	46
Figura 14.	Diagrama de caso de uso - Editar Perfil - Usuario Editor	47
Figura 15.	Diagrama de caso de uso - Gestionar Colección - Usuario Editor	47
Figura 16.	Diagrama de caso de uso - Gestionar Contenido - Usuario Editor	48
Figura 17.	Diagrama de caso de uso - Buscar Información - Usuario Administrador	48
Figura 18.	Diagrama de caso de uso - Editar perfil - Usuario Administrador	48

Figura 19.	Diagrama de caso de uso - Gestionar Contacto - Usuario Administrador	49
Figura 20.	Tabla de Clasificación - Datos para todos	60
Figura 21.	Tabla de Clasificación - Datos de Publicación	62
Figura 22.	Tabla de Clasificación - Datos de Resumen Analítico	63
Figura 23.	Tabla de Clasificación - Datos Específicos	65
Figura 24.	Tabla de Clasificación - Traducción	67
Figura 25.	Vista general del esquema Base de Datos	68
Figura 26.	Esquema Base de Datos - parte A	69
Figura 27.	Esquema Base de Datos parte B	70
Figura 28.	Diagrama de Secuencia DAO	71
Figura 29.	Patrón de Diseño Facade	72
Figura 30.	Modelo Vista Controlador	73
Figura 31.	Paquete .model	77
Figura 32.	Paquete .admin.dao	79
Figura 33.	Paquete admin.controller	80
Figura 34.	Arquitectura propuesta para la gestión de archivo en el grupo de investigación	89

TABLA DE TABLAS

Tabla 1. Descripción del Actor Usuario Visitante	39
Tabla 2. Descripción de Usuario Registrado	39
Tabla 3. Usuario Editor de Contenido	39
Tabla 4. Administrador del Sistema	40
Tabla 5. Buscar Información - Caso de Uso	49
Tabla 6. Autenticación Administración - Caso de Uso	50
Tabla 7. Contacto - Caso de Uso	51
Tabla 8. Editar Perfil - Caso de Uso	53
Tabla 9. Gestionar Contenido - Caso de Uso	54
Tabla 10. Gestionar Colección - Caso de Uso	55
Tabla 11. Gestionar Usuario - Caso de Uso	57
Tabla 12 Primera entrega – Diseño e implementación de interfaces	74
Tabla 13 Segunda Entrega – Diseño e implementación de funcionalidades del sistema	75
Tabla 14 Tercera Entrega – Corrección de estilos y mejoras funcionales	75
Tabla 15 Lista y Funciones Paquete model	77
Tabla 16 Lista y Funciones Paquete .admin.dao	78
Tabla 17 Lista y Funciones Paquete admin.controller y portal.controller	80
Tabla 18 Caso Prueba Autenticación	84
Tabla 19 Caso Prueba Crear Colección	85
Tabla 20 Caso de Prueba Comentario	85

RESUMEN

TITULO: ARCHIVO DIGITAL PARA FILÓSOFO COLOMBIANOⁱ

AUTOR: YOLMAN LENIN CARRILLO GARAVITOⁱⁱ

PALABRAS CLAVES: ARCHIVO DIGITAL, PORTAL, HERRAMIENTA, ARCHIVO BIBLIOGRÁFICO, FENOMENOLOGÍA, DANIEL HERRERA RESTREPO.

DESCRIPCION

El archivo digital hoy por hoy se ha convertido en una herramienta muy importante utilizada para preservar y publicar el acervo intelectual de una persona o entidad a lo largo de su trayectoria. En este trabajo se pretendió caracterizar y desarrollar una aplicación software que permitiera alojar, organizar y publicar el archivo de un filósofo colombiano.

El trabajo realizado a lo largo del proyecto permitió establecer los requisitos mínimos para la conservación, manipulación y publicación de un archivo digital, también posibilitó la exploración de herramientas software utilizadas en la digitalización de documentos y repositorios virtuales.

Esta aplicación fue desarrollada a través de un proceso de modelado de requisitos guiado bajo instrucciones previas que Daniel Herrera Restrepo como experto y autor del archivo consideró; desde los contenidos que forman parte de las bases de datos hasta la aprobación de su diseño.

La aplicación permite alojar, clasificar y modificar referencias que cuentan con su respectivo enlace al repositorio donde se encuentra albergado el documento digital, además es posible realizar descripciones sobre las mismas y permite a los usuarios registrados realizar comentarios y búsquedas avanzadas. La herramienta permite publicar noticias sobre temas a fines y cuenta con una sección de páginas relacionadas.

ⁱ Trabajo de grado

ⁱⁱ Facultad de ingenierías físico-mecánicas. Escuela de ingeniería de Sistemas e Informática.
Director Sonia Cristina Gamboa Sarmiento. Codirector: Carlos Humberto Carreño Díaz

ABSTRACT

TITLE: DIGITAL ARCHIVE FOR COLOMBIAN PHILOSOPHERⁱ

AUTHOR: YOLMAN LENIN CARRILLO GARAVITOⁱⁱ

KEYWORDS: DIGITAL ARCHIVE, PORTAL, WEB TOOL, BIBLIOGRAPHIC ARCHIVE, PHENOMENOLOGY, DANIEL HERRERA RESTREPO.

DESCRIPTION

Nowadays, the digital archive has become a really important tool used to preserve, publish intellectual production of a person or entity along their working experience. The purpose of this project was to characterize and develop a software application that would allow to host, organize and publish the archive of a Colombian philosopher.

The work developed throughout this project allowed us to establish the requirements for the preservation, manipulation, and publication of a digital archive. It also permitted the exploration of software tools used for digitalization of documents and virtual repositories.

This application was created through a process of modeled requirements guided under the previous directions that Daniel Herrera, expert and creator of the archive, considered appropriate; both the contents that are part of the data base and the design approval.

The application allows you to host, sort and modify references that have their respective link to the data base where the digital document is housed, it is also possible to make descriptions about the references and it allows registered users to make comments and advanced searches. The tool allows you to publish news about similar topics and has a section of related pages.

ⁱ Graduation project

ⁱⁱ Physical and Mechanical Engineering Faculty. School of System and Informatics Engineer.
Principal: Sonia Cristina Gamboa Sarmiento. Tutor: Carlos Humberto Carreño Díaz

INTRODUCCIÓN

La fenomenología y las disciplinas que se derivan de ella, como la hermenéutica, la deconstrucción y la arqueología, han hecho del concepto de *archivo* no sólo una densa teoría, sino una sólida práctica. El *Movimiento Fenomenológico* históricamente ha tomado como criterio y fuente de referencia al *Archivo Husserl en Lovaina* para soportar procesos de investigación y para dilucidar problemas de validez y de legitimidad en el análisis o la interpretación de la disciplina misma, como si se tratara de un recurso a la instancia última.

En América Latina generación de fenomenólogos, los “modernizadores de la filosofía” en América Latina, que atendieron clases, participaron en seminarios y desarrollaron sus investigaciones doctorales en Lovaina han determinado el impacto de la fenomenología tanto en idioma castellano como en esta región del planeta. Cabe mencionar, entre ellos, a Carlos Astrada, Walter de Reyna, Danilo Cruz Vélez, Daniel Herrera Restrepo y Guillermo Hoyos.

En la emergencia y expansión de la fenomenología, así como en el establecimiento de su sentido y su validez, el *archivo* se ha venido constituyendo en un dispositivo de y para el diálogo virtual: entre los vivos y los muertos entre sí vía sus legados y la interpretación de los mismos por los vivos, con las generaciones por venir.

Cabe, entonces, preguntarse por la naturaleza de este dispositivo en un contexto permeado por las tecnologías de la información y la comunicación, que ha llevado a representar y construir múltiples contextos de la vida humana, en entornos virtuales que han propiciado, no solamente otras formas de interacción, sino

nuevas formas de almacenamiento, búsqueda, presentación de información, y, por
qué no, de construcción del conocimiento.

JUSTIFICACIÓN

En el marco del Proyecto de investigación "Modelo de procesamiento de información para archivo en fenomenología y hermenéutica", financiado por la VIE con el código 5573 el cual pretende diseñar un modelo de procesamiento de información para archivos en fenomenología y hermenéutica, se requiere conocer qué características (datos, descripciones, procesos) que se deben tener en cuenta para clasificar, almacenar y publicar los documentos que conforman el *Archivo* de un filósofo. Este proyecto, específicamente, pretende abordar el acervo del filósofo colombiano Daniel Herrera Restrepo, atendiendo a los criterios que él mismo, como propietario y como experto, determine para la conservación de su *Archivo*.

De esta forma se busca elaborar una aplicación software con todas las características necesarias para la manipulación de todos los documentos contemplados en este archivo digital, realizando un barrido a lo largo de la carrera de este prestigioso filósofo, y buscando aspectos relevantes sobre su aporte en la filosofía; de tal forma que facilite su acceso y reúna, en un solo lugar, el material de su autoría dirigido hacia el público o entidad interesada en su vida y obra.

1. OBJETIVOS

1.1. OBJETIVO GENERAL

Caracterizar y desarrollar una aplicación software para alojar, organizar y publicar el Archivo de un filósofo colombiano.

1.2. OBJETIVOS ESPECÍFICOS

- Realizar la especificación de requerimientos y el diseño de la arquitectura necesaria para la gestión del Archivo digital.
- Diseñar los modelos de bases de datos que comprenden la estructura de Archivo digital.
- Proponer el diseño de una estructura de datos que permita el almacenamiento, manipulación y consulta de los elementos que conforman el Archivo digital establecido.
- Publicar un sitio web con el Archivo del filósofo, a partir de las estructuras propuestas.

2. MARCO TEÓRICO

2.1. ARCHIVO

La definición formal de archivo, según la RAE¹, propone un conjunto ordenado de documentos que una persona, una sociedad, una institución, etc., producen en el ejercicio de sus funciones o actividades. Por su parte, el Diccionario de Terminología Archivística (citado por Fuster Ruiz, 1999) establece el archivo como: (1) Conjunto orgánico de documentos producidos y/o recibidos en el ejercicio de sus funciones por las personas físicas o jurídicas, públicas y privadas. (2) La institución cultural donde se reúnen, conservan, ordenan y difunden los conjuntos orgánicos de documentos para la gestión administrativa, la información, la investigación y la cultura. (3) El archivo también es el local donde se conservan y consultan los conjuntos orgánicos de documentos.

Un conjunto de datos organizados constituyen información dentro de un mensaje que cambia el estado de conocimiento del sujeto quien lo acoge. En todo tipo de disciplinas y ciencias se busca la forma de preservar la exploración del discernimiento sobre algún tema en específico; durante mucho tiempo el conocimiento fue preservado mediante la herencia del mismo, de generación en generación se transmitían los saberes y a partir de estos se realizaban nuevos aportes; es la creación de archivos una forma más fácil de registrar el conocimiento acerca de algo, de aquí la importancia de conservar las enseñanzas o saberes para su difusión universal.

Por su parte, la fenomenología, así como las disciplinas derivadas de ésta entre las cuales cabe mencionar: la hermenéutica, la deconstrucción y la arqueología,

¹ Definición disponible en "[http://lema.rae.es/drae/?val=archivo,](http://lema.rae.es/drae/?val=archivo)"

convirtieron el *archivo* en una teoría, acompañada, igualmente, de una concreta práctica. El concepto de archivo y su manejo se hace presente en esta disciplina en un hecho histórico en el interior mismo del *Movimiento Fenomenológico*, corriente asociada al *Archivo Husserl en Lovaina* el cual se toma como criterio de referencia para dilucidar los problemas tanto de interpretación como de desarrollo de los procesos de investigación. De tal forma el concepto de archivo digital en el siglo XXI, se consolida como una forma de asegurar la preservación del conocimiento y la información a través de su vínculo con la tecnología para su propagación; aprovecha el alcance universal de la web como una manera de conservar y difundir la información.

El presente proyecto promueve la creación del “Archivo Virtual para Filósofo Colombiano”, vida y obra del filósofo y escritor Daniel Herrera Restrepo. Para lo cual es necesario el establecimiento de un repositorio digital como medio de almacenamiento donde se pretende alojar sus referencias bibliográficas junto con sus obras más reconocidas.

2.2. REPOSITORIO DIGITAL

Un repositorio digital consiste en un almacén de objetos digitales que permite almacenar “(...) una imagen, un documento Word o Excel, un documento digitalizado, un libro electrónico, una página html, etc. [que] puede formar parte del ‘repositorio digital’ de la organización. Es decir, es un concepto que va mucho más allá de la digitalización de documentos” (Alós-Mone, 2010, pp. 205-210).

Teniendo en cuenta que hoy en día se habla mucho sobre digitalización de la información, este concepto es pertinente, dado que mediante este método es posible obtener información necesaria (textos, fotos, audio, etc.). No obstante lo anterior, la mayor cantidad de información que se produce día a día es encontrada

de forma escrita, tipográfica o manuscrita, si se quiere contar con esta información en la forma descrita, digital, es indispensable pasar por un proceso que permita la administración del contenido de los objetos digitales, llamado *reconocimiento óptico de caracteres* –ROC–.

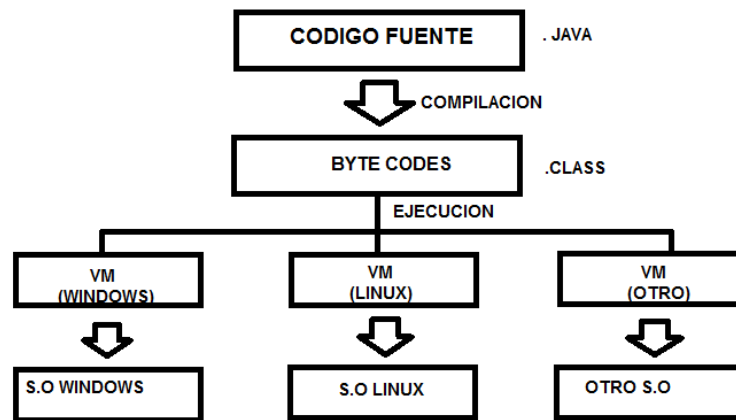
2.3. HERRAMIENTAS TECNOLÓGICAS

Las herramientas de uso libre (GNU y Open Source) permiten realizar proyectos software de variada envergadura debido a la amplia comunidad y bajo costo en torno a ellas. A continuación se describirán las herramientas utilizadas en el proyecto.

2.3.1. Java

Java es un lenguaje de programación creado en 1991 por James Goslin de la empresa Sun Microsystem, este realizó el diseño e implementación original del compilador y la máquina virtual Java. Se caracteriza por ser un lenguaje orientado a objetos, su sintaxis procede en parte de lenguajes como C y C++. Las aplicaciones pueden ejecutarse en cualquier entorno virtual java, sin importar la arquitectura de la máquina (Shi, 2005).

Figura 1. Proceso de compilación y ejecución de aplicaciones java



Fuente: (Sierra, 2008)

En la figura 2, se muestra el proceso de compilación y ejecución de una aplicación java. Se presenta de una manera gráfica como el código es compilado y ejecutado por máquinas virtuales sin importar el sistema operativo del cliente.

Entre las principales características destacan el ser un lenguaje orientado a objetos, el uso de librerías que permiten al programador reducir considerablemente el tiempo de desarrollo, portabilidad de sus aplicaciones, seguridad en ejecución al proporcionar restricciones de acceso y amplio soporte (Sierra, 2008).

La gran popularidad de este lenguaje ha propiciado su utilización en diversos entornos tales como dispositivos móviles, navegadores web, servidores, aplicaciones de escritorio, entre otras.

2.3.2. Java y JDK (Java Development Kit)

Java es un lenguaje desarrollado por Sun Microsystems, en el año 2009 fue adquirida por la compañía Oracle. Permite escribir aplicaciones que puedan ejecutarse en casi cualquier plataforma. El lenguaje toma parte de la sintaxis de C

y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Además, cuenta con una característica denominada “recolección de basura”, que examina la memoria y libera cualquier variable u objeto que no esté siendo usado.

El JDK es un software que provee herramientas de desarrollo para la creación de programas en java. Para trabajar con Java se necesita un kit de desarrollo que proporciona:

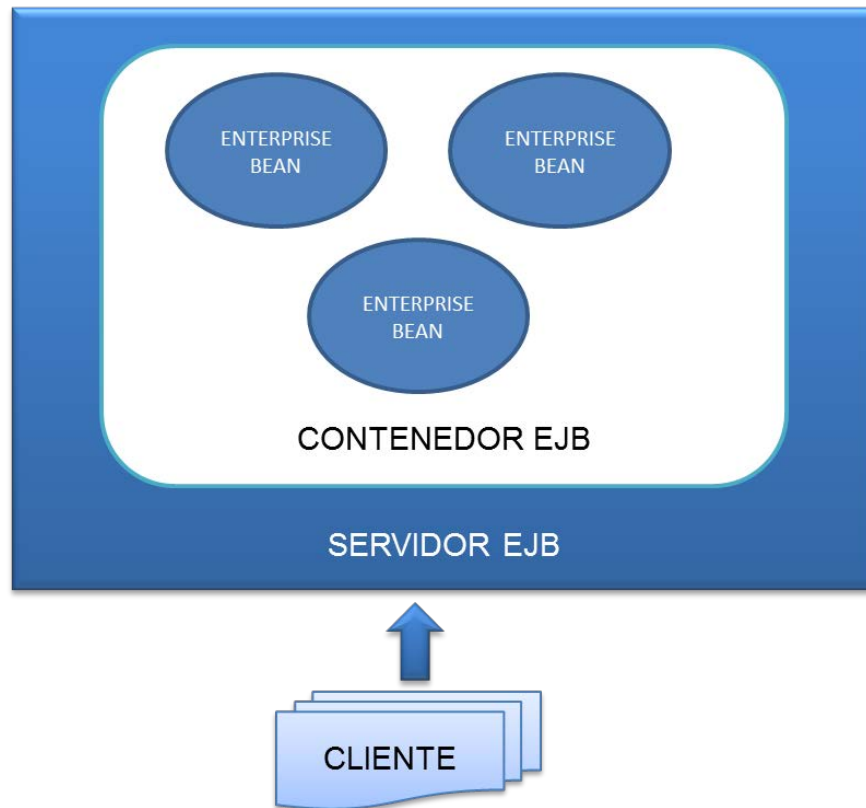
- Un compilador: *javac*
- Un intérprete: *java*
- Un generador de documentación: *javadoc*
- Un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java: Appletviewer.

2.3.3. EJB

Enterprise Java Beans (EJB), es un arquitectura por componentes del lado del servidor que encapsula la lógica del negocio de una aplicación utilizando el lenguaje de programación java. La tecnología EJB se caracteriza por implementar la ejecución en un contenedor.

El contenedor se encuentra dentro de un servidor de aplicaciones que presta servicios como: ciclo de vida de los EJB, balanceo de carga, seguridad de acceso, gestión de persistencia entre otros.

Figura 2. Arquitectura EJB.



Elaboración propia. A partir de Java EE – Enterprise Beans (EJB) (Pavón Maestras, 2013)

Existen 3 tipos de componentes EJB presentados a continuación:

- Enterprise Bean de Sesión (Session Bean): permiten encapsular la lógica de negocio. Están divididos en:
 - Stateless Session Beans: sin estado entre invocaciones.
 - Stateful Session Beans: mantienen estado de un cliente.
 - Singletons: compartida por varios clientes.

- Enterprise Bean Dirigido por Mensaje: se activan a través de un mensaje, se caracteriza por que no mantienen estado.
- Entity Bean: representan los datos almacenados en una base de datos mediante Java Persistence API(JPA)

2.3.4. JSF

Java Server Faces (JSF) es un framework de Java para aplicaciones web que utiliza como una estructura de trabajo con interfaces de usuario del lado del servidor, es decir, JSF permite construir aplicaciones web basadas en un entorno de trabajo que administra las acciones generadas por el usuario en su página HTML y las convierte a eventos que son enviados al servidor.

JSF Utiliza el patrón modelo – vista – controlador (M-V-C). Los principales componentes de JSF se caracterizan por definir la navegación entre páginas, proporcionar extensibilidad, representar componentes de interfaz de usuario y librerías de etiquetas JSP.

2.3.5. Primefaces

Es un conjunto de componentes Java Server Faces creado por Cagatayt Civici (Optimus Prime en la comunidad de Primefaces) contiene gran variedad de extensiones que facilitan el desarrollo de aplicaciones web. Los componentes que contiene primefaces son *HTML* y *XHTML editor*, *Dialog*, *Auto Complete*, *Charts* además de Soporte de inserción a través de Ajax web sockets (Çivici, 2008).

2.3.6. HTML Y XHTML

Es un lenguaje desarrollado por Tim Bernee-Lee, durante la década de los años noventa para la comunicación entre un servidor y un cliente usando el protocolo http¹, debido a la necesidad de compartir información entre computadoras del laboratorio de física CERN en Europa.

Los navegadores web fueron creados con el objetivo de para mostrar al cliente el código HTML de una manera gráfica. HTML permite escribir documentos de texto mostrados de forma estructurada, con enlaces que llevan a otros documentos. Su sintaxis usa etiquetas rodeadas de corchetes angulares similares a los signos mayor que o menor que.

Con el paso de los años como la gran mayoría de lenguajes HTML ha mostrado distintas versiones de acuerdo a las necesidades del entorno web, una de la más significativas es XHTML que nació de la limitaciones que tenía HTML con las herramientas basadas en XML² (lenguaje de marcas extensibles) mejorando así la forma como se muestran los datos en el navegador (Jac, 2007).

2.3.7. Java Script

JavaScript es un lenguaje de programación creado a mediados de los años noventa por Brendan Eich, quien adaptó la tecnología existente (ScriptEase) al navegador Netscape Navigator debido al auge de las páginas web que exigían formularios cada vez más complejos.

JavaScript es un lenguaje interpretado que no necesita compilación previa a la ejecución, es decir, cualquier programa desarrollado puede probarse en cualquier navegador web. El nombre JavaScript no tiene ninguna relación directa con el lenguaje java, se denominó de esta forma por una alianza con Sun Microsystem (Pérez, 2009).

2.3.8. MySQL

Es uno de los sistemas de gestión de bases de datos (open source) relacional, multiusuario, más popular entre el mundo de desarrolladores. Es un sistema de base de datos operacional considerado uno de los más importantes y utilizados por usuarios del medio para el diseño y programación de base de datos de tipo

relacional. MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo.

La característica más interesante de MySQL es que permite recurrir a las bases de datos multiusuario a través de la web y en diferentes lenguajes de programación y diferentes plataformas que se adaptan a diferentes necesidades y requerimientos, además MySQL es conocida por desarrollar alta velocidad de búsqueda de datos e información, a diferencia de sistemas anteriores.

Ventajas de MySQL

- El MySQL es un Open Source, o sea código abierto que puede ser usado y modificado.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consume puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL altamente apropiado para acceder a bases de datos en internet.

3. METODOLOGÍA

3.1. METODOLOGÍA SELECCIONADA

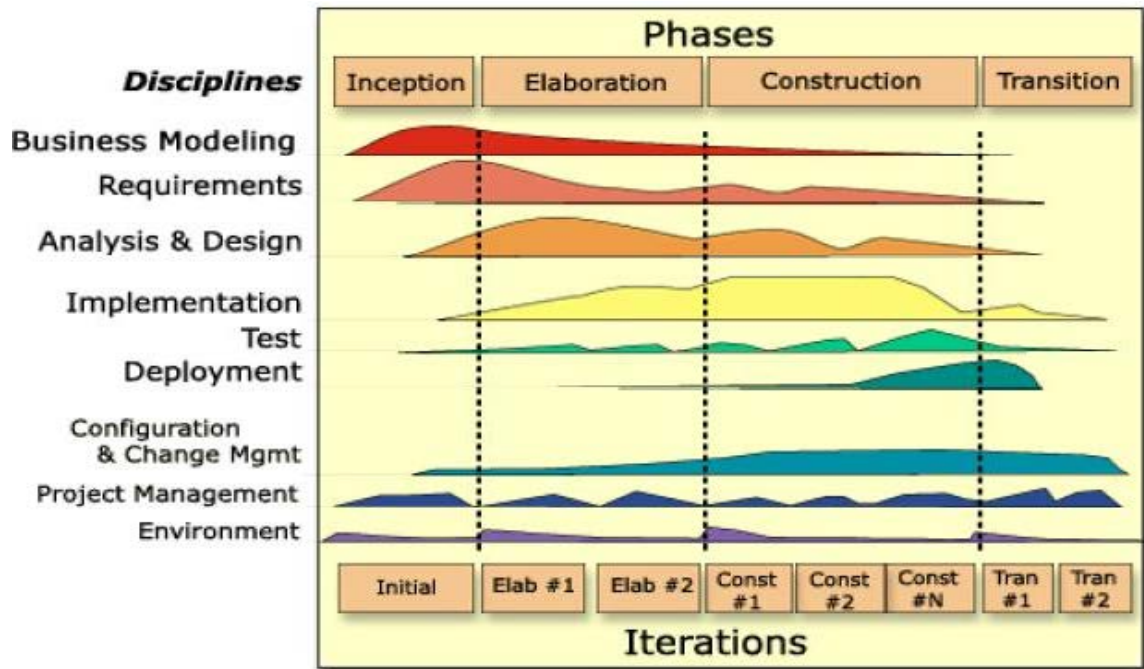
En el desarrollo de la herramienta resultante de este proyecto se utilizó el proceso Open UP, el cual es un método de trabajo que involucra un conjunto mínimo de prácticas tendientes a guiar a un equipo de trabajo pequeño en el análisis, diseño, desarrollo y despliegue de un producto de software. Los objetivos que persiguen son:

- 1- Promover la colaboración y compartir conocimientos alineando intereses del equipo de trabajo y los usuarios.
- 2- Ayudar al equipo a enfocarse en la arquitectura de forma rápida; de tal forma que se minimicen los riesgos y se organice el desarrollo.
- 3- Ayudar al equipo a balancear prioridades en conflicto para maximizar el valor obtenido por los interesados en el proyecto.
- 4- Ayudar al equipo en la evolución continua del producto para obtener realimentación continua y fomentar el mejoramiento.
- 5- Permitir a los administradores del proyecto realizar seguimientos a las avances basados en metas e indicadores
- 6- Permitir que los integrantes del equipo entiendan rápidamente como realizar el trabajo para alcanzar los objetivos y metas proyectadas.

OPEN UP es un proceso iterativo e incremental que se distribuye a través de cuatro fases: inicio, elaboración, construcción y transición. En las cuales se desarrollan transversalmente una serie de subprocesos entendiéndose estos últimos como un conjunto de actividades, personas (roles), prácticas (guías) y

productos de trabajo (artefactos) que orientan el desarrollo de software a través del tiempo.

Figura 3. Open UP



Fuente: <http://epf.eclipse.org/wikis/openup/index.htm>

Como se puede observar en la figura existen cuatro fases principales, en las que se realizan múltiples iteraciones a través de las cuales se obtienen resultados parciales de la herramienta software a desarrollo. Las fases que componen Open UP son las siguientes:

Fase de Inicio: Primera fase del proceso, donde los stakeholders y los integrantes del equipo de desarrollo, colaboran para determinar el ámbito del proyecto, sus objetivos y determinar si el proyecto es viable.

Fase de Elaboración: Es la segunda fase dentro del ciclo de vida del proyecto. En ella los riesgos significativos que influyen en la arquitectura son identificados y considerados.

En esta fase:

- Se obtiene un entendimiento más detallado de los requerimientos y requisitos
- Se diseña, implementa valida y establece la línea base de la arquitectura..
- Se mitigan los riesgos esenciales.
- Se produce un cronograma detallado.
- Se realiza una mejor estimación de costos

Fase de Construcción: Esta es la tercera fase del proceso, se enfoca en detallar los requisitos y requerimientos, diseñar, implementar y probar el grueso del software y completar el desarrollo del sistema basado en la arquitectura.

Fase de Transición: Es la cuarta fase del proceso. Se enfoca en la transición del producto de software a la plataforma tecnológica del cliente logrando que los interesados convengan que el desarrollo del producto cumple con los requerimientos planteados.

4. INICIO DEL PROYECTO

Durante la fase de inicio se procedió a determinar el ámbito del proyecto y especificar los objetivos que ayudarán a determinar el alcance del trabajo a realizar. De tal forma que en Mayo del 2013, se viajó a la Universidad Pedagógica Nacional con sede en Bogotá, donde se dio inició el proyecto “Archivo Virtual para Filósofo colombiano”, dedicado a Daniel Herrera Restrepo, Fenomenólogo formado en Lovaina (Bélgica) y Friburgo (Alemania), quien es reconocido por su aporte a la Fenomenología en Colombia. Allí se procedió a determinar las características principales del problema de la recolección de su acervo campo académico e intelectual.

A través de reuniones con los principales *stakeholders*² del proyecto, representados por el filósofo Daniel Herrera Restrepo, el doctor Germán Vargas Guillén de la Universidad Pedagógica Nacional y la doctora Sonia Cristina Gamboa de la Universidad Industrial de Santander, se estableció el conjunto de escritos realizados a lo largo de la carrera de Daniel, los cuales harán parte del proyecto “Archivo Virtual para Filósofo colombiano”.

Posteriormente se estableció un cronograma de trabajo donde se fijan las siguientes actividades:

- **Realizar el inventario de las publicaciones del filósofo, así como de su trabajo inédito:** Esta meta consiste en consolidar un listado de las referencias bibliográficas de los trabajos editados y una descripción similar

² El término *Stakeholder* representa a los “participantes”, “inversores”, “accionistas” y se utiliza para referirse a los grupos de interés para una empresa. De manera general el término define a todas aquellas personas o entidades que pueden afectar o son afectados por las actividades que hacen parte de la lógica de negocio de la organización (<http://www.guioteca.com/>).

de los objetos inéditos (textos, fotos, videos, audios, etc.). En la clasificación inicial del archivo, se realizó la siguiente clasificación:

Editado:

libro

capítulo de libro

artículo

reseña

repertorio bibliográfico

documento oficial

Inédito:

nota suelta

correspondencia

artículo inédito

ponencia

nota de trabajo

participación en documentos

oficiales

audio, video y fotografía.

- **Digitalizar y organizar todo el material posible que corresponda con el inventario realizado:** Cada una de las piezas digitalizadas se llamará *objeto*; y estos se clasificarán en primera instancia en dos categorías: *editado* e *inédito*. La digitalización de la obra del autor consiste en escanear cada una de las piezas y luego llevar a cabo un proceso de *reconocimiento óptico de caracteres* (ROC, u OCR en inglés). Este último proceso garantiza el tratamiento automático de los textos (edición, búsqueda de cadenas de caracteres, cambio de formato, elaboración de índices, etc.).
- Publicar los objetos digitalizados en el repositorio digital www.archive.org, el cual, es una librería digital de sitios de internet y otros artefactos culturales en formato digital.

4.1. DIGITALIZACIÓN DE OBJETOS

El reconocimiento óptico de caracteres u OCR (optical character recognition), es un proceso orientado a la digitalización de documentos en el que se identifican como imagen cada uno de símbolos que comprende un escrito en un idioma determinado, para almacenarlos en forma de datos automáticamente y posteriormente proceder a manipularlos con cualquier editor de texto. De acuerdo con Arlandis (2013), el problema del reconocimiento óptico de caracteres es “bien conocido y ha estado abordado de manera intensa por disciplinas científicas, como el Reconocimiento de Formas y Visión Artificial.” (Arlandis,J., 2013. p. 9-11).

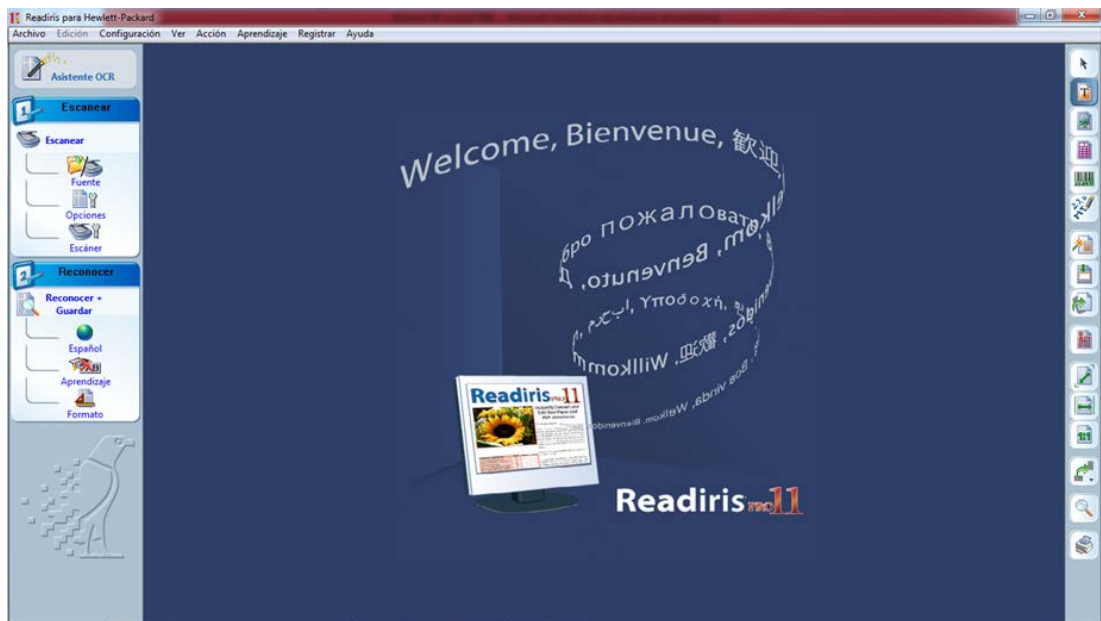
Para ello se realizó la búsqueda y análisis de algunas de las herramientas software más reconocidas para el reconocimiento óptico de caracteres (OCR), identificando las siguientes herramientas:

- OmniPage Standard
- Presto! OCR
- Readiris Pro
- PDF Transformer
- Acrobat Standard

También se determinó que para el proceso de digitalización, la Universidad Pedagógica Nacional (UPN), era necesario adquirir un escáner cuyas características y beneficios se adecuaban al problema en cuestión. Para ello se adquirió el scanner Hewlett Packard Scanjet 5590, el cual fue instalado y con el que se hicieron las primeras pruebas de digitalización utilizando el software Readiris Pro 11 incluido en la compra.

Adicionalmente se integró un grupo cuatro(4) de auxiliares estudiantes de filosofía de la UPN, los cuales fueron entrenarlos en el uso del escáner y manejo del software, adelantando así este proceso necesario para el buen desarrollo del proyecto.

Figura 4. Readiris Pro 11, Impresión en Pantalla del software



5. DISEÑO DE LA HERRAMIENTA

5.1. ESPECIFICACIÓN DE REQUERIMIENTOS DEL SISTEMA

Como primera actividad diseño se establecieron los requerimientos del proyecto por medio de reuniones con los *stakeholders* del proyecto, teniendo en cuenta la necesidad de tener un mecanismo para conservar la obra de un filósofo, buscando cumplir criterios de escalabilidad, de manera que el proyecto pueda ser aplicado a otros contextos.

Como requerimientos principales del sistema se pudieron establecer los siguientes:

- Hacer una recopilación de la obra de un filósofo para su publicación web
- Preservar los aportes hechos por el filósofo mediante un archivo digital alojado en un repositorio virtual.
- Crear un sitio web donde pueda tener acceso la comunidad en general a la información y a su vez tener contacto con miembros de la sociedad científica en Colombia especialistas en esta área.
- Mostrar información actualizada sobre el filósofo y la fenomenología.

5.2. MÓDULOS A IMPLEMENTAR

Una vez fueron establecidos los requerimientos del proyecto se realizó un análisis sobre los módulos que harán parte de la herramienta. Los módulos propuestos son:

1. Módulo de Usuario: Este módulo permite la creación y edición de los datos que comprenden el perfil de usuario como:
 - Datos personales
 - Imagen de perfil
 - Contacto
2. Módulo Autenticación: permite la autenticación del usuario al sitio web mediante el uso de credenciales.
3. Módulo Contacto: muestra un formulario para establecer contacto con el administrador del sitio web.
4. Módulo Noticias y Enlaces web: Permite mostrar noticias acerca de temas relacionados con el tema del sitio y la creación y edición de enlaces web a redes sociales, grupos de investigación afines, páginas institucionales, repositorios, etc.
5. Módulo Bibliografía: muestra recursos bibliográficos sobre el autor y los listados en el sitio web.
6. Modulo Colecciones: permite crear y editar objetos pertenecientes al archivo virtual.

5.3. DISEÑO DE CASOS DE USO

La definición de los casos de uso ayuda a brindar una visión del alcance del proyecto y permiten explorar la aplicación de forma conceptual, tomando como punto de referencia lo establecido en los objetivos específicos del proyecto y el análisis de requerimientos realizado en la fase anterior.

Para ello se inició identificando los actores principales del sistema:

5.3.1. Actores Identificados:

1. Usuario Visitante
2. Usuario Registrado
3. Usuario Editor de Contenido
4. Administrador del Sistema

Tabla 1. Descripción del Actor Usuario Visitante

Actor:	Usuario Visitante
Caso de uso:	Buscar Información, Contacto
Descripción:	Usuario con permiso de buscar información
Tipo:	Primario

Tabla 2. Descripción de Usuario Registrado

Actor:	Usuario Registrado
Caso de uso:	Buscar información, Contacto, Editar perfil de Usuario Autenticar editar perfil de usuario, Realizar comentario
Descripción:	Usuario con permisos de editar su perfil de usuario, realizar comentarios en publicaciones, buscar información.
Tipo:	Primario

Tabla 3. Usuario Editor de Contenido

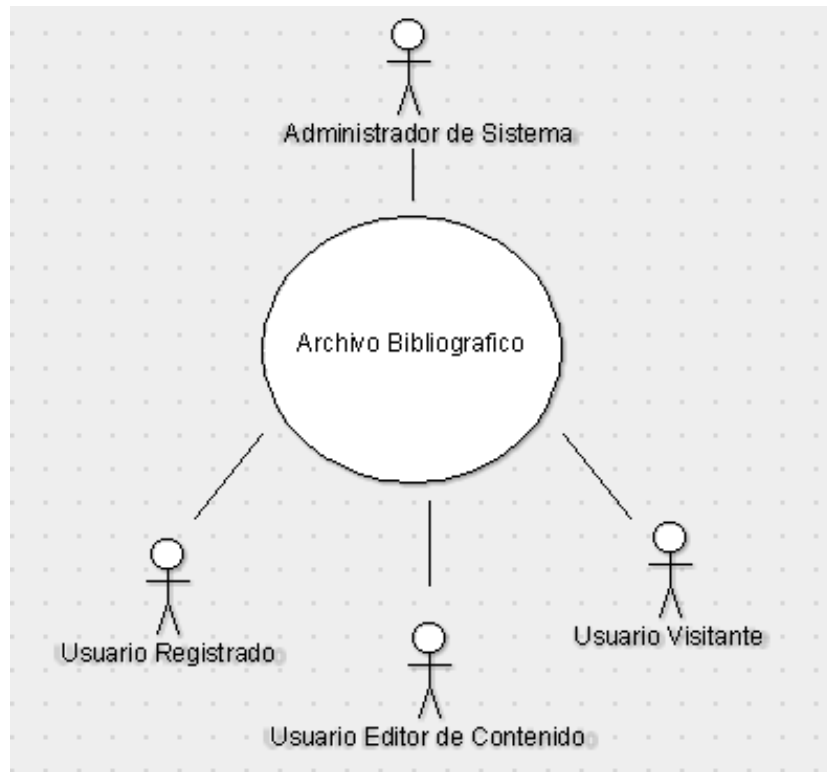
Actor:	Usuario Editor de Contenido
Caso de uso:	Buscar información, Contacto, Editar perfil de Usuario, Autenticar Administración, Autenticar editar perfil de usuario, Realizar comentario, Gestionar colección , Gestionar Contenido
Descripción:	Usuario con permisos de gestionar contenido como (noticias,

	colecciones, enlaces de interés), editar su perfil de usuario, realizar comentarios en publicaciones, buscar información.
Tipo:	Primario

Tabla 4. Administrador del Sistema

Actor:	Administrador del Sistema
Caso de uso:	Buscar información, Gestionar Usuarios, Editar perfil de Usuario, Autenticar Administración, Autenticar editar perfil de usuario, Realizar comentario, Gestionar colección, Gestionar contenido, Gestionar servicios, Gestionar Contacto.
Descripción:	Usuario con todos los privilegios de control y configuración del portal, tiene la capacidad de gestionar en su totalidad las funcionalidades del sistema.
Tipo:	Primario

Figura 5. Actores del Sistema.

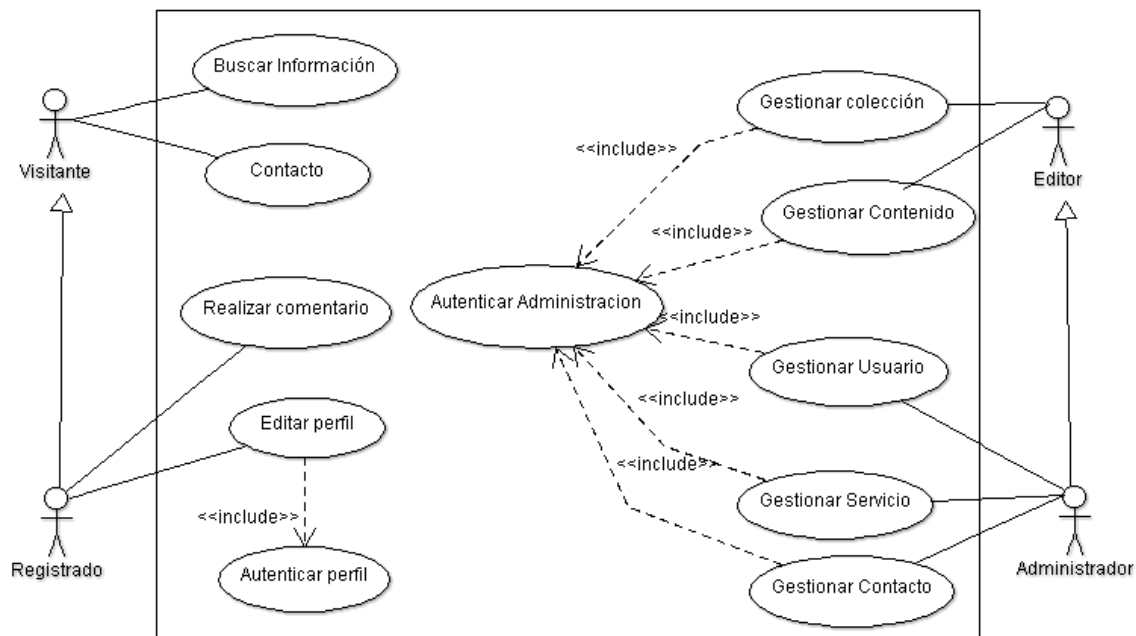


5.3.2. Casos de Uso Identificados:

Una vez determinados los actores del sistema, se procedió a la identificación y diseño de los principales casos de la herramienta software a desarrollar. Los casos identificados fueron:

1. Registro de Usuario
2. Autenticación usuario
3. Editar perfil de usuario
4. Eliminar usuario
5. Crear usuario..
6. Crear Colección
7. Editar Colección
8. Elimina Colección
9. Crear evento (Noticia).
10. Eliminar evento(Noticia)
11. Editar evento(Noticia)
12. Crea Comentario
13. Edita Comentario
14. Elimina Comentario
15. Descarga Contenido.
16. Búsqueda.

Figura 6. Diagrama de Usuario y Casos de Uso del Sistema



5.3.3. Escenarios organizados por autores

En búsqueda de clasificar los diferentes escenarios en los cuales los actores del sistema interactuarán con la herramienta. Se establecieron los principales escenarios de uso para cada usuario. Esta especificación permitirá definir los niveles de permiso del sistema.

Figura 7. Diagrama modelo caso de uso - Actor: Usuario Visitante

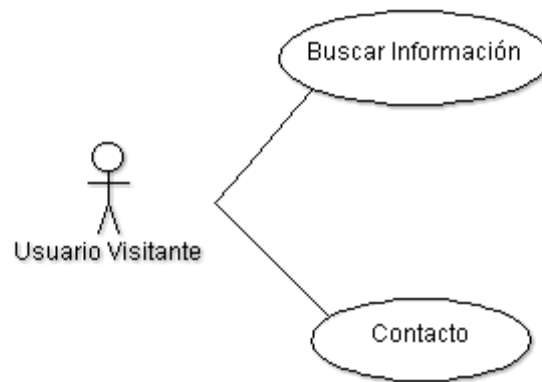


Figura 8. Diagrama modelo caso de uso - Actor: Usuario Registrado

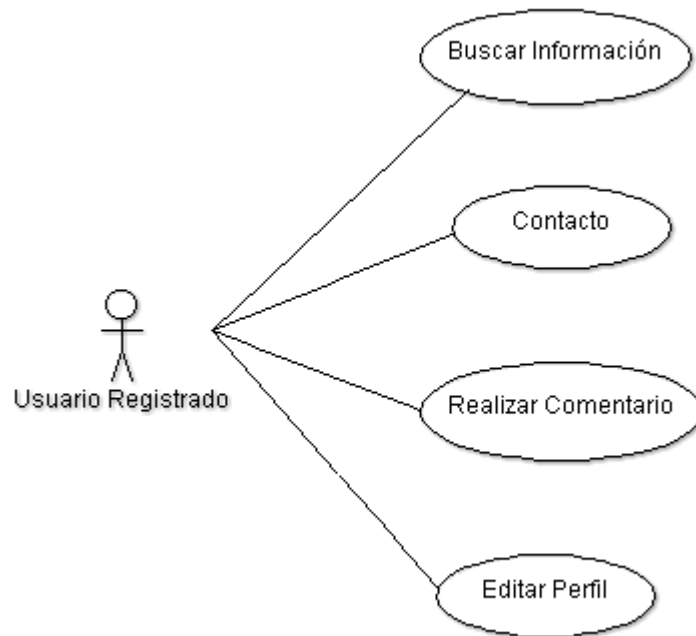


Figura 9. Diagrama modelo caso de uso - Actor: Usuario Editor

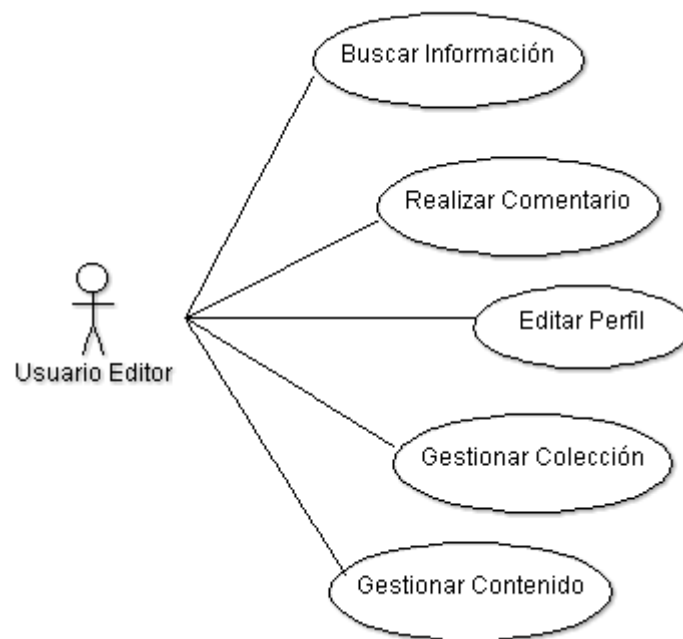


Figura 10. Diagrama modelo caso de uso - Actor: Usuario Administrador

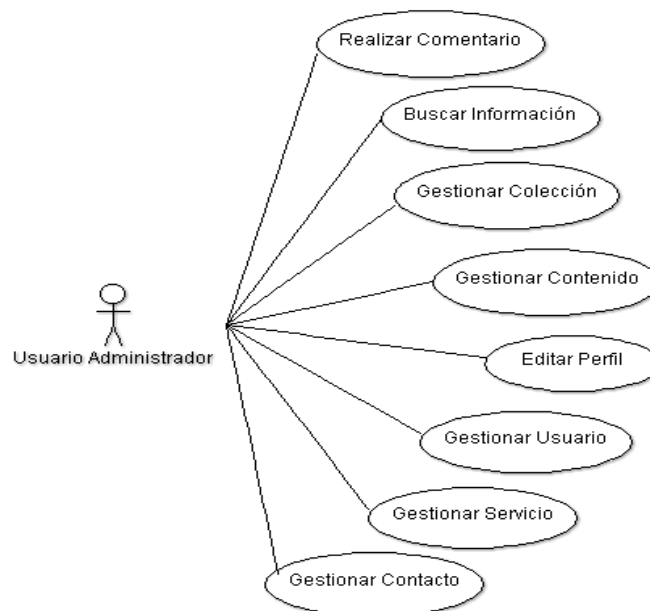


Figura 11. Diagrama de caso de uso - Editar Perfil - Usuario Registrado



Figura 12. Diagrama de caso de uso - Realizar Comentario - Usuario Registrado



Figura 13. Diagrama de caso de uso - Buscar Información - Usuario Editor



Figura 14. Diagrama de caso de uso - Editar Perfil - Usuario Editor

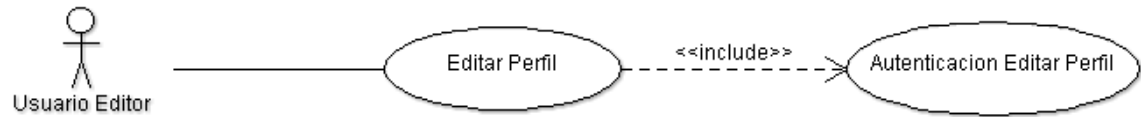


Figura 15. Diagrama de caso de uso - Gestionar Colección - Usuario Editor

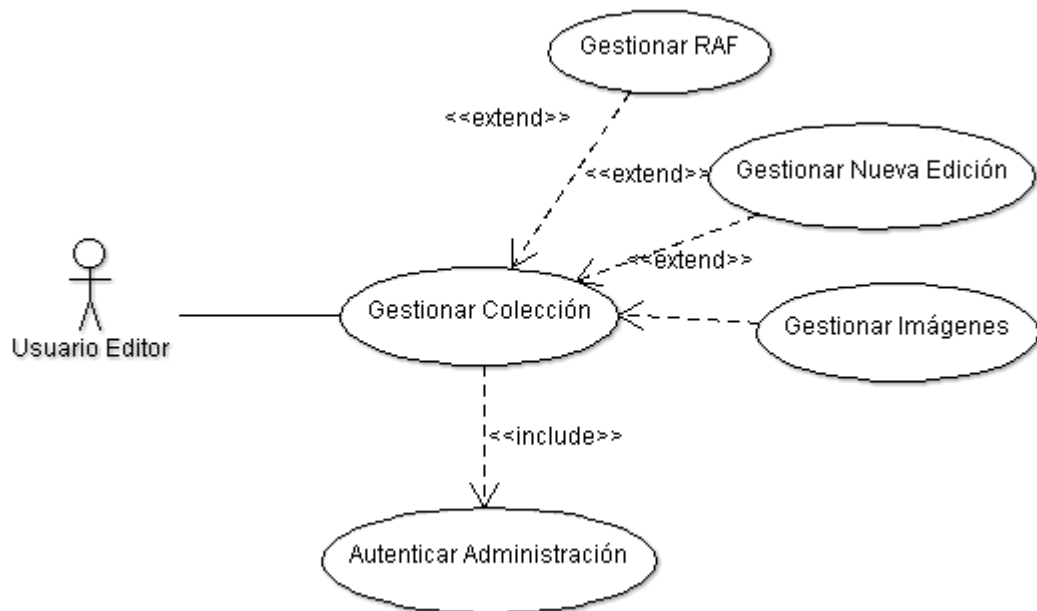


Figura 16. Diagrama de caso de uso - Gestionar Contenido - Usuario Editor

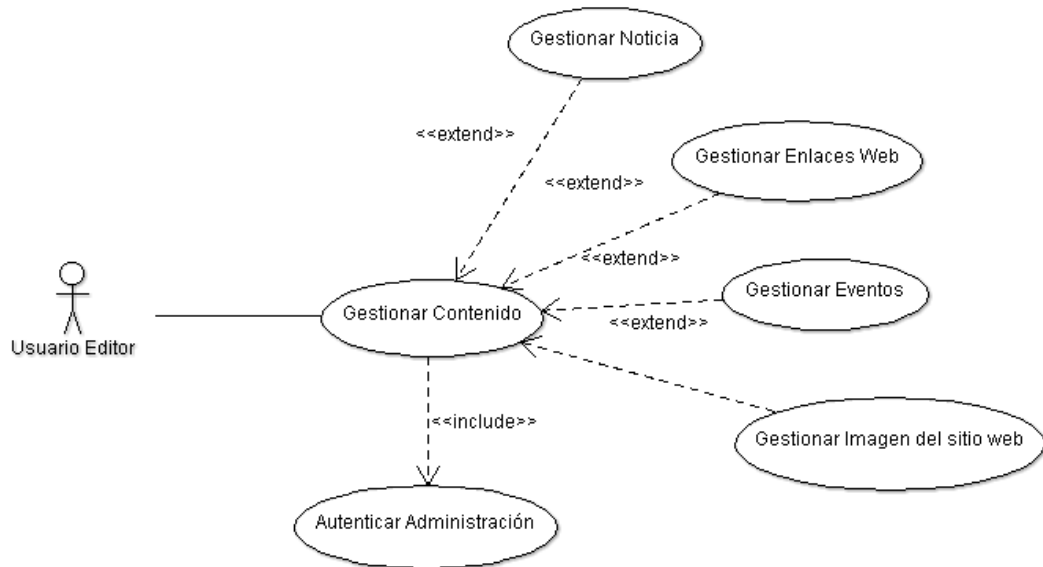


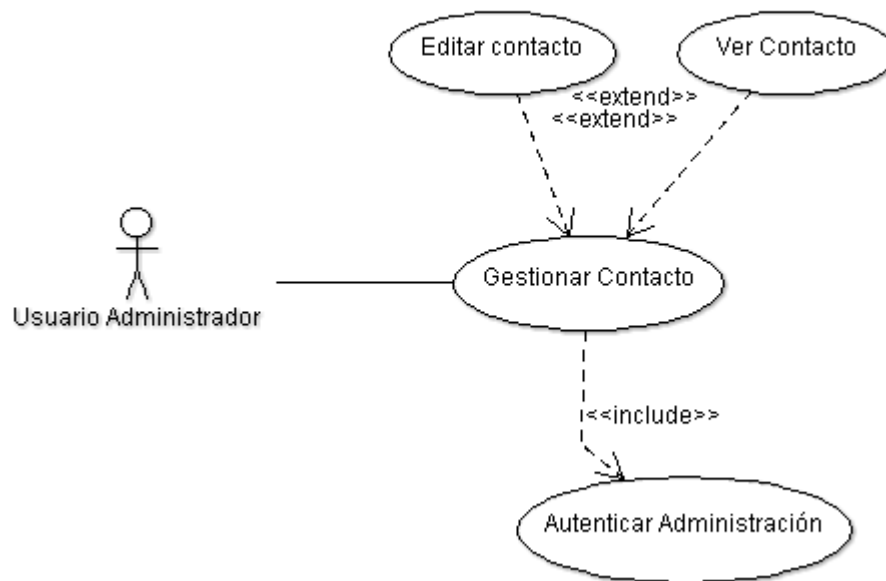
Figura 17. Diagrama de caso de uso - Buscar Información - Usuario Administrador



Figura 18. Diagrama de caso de uso - Editar perfil - Usuario Administrador



Figura 19. Diagrama de caso de uso - Gestionar Contacto - Usuario Administrador



5.3.4. Descripción detallada de los casos de uso

Tabla 5. Buscar Información - Caso de Uso

Caso de uso:	Buscar Información
Actores:	Administrador del Sistema, Usuario Editor de Contenido, Usuario Registrado, Usuario Visitante
Propósito:	Facilitar a los usuarios del sitio web la búsqueda de la información dentro del mismo por medio de la herramienta "Buscar".
Resumen:	Este caso de uso muestra cómo se realiza una búsqueda de información dentro del sitio web por medio de una caja de texto

	en la que se digitan palabras claves para posteriormente mostrar el resultado de esta búsqueda.
Precondición	Los usuarios deben digitar en la caja de texto con nombre “Buscar” palabras claves relacionadas con el tema.
Flujo Principal	El caso de uso se inicia cuando el usuario digita en la caja de texto la información que solicita. Luego a esto se muestran las palabras clave utilizadas así como las secciones del sitio web donde se encontraron coincidencias.
Sub-flujo:	
Excepciones:	E-1 No se encontraron coincidencias, la excepción es mostrada cuando las palabras clave no coinciden con ningún registro en el sitio web.

Tabla 6. Autenticación Administración - Caso de Uso

Caso de uso:	Autenticación Administración.
Actores:	Administrador del Sistema, Usuario Editor de Contenido
Propósito:	Concede a los usuarios del sistema la posibilidad de ingresar y adquirir privilegios de administración.
Resumen:	Este caso de uso está diseñado para ser utilizado por aquellos usuarios con permisos de administración.
Precondición	El usuario debe autenticarse y acceder a la sección administrar que se encuentra ubicada debajo de la opción de editar perfil.
Flujo Principal	Cuando inicia el caso de uso, se muestra dos cajas de texto donde se deben ingresar los datos (usuario y contraseña),

	además de un botón de inicio de sesión que envía los datos para su autenticación.
Sub-flujo:	SF-1 Datos Usuario Se muestra dos cajas de texto para el ingreso de datos personales del usuario. En la primera caja se ingresa el nombre del usuario y en la segunda se ingresa la contraseña
Excepciones:	E-1 Debe especificar su usuario, el campo usuario se encuentra vacío, debe ingresar su nombre de usuario en el campo. E-2 Debe especificar su contraseña, el campo contraseña se encuentra vacío debe ingresar su contraseña en el campo.

Tabla 7. Contacto - Caso de Uso

Caso de uso:	Contacto
Actores:	Usuario Visitante,
Propósito:	Permite a los usuarios no registrados establecer un medio de comunicación con el sitio web
Resumen:	El caso de uso permite al usuario visitante ponerse en contacto con el sitio web mediante una forma que contiene nombre, asunto, email y mensaje.
Precondición	Los usuarios deben acceder al menú de contacto. .
Flujo	El caso de uso inicia cuando un usuario desea establecer una

Principal	comunicación con el sitio web y no ha sido previamente registrado, ingresa en el menú principal al módulo de contacto y allí se muestra un formulario con los campos nombre, asunto, email y mensaje que serán enviados a los correos configurados por el administrador mediante al botón enviar.
Sub-flujo:	<p>SF-1 <i>Nombre</i></p> <p>Muestra un campo para ingresar el nombre del usuario visitante.</p> <p>SF-2 <i>Email</i></p> <p>Muestra un campo para ingresar el correo electrónico del usuario.</p> <p>SF-3 <i>Asunto</i></p> <p>Muestra un campo para ingresar el asunto del mensaje.</p> <p>SF-4 <i>Mensaje</i></p> <p>Muestra un campo para ingresar el contenido del mensaje.</p> <p>SF-5 <i>Enviar</i></p> <p>En esta instancia todos los campos son obligatorios para establecer contacto con el sitio web</p>

Excepciones:	<p>E-1 <i>Debe especificar un nombre</i>, el campo nombre se encuentra vacío, debe ingresar un nombre en el campo.</p> <p>E-2 <i>Debe especificar una dirección de correo electrónico</i>, el campo correo electrónico se encuentra vacío, debe ingresar un correo electrónico en el campo.</p> <p>E-3 <i>Formato de email inválido</i>, los caracteres ingresados deben corresponder a un formato correo electrónico válido.</p> <p>E-2 <i>Debe especificar un asunto</i>, el campo asunto se encuentra vacío, debe ingresar un asunto en el campo.</p> <p>E-3 <i>El contenido del mensaje no puede ser vacío</i>, el campo mensaje se encuentra vacío, debe ingresar un mensaje en el campo.</p>
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 8. Editar Perfil - Caso de Uso

Caso de uso:	Editar perfil
Actores:	Administrador del Sistema, Usuario Editor de Contenido, Usuario Registrado
Propósito:	Permite a los usuarios personalizar el perfil
Resumen:	El caso de uso permite la edición del perfil de usuario respecto a la información personal y profesional, mostrando su contraseña, nombre, apellido, título, información, teléfono, celular, correo, genero, imagen.
Precondición	Los usuarios deben estar registrados y autenticados.

Flujo Principal	Cuando se inicia sesión de usuario, se muestra la información personal y profesional del usuario. El usuario puede modificar la información a su gusto.
Sub-flujo:	<p>S-1 Los campos con un (*) son requeridos</p> <p>Se indica al usuario los campos obligatorios para una transacción exitosa.</p> <p>S-2 Usuario actualizado exitosamente</p> <p>Señala que la transacción cumplió los requisitos y fue actualizada correctamente.</p>
Excepciones:	<p>E-1 El campo (Nombre) es requerido, campo que no puede estar vacío. El usuario debe ingresar su nombre para que la transacción sea exitosa</p> <p>E-2 El campo (Apellido) es requerido, campo que no puede estar vacío. El usuario debe ingresar su apellido para que la transacción sea exitosa</p> <p>E-3 El campo Correo no es válido, los caracteres ingresados deben corresponder a un formato correo electrónico válido.</p>

Tabla 9. Gestionar Contenido - Caso de Uso

Caso de uso:	Gestionar Contenido
Actores:	Usuario Editor, Usuario Administrador
Propósito:	Permite gestionar al usuario editor o administrador contenido en

	el sitio web
Resumen:	Este caso de uso permite a los usuarios editor o administrador gestionar los enlaces web, eventos, imágenes del sitio web, y noticias.
Precondición	Debe ser usuario administrador o editor registrado y autenticado para poder gestionar los contenidos del sitio web.
Flujo Principal	Cuando se inicia este caso se muestra el módulo de gestión de contenidos que contiene la gestión de enlaces web, eventos, imágenes del portal y noticias.
Sub-flujo:	S-1 Noticia gestionada exitosamente Señala que la transacción cumplió los requisitos y fue realizada correctamente.
Excepciones:	E-1 El campo (Titulo) es requerido, no puede estar vacío. El administrador debe ingresar el título para que la transacción sea exitosa E-2 El campo (Contenido) es requerido, no puede estar vacío. El administrador debe ingresar el contenido para que la transacción sea exitosa

Tabla 10. Gestionar Colección - Caso de Uso

Caso de uso:	Gestionar Colección
Actores:	Usuario Editor, Usuario Administrador
Propósito:	Permite gestionar al usuario editor o administrador colecciones de documentos
Resumen:	Este caso de uso permite a los usuarios editor o administrador gestionar colecciones de documentos publicados o inéditos previamente clasificadas

Precondición	Debe ser usuario administrador o editor registrado y autenticado para poder gestionar colecciones del sitio web.
Flujo Principal	
Sub-flujo:	
Excepciones:	<p>E-1 El campo (Tipo Referencia) es requerido, no puede estar vacío. El administrador debe ingresar el Tipo Referencia para que la transacción sea exitosa.</p> <p>E-2 El campo (Titulo) es requerido, no puede estar vacío. El administrador debe ingresar el título para que la transacción sea exitosa.</p> <p>E-3 El campo (Año) es requerido, no puede estar vacío. El administrador debe ingresar el Año para que la transacción sea exitosa.</p> <p>E-4 El campo (Formato) es requerido, no puede estar vacío. El administrador debe ingresar el formato para que la transacción sea exitosa.</p> <p>E-5 El campo (Descripción) es requerido, no puede estar vacío. El administrador debe ingresar la descripción para que la transacción sea exitosa.</p> <p>E-6 El campo (Palabras Clave) es requerido, no puede estar vacío. El administrador debe ingresar las palabras clave para que la transacción sea exitosa.</p>

	<p>E-7 El campo (Estado) es requerido, no puede estar vacío. El administrador debe ingresar el estado para que la transacción sea exitosa.</p> <p>E-8 El campo (Url) es requerido, no puede estar vacío. El administrador debe ingresar la url para que la transacción sea exitosa</p> <p>E-9 El campo (Idioma) es requerido, no puede estar vacío. El administrador debe ingresar el idioma para que la transacción sea exitosa</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 11. Gestionar Usuario - Caso de Uso

Caso de uso:	Gestionar Usuario
Actores:	Usuario Administrador
Propósito:	Gestionar cuentas de usuario con sus respectivos privilegios
Resumen:	Este caso de uso permite al usuario administrador crear cuentas de usuario editor, otorgándole privilegios de edición sobre el contenido del sitio web
Precondición	El usuario debe ser administrador y estar autenticado para gestionar cuentas de usuario editor.
Flujo Principal	Cuando se inicia el caso de uso el modulo gestionar usuario permite crear, editar y eliminar usuarios del sistema
Sub-flujo:	S-1 Usuario gestionado exitosamente

	Señala que la transacción cumplió los requisitos y fue realizada correctamente.
Excepciones:	<p>E-1 Usuario editor no puede crear nuevos usuarios del sistema aunque tenga algunos privilegios de administración.</p> <p>E-2 El campo (Nombre) es requerido, campo que no puede estar vacío. El usuario debe ingresar su nombre para que la transacción sea exitosa</p> <p>E-3 El campo (Apellido) es requerido, campo que no puede estar vacío. El administrador debe ingresar el apellido para que la transacción sea exitosa</p> <p>E-4 El campo (Contraseña) es requerido, campo que no puede estar vacío. El administrador debe ingresar su apellido para que la transacción sea exitosa</p>

5.4. DISEÑO DE BASE DE DATOS

Luego de la categorización de objetos propuesta y teniendo en cuenta las especificaciones del cliente, se realizó un refinamiento en la clasificación debido a que se integró una mayor cantidad de documentos que a futuro podrán ser contemplados para formar parte del archivo virtual. De tal forma que se agregaron nuevas categorías generales para los atributos en común que estos objetos pueden tener, encontrando 28 tipos de documentos entre los que se encuentran: Artículo, Artículo inédito, Audio, Capítulo de libro, Correspondencia, Documento oficial, Documento legal/ Legislación, Estado del Arte, Fotografía, Imagen, Informe

de Avance de Investigación, Informe de Reunión, Informe Final de Investigación, Libro, Material Instruccional, Manual, Nota de trabajo, Nota suelta, Pagina/Sitio web, Patente, Ponencia/Conferencia, Publicación Personal, Repertorio bibliográfico, Reseña, Sub-Producto de Investigación, Tesis de Grado, Traducción, Video.

Las categorías generales que se determinaron para clasificar atributos correspondientes a los objetos son las siguientes:

5.4.1. Datos para todos los documentos

Representan los datos mínimos requeridos en una publicación para su identificación y clasificación dentro de la herramienta como:

- Autor(es)
- Título original
- Palabras clave (máx. 6)
- Descripción
- Idioma
- País de elaboración o publicación
- Año de publicación o elaboración
- Formato de publicación o elaboración

La tabla que se muestra a continuación, presenta la clasificación de objetos indicando los datos correspondientes a cada uno de ellos.

Figura 20. Tabla de Clasificación - Datos para todos

Tipo de documento	Editado/ inédito*	Datos para todos							
		Autor(es)	Título original**	Palabras clave (máx. 6)	Descripción	Idioma	País de elaboración o publicación	Año de publicación o elaboración	Formato de publicación o elaboración
Artículo	E	x	x	x	x	x	x	x	x
Artículo inédito	I	x	x	x	x	x	x	x	x
Audio	E/I	x	x	x	x	x	x	x	x
Capítulo de libro	E	x	x	x	x	x	x	x	x
Correspondencia	I	x	x	x	x	x	x	x	x
Documento oficial	E	x	x	x	x	x	x	x	x
Documento legal/ Legislación	E	x	x	x	x	x	x	x	x
Estado del Arte	E	x	x	x	x	x	x	x	x
Fotografía	E/I	x	x	x	x		x	x	x
Imagen	E	x	x	x	x		x	x	x
Informe de Avance de Investigación	E	x	x	x	x	x	x	x	x
Informe de Reunion	E	x	x	x	x	x	x	x	x
Informe Final de Investigación	E	x	x	x	x	x	x	x	x
Libro	E	x	x	x	x	x	x	x	x
Material Instruccional	E	x	x	x	x	x	x	x	x
Manual	E	x	x	x	x	x	x	x	x
Nota de trabajo	I	x	x	x	x	x	x	x	x
Nota suelta	I	x	x	x	x	x	x	x	x
Página/Sitio web	E	x	x	x	x	x	x	x	x
Patente	E	x	x	x	x	x	x	x	x
Ponencia/Conferencia	E/I***	x	x	x	x	x	x	x	x
Publicación Personal	E/I	x	x	x	x	x	x	x	x
Repertorio bibliográfico	E	x	x	x	x	x	x	x	x
Reseña	E	x	x	x	x	x	x	x	x
Sub-Producto de Investigación	E	x	x	x	x	x	x	x	x
Tesis de Grado	E	x	x	x	x	x	x	x	x
Traducción	E	x	x	x	x	x	x	x	x
Video	E/I	x	x	x	x	x	x	x	x

1- Datos de Publicación acoge datos más detallados de la publicación como:

- Ciudad
- Editorial
- No. de páginas
- ISBN
- Edición No.
- Rango de páginas
- ISSN
- Serie o colección
- Volumen
- Temporada
- Periodicidad temporada
- Publicado en (libro o revista)
- Medio de publicación
- Nivel de Circulación

Figura 21. Tabla de Clasificación - Datos de Publicación

Tipo de documento	Datos de publicación															
	Referencia bibliográfica *****	Ciudad	Editorial	No. de páginas	ISBN	Edición No.	Rango de páginas	ISSN	Serie o colección	Volumen	Temporada	Periodicidad temporada	No.	Publicado en (libro o revista)	Medio de publicación	Nivel de Circulación
Artículo	x	x	x				x	x		x	x	x	x	x	x	x
Artículo inédito																
Audio	x	x														
Capítulo de libro	x	x	x		x	x	x				x	x		x	x	x
Correspondencia																
Documento oficial	x	x	x	x											x	
Documento legal/ Legislación		x		x										x	x	x
Estado del Arte	x	x	x	x										x	x	
Fotografía	x	x							x	x	x	x		x	x	x
Imagen	x	x							x	x	x	x		x	x	x
Informe de Avance de Investigación	x	x	x	x				x					x		x	
Informe de Reunion		x		x											x	x
Informe Final de Investigación	x	x	x	x										x	x	
Libro	x	x	x	x	x	x		x	x				x		x	x
Material Instruccional	x	x	x	x									x		x	
Manual	x	x	x	x		x										
Nota de trabajo																
Nota suelta				x												
Página/Sitio web																
Patente	x	x		x										x	x	x
Ponencia/Conferencia	x	x	x								x	x				
Publicación Personal	x	x	x	x	x	x		x	x					x	x	x
Repertorio bibliográfico	x	x	x	x	x		x	x						x	x	
Reseña	x	x	x	x				x			x	x		x	x	x
Sub-Producto de Investigación	x	x	x	x											x	
Tesis de Grado	x	x	x	x											x	x
Traducción																

5.4.2. Datos de Resumen Analítico de Filosofía (RAF)

Es una categoría exclusiva de este tipo de documento para realizar su categorización como:

- Contenido
- Metodología Conclusiones
- Autor de RAF
- Fecha de realización

Figura 22. Tabla de Clasificación - Datos de Resumen Analítico

Tipo de documento	Datos de resumen analítico				
	Contenido	Metodología	Conclusiones	Autor de RAF	Fecha de realizacion
Artículo	x	x	x	x	x
Artículo inédito					
Audio					
Capítulo de libro	x	x	x	x	x
Correspondencia					
Documento oficial	x	x	x	x	x
Documento legal/ Legislacion					
Estado del Arte	x	x		x	x
Fotografía					
Imagen					
Informe de Avance de Investigacion	x	x	x	x	x
Informe de Reunion	x	x	x	x	x
Informe Final de Investigacion	x	x	x	x	x
Libro	x	x	x	x	x
Material Instruccional	x	x	x	x	x
Manual					
Nota de trabajo					
Nota suelta					
Pagina/Sitio web					
Patente	x	x	x	x	x
Ponencia/Conferencia					
Publicacion Personal	x	x	x	x	x
Repertorio bibliográfico	x	x	x	x	x
Reseña	x	x	x	x	x
Sub-Producto de Investigacion	x	x	x	x	x
Tesis de Grado	x	x	x	x	x
Traducción	x	x	x	x	x

5.4.3. Datos Específicos

Esta categoría está diseñada para algunas publicaciones especiales las cuales pueden ser de carácter personal, hay más de un participante o fue hecha en algún evento específico.

Figura 23. Tabla de Clasificación - Datos Específicos

2		Datos específicos (sólo para algunas categorías)		
3	Tipo de documento	Participantes	Destinatario	Evento
4	Artículo			
5	Artículo inédito			
6	Audio	x		
7	Capítulo de libro			
8	Correspondencia		x	
9	Documento oficial			
10	Documento legal/ Legislación	x	x	
11	Estado del Arte	x		
12	Fotografía	x		x
13	Imagen			x
14	Informe de Avance de Investigación	x		
15	Informe de Reunion	x	x	x
16	Informe Final de Investigación	x		
17	Libro	x		
18	Material Instruccional			
19	Manual	x		
20	Nota de trabajo			
21	Nota suelta			
22	Página/Sitio web	x		
23	Patente			
24	Ponencia/Conferencia			x
25	Publicación Personal			
26	Repertorio bibliográfico			
27	Reseña			
28	Sub-Producto de Investigación	x		
29	Tesis de Grado			
30	Traducción			

5.4.4. Traducción

Traducción es una categoría específica de los objetos que son una misma versión de otro en un lenguaje diferente hecho por un traductor o por el autor original.

Figura 24. Tabla de Clasificación - Traducción

2		Traducción			
3	Tipo de documento	Autor original**	Idioma original**	Traductor	Título de la traducción*
4	Artículo				
5	Artículo inédito				
6	Audio				
7	Capítulo de libro				
8	Correspondencia				
9	Documento oficial				
10	Documento legal/ Legislación				
11	Estado del Arte	x	x	x	x
12	Fotografía				
13	Imagen				
14	Informe de Avance de Investigación				
15	Informe de Reunión				
16	Informe Final de Investigación	x	x	x	x
17	Libro				
18	Material Instruccional	x	x	x	x
19	Manual	x	x	x	x
20	Nota de trabajo				
21	Nota suelta				
22	Página/Sitio web				
23	Patente	x	x	x	x
24	Ponencia/Conferencia				
25	Publicación Personal	x	x	x	x
26	Repertorio bibliográfico				
27	Reseña				
28	Sub-Producto de Investigación	x	x	x	x
29	Tesis de Grado				
30	Traducción	x	x	x	x

Teniendo en cuenta esta clasificación, se propone un modelo de base de datos relacional elaborado en un sistema (open source) multiusuario MySQL, partiendo

del cambio de “Objeto” por “Referencia” como base de la clasificación se da comienzo a la creación de tablas y atributos pertenecientes a cada una.

Este es diseño del esquema de la base de datos asociada a la clasificación de documentos se presenta a continuación:

Figura 25. Vista general del esquema Base de Datos

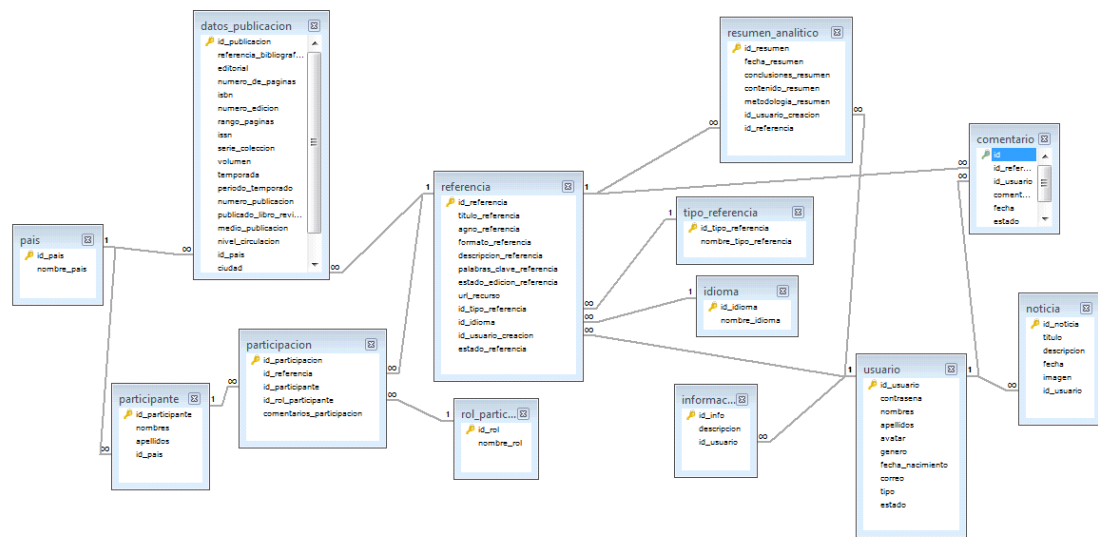


Figura 26. Esquema Base de Datos - parte A

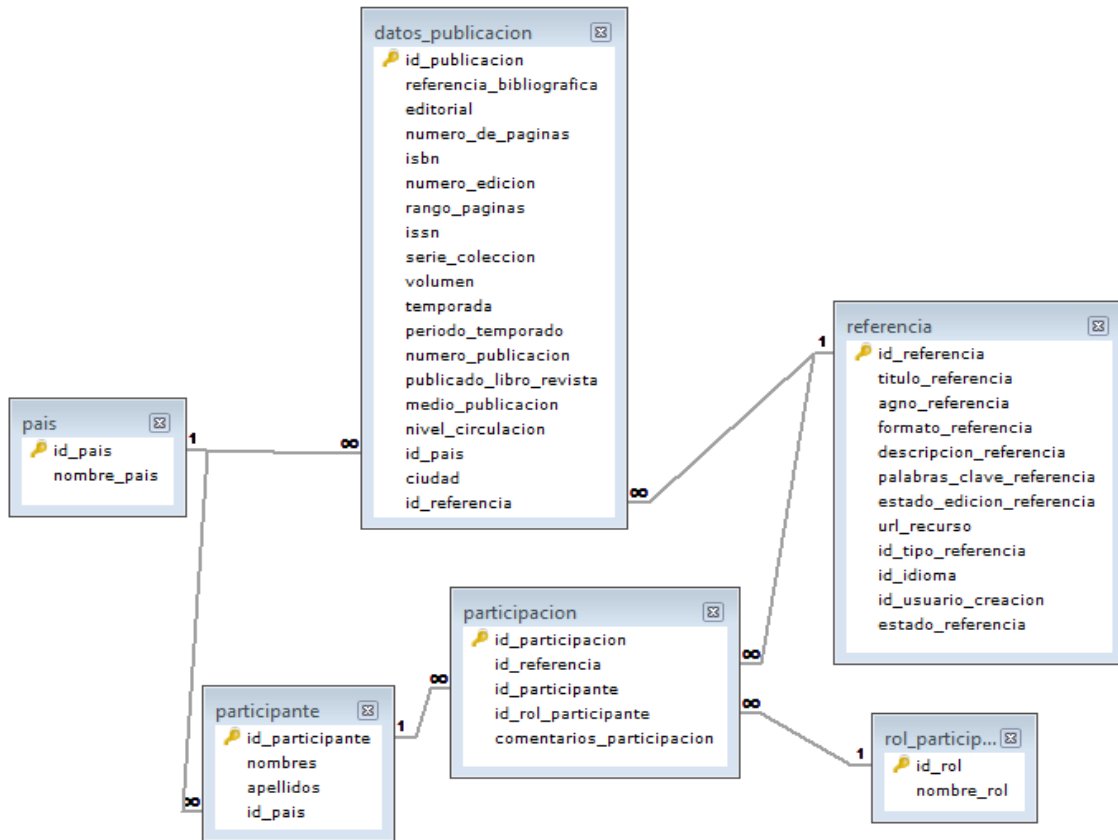
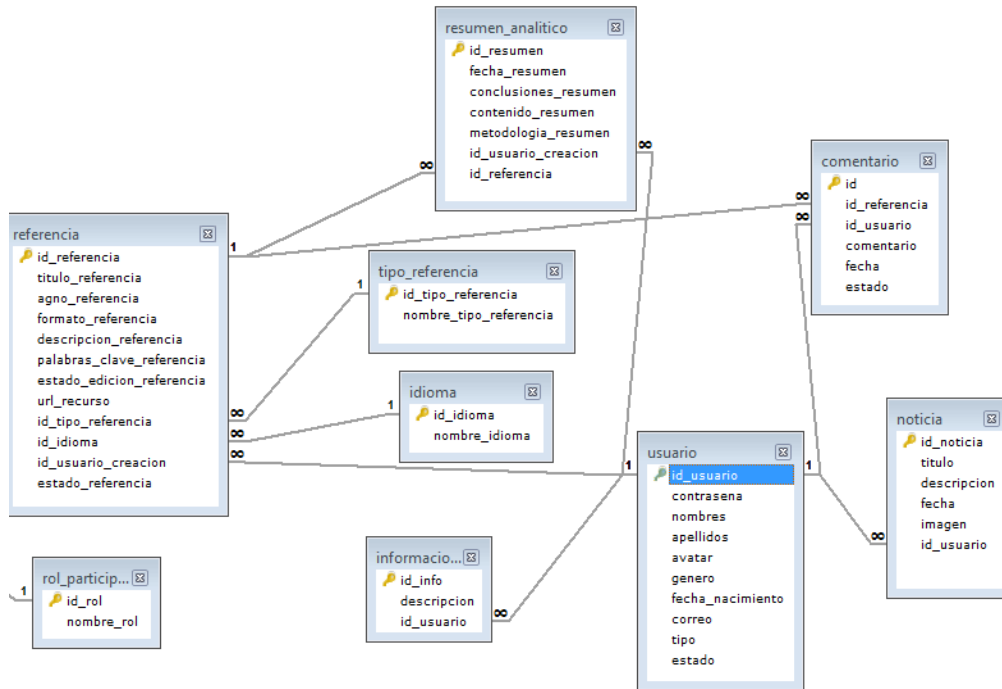


Figura 27. Esquema Base de Datos parte B



5.5. PATRONES DE SOFTWARE A UTILIZAR

De acuerdo a la problemática a tratar, se optó por la búsqueda y adecuación de patrones de diseño que pudieran soportar el desarrollo de la herramienta, de manera que se pudiera estandarizar la forma como se realiza el diseño, establecer elementos reutilizables y especialmente reducir el tiempo de desarrollo al presentar una solución a problemas conocidos que han sido solucionados previamente.

5.5.1. DAO

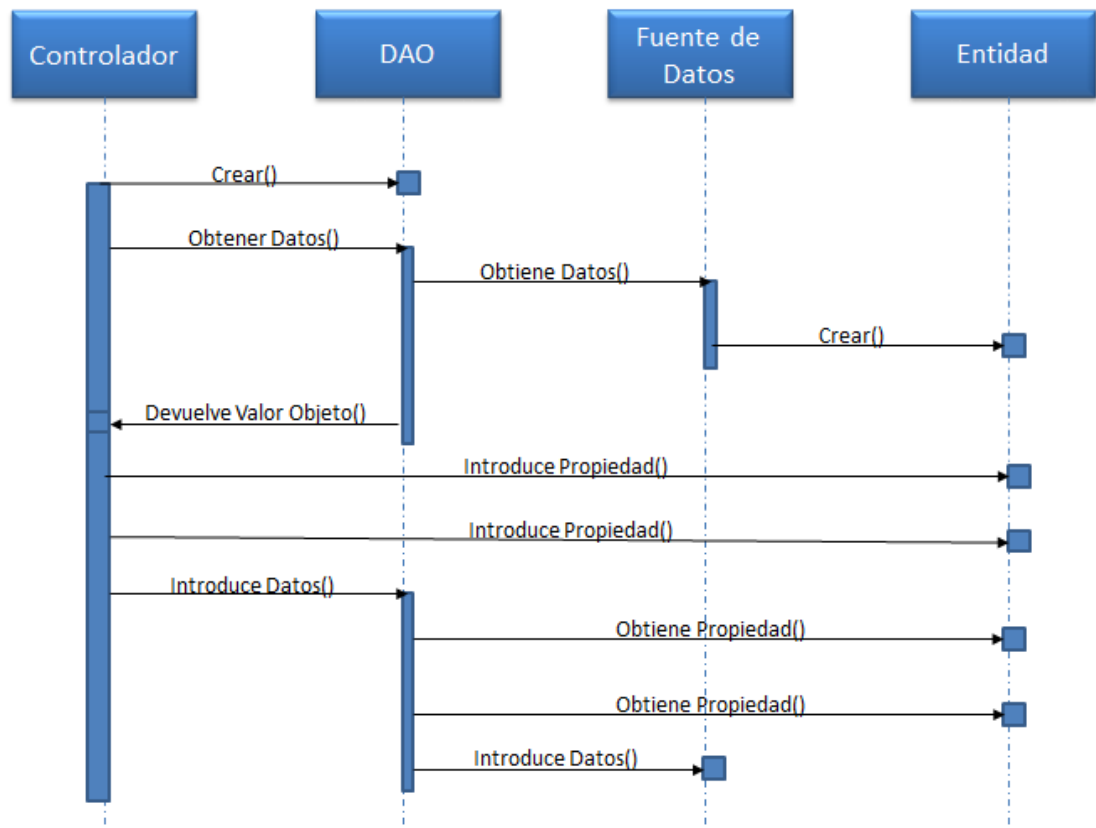
En el desarrollo del proyecto se utilizó el patrón de diseño DAO, el cual proporciona una técnica para la separación de persistencia de objetos y lógica de acceso a datos desde cualquier mecanismo de persistencia o API en particular. El enfoque de Java DAO, proporciona flexibilidad para cambiar mecanismo de persistencia de una aplicación con el tiempo sin la necesidad de lógica de re-

ingeniería en la aplicación que interactúa con el nivel de objetos de acceso de datos.

En el contexto del presente proyecto, el patrón DAO será utilizado para:

- ✓ Abstractar y encapsular los accesos
- ✓ Gestionar las conexiones de la fuente de datos
- ✓ Obtener los datos almacenados

Figura 28. Diagrama de Secuencia DAO

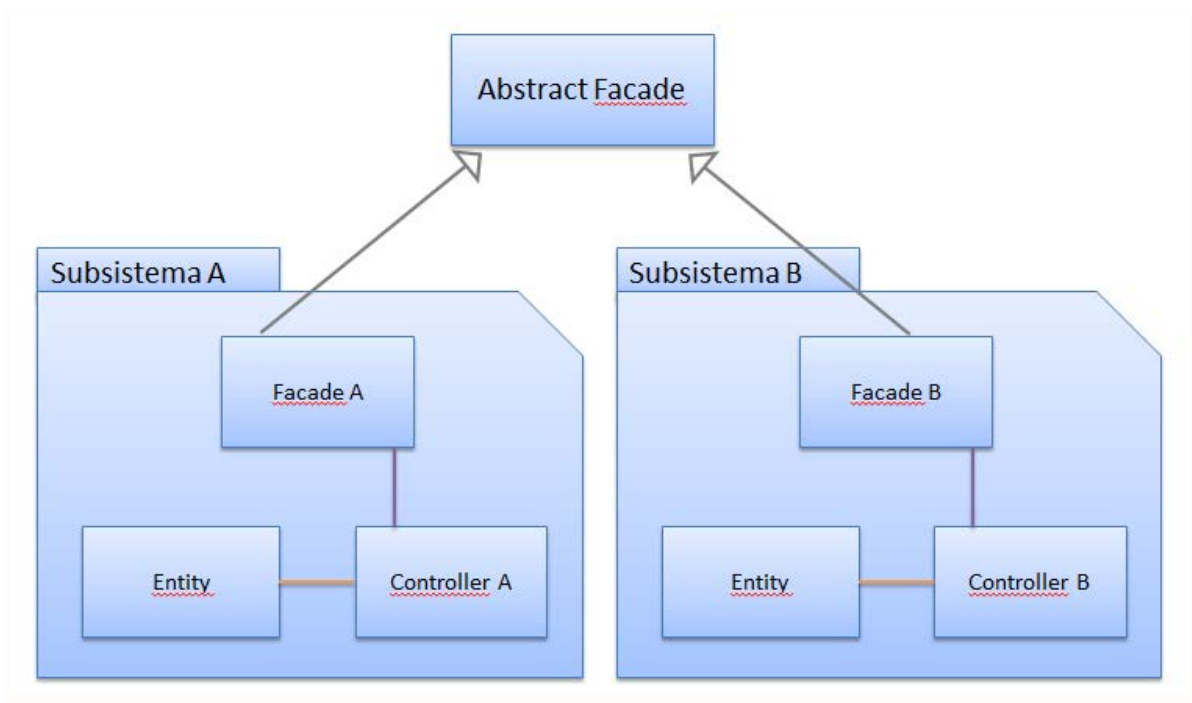


5.5.2. Facade

Es un patrón de tipo estructural que proporciona una interfaz unificada para un conjunto de interfaces de un sistema complejo, haciendo que el manejo de los subsistemas sea más fácil de usar y eliminando dependencia entre cliente y

sistema. En este proyecto el patrón Facade fue seleccionado ya que se requería un patrón para minimizar la comunicación entre módulos del sistema, además de proteger a los clientes de los componentes del subsistema ya que permite modificar componentes del subsistema sin afectar a sus clientes.

Figura 29. Patrón de Diseño Facade



5.5.3. MVC

El patrón de arquitectura MVC (Modelo Vista Controlador) es un patrón que define la organización independiente del **Modelo** (Objetos de Negocio), la **Vista** (interfaz con el usuario u otro sistema) y el **Controlador** (controlador del *workflow* de la aplicación). De tal forma, en el contexto del proyecto se encuentra compuesta por:

- **Modelo (Entidades y SessionBeans)**

Contiene el núcleo de la funcionalidad (dominio) de la aplicación.

Encapsula el estado de la aplicación.

No sabe nada / independiente del Controlador y la Vista.

- **Vista (Páginas JSF)**

Es la presentación del Modelo.

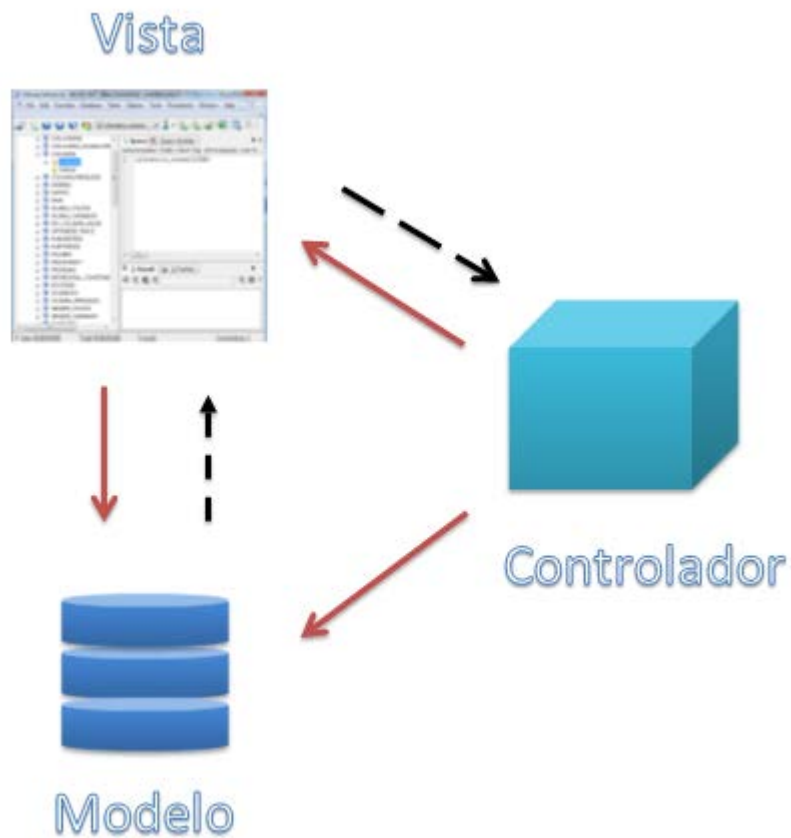
Puede acceder al Modelo pero nunca cambiar su estado.

Puede ser notificada cuando hay un cambio de estado en el Modelo.

- **Controlador (ManagedBeans)**

Reacciona a la petición del Cliente, ejecutando la acción adecuada y creando el modelo pertinente

Figura 30. Modelo Vista Controlador



6. DESARROLLO

En este capítulo se presenta una descripción de la planeación realizada a lo largo del desarrollo del proyecto, así como el proceso y avance de su ejecución. Se realizaron tres hitos fundamentales:

- Diseño de Plantilla del portal.
- Codificación de Funcionalidades del Sistema
- Corrección de Estilos y Mejoras Funcionales

6.1. EJECUCIÓN DE LA PLANEACIÓN DEL CODIFICACIÓN DE SOFTWARE

Antes de iniciar con el proceso de desarrollo e implementación del portal se realizó un plan de actividades, con el fin de estimar y programar el tiempo de producción de cada prototipo a desarrollar. Se adicionó una columna de control con el objetivo de contrastar el tiempo estimado con el tiempo realmente empleado en la realización de las actividades.

Tabla 12 Primera entrega – Diseño e implementación de interfaces

Orden	Actividad	Tiempo de desarrollo (Días)	
		Estimado	Ejecutado
1	Diseño inicial de la interfaz de usuario del sitio a partir de las especificaciones dadas por el cliente	3	4
2	Presentación del diseño inicial para el prototipo	1	1
3	Desarrollo del esqueleto de la interfaz en HTML a partir del diseño creado	3	3
4	Creación de las interfaces de los módulos	5	1

5	Presentación de la plantilla y ajustes de diseño	2	1
---	--------------------------------------------------	---	---

Tabla 13 Segunda Entrega – Diseño e implementación de funcionalidades del sistema

Orden	Actividad	Tiempo de desarrollo (Días)	
		Estimado	Ejecutado
1	Estudio y aprendizaje de las herramientas tecnológicas a emplear.	7	7
2	Diseño y creación de la base de datos	3	5
3	Desarrollo del módulo de presentación del <i>front-end</i>	3	5
4	Desarrollo del módulo de administración del <i>back-end</i>	8	6
5	Presentación de la plantilla y funcionalidades	1	1
6	Ajustes a las funcionalidades	4	4

Tabla 14 Tercera Entrega – Corrección de estilos y mejoras funcionales

Orden	Actividad	Tiempo de desarrollo (Días)	
		Estimado	Ejecutado
1	Ajustes de diseño a interfaces	3	2
2	Verificación de compatibilidad con diferentes navegadores	3	6
3	Verificación de compatibilidad con dispositivos móviles	5	6
4	Presentación de versión alpha	1	1
5	Ajustes a las funcionalidades o diseño	3	2
6	Presentación de versión beta del proyecto	1	1

6.2. IMPLEMENTACIÓN DE DISEÑO DE CLASES

6.2.1. Organización de paquetes

De acuerdo con el diseño realizado, se procedió a realizar su implementación a través de EJB. Conforme a lo establecido por el grupo de Filosofía y Enseñanza de la filosofía, el desarrollo de la herramienta hará parte de un framework que se desarrollará en diversos proyectos.

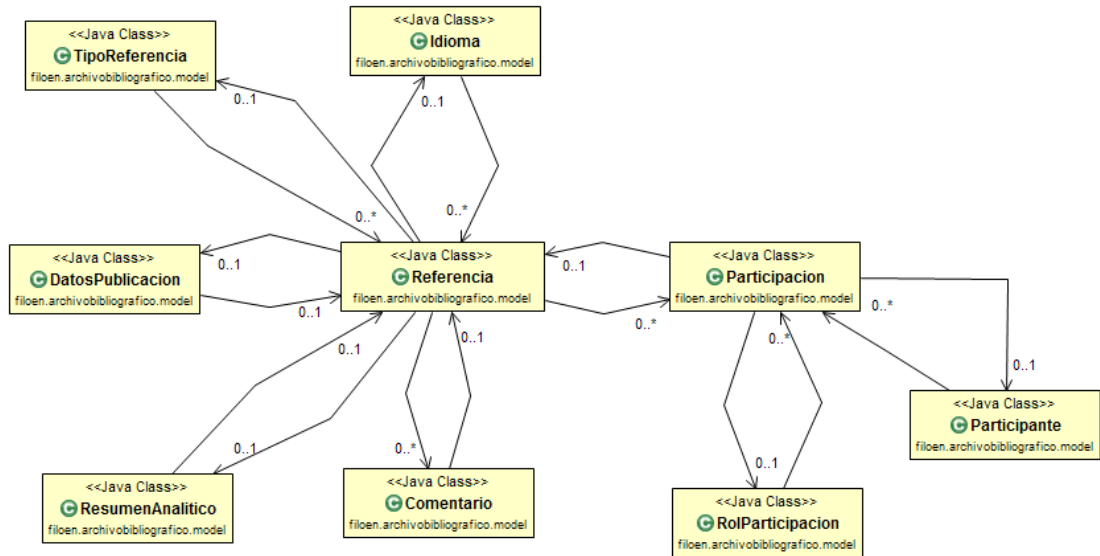
De acuerdo con esto, los paquetes del proyecto quedarán agrupado en el paquete *filoen.archivobibliografico*, que a su vez se encuentra organizado de la siguiente manera:

- Paquete model: contiene las entidades (@Entity) a utilizar
- Paquete dao: contiene los objetos de acceso a datos o DAO (@SessionBean)
- Paquete admin.controller: contiene todas las clases requeridas para la administración del *backend* de la herramienta
- Paquete admin.controller.util: contiene los clases para soporte y conversión de datos requeridos en otras clases
- Paquete portal.controller: contiene todas las clases requeridas para la administración del *frontend* de la herramienta

6.2.2. Clases de modelo de negocio (Entidades y Session BEAN)

Conforme a lo anterior, se generaron las clases de entidad (@Entity) las cuales representan las diferentes tablas de la base de datos. La figura 31 presenta el diagrama de clases de las entidades asociados al registro o referencia bibliográfica

Figura 31 Paquete .model



Las tablas 15 y 16 presenta las clases asociadas al paquete modelo y el paquete, cuyas clases tienen como objetivo encapsular los objetos del lado del servidor que almacena los datos, así como las clases asociadas al proceso de acceso a los datos de las bases de datos.

Tabla 15 Lista y Funciones Paquete modelo

Filen.archivobibliografico.model	Función
Comentario.java	Entidad asociada a la tabla Comentario
DatosPublicacion.java	Entidad asociada a la tabla Datos_Publicacion
Idioma.java	Entidad asociada a la tabla Idioma
InformacionSitio.java	Entidad asociada a la tabla Informacion_Sitio
Noticia.java	Entidad asociada a la tabla Noticia
Pais.java	Entidad asociada a la tabla Pais
Participacion.java	Entidad asociada a la tabla Participacion
Participante.java	Entidad asociada a la tabla Referencia
Referencia.java	Entidad asociada a la tabla Referencia
ResumenAnalitico.java	Entidad asociada a la tabla Resumen_Analitico
RolParticipacion.java	Entidad asociada a la tabla

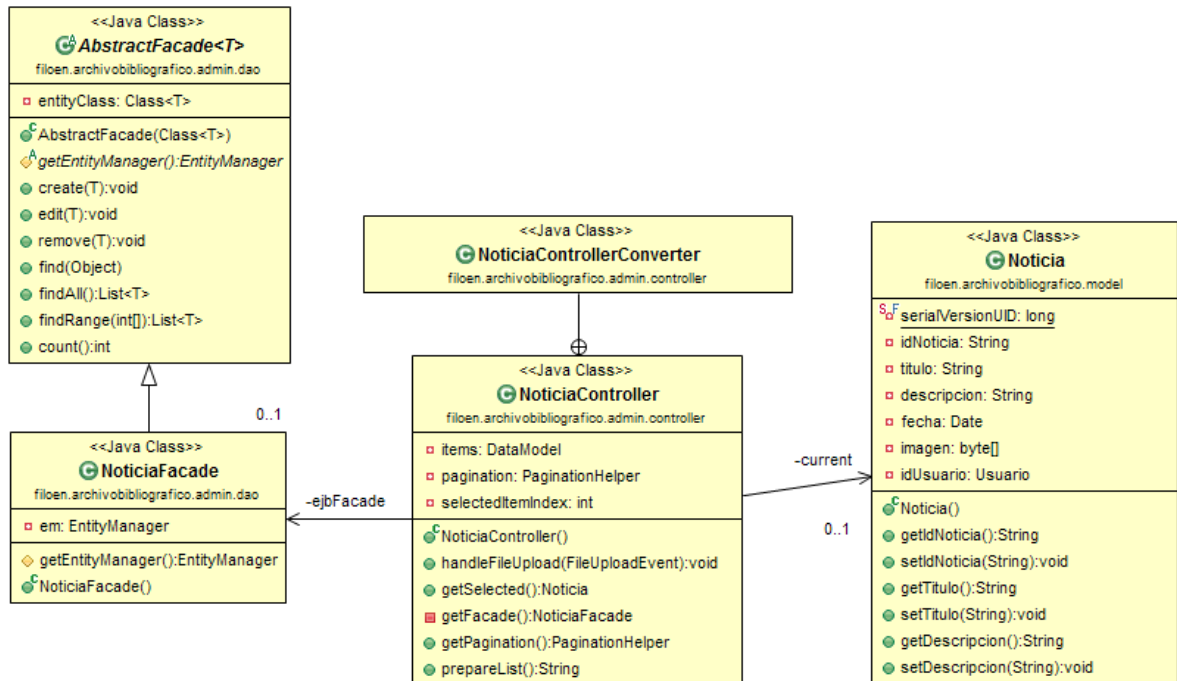
	Rol_Participacion
TipoReferencia.java	Entidad asociada a la tabla Tipo_Referencia
Usuario.java	Entidad asociada a la tabla Usuario

Tabla 16 Lista y Funciones Paquete .admin.dao

Filoen.archivobibliografico.admin.dao	Función
AbstractFacade.java	Clase abstracta con la definición de los métodos principales para el acceso a datos que utilizarán las demás clases Facade.
ComentarioFacade.java	Clase de acceso a datos de la tabla Comentario
DatosPublicacionFacade.java	Clase de acceso a datos de la tabla Datos_Publicacion
IdiomaFacade.java	Clase de acceso a datos de la tabla Idioma
InformacionSitioFacade.java	Clase de acceso a datos de la tabla Informacion_Sitio
NoticiaFacade.java	Clase de acceso a datos de la tabla Noticia
PaisFacade.java	Clase de acceso a datos de la tabla Pais
ParticipacionFacade.java	Clase de acceso a datos de la tabla Participacion
ParticipanteFacade.java	Clase de acceso a datos de la tabla Participante
ReferenciaFacade.java	Clase de acceso a datos de la tabla Referencia
ResumenAnaliticoFacade.java	Clase de acceso a datos de la tabla Resumen_Analitico
RolParticipacionFacade.java	Clase de acceso a datos de la tabla Rol_Participacion
TipoReferenciaFacade.java	Clase de acceso a datos de la tabla Tipo_Referencia
UsuarioFacade.java	Clase de acceso a datos de la tabla Usuario

Teniendo en cuenta que optó por la utilización de los patrones de diseño Facade y DAO, se puede apreciar un ejemplo de cómo estas clases se implementaron en el caso específico de la entidad noticia (ver figura 32).

Figura 32. Ejemplo de clases asociadas a los patrones Facade y DAO

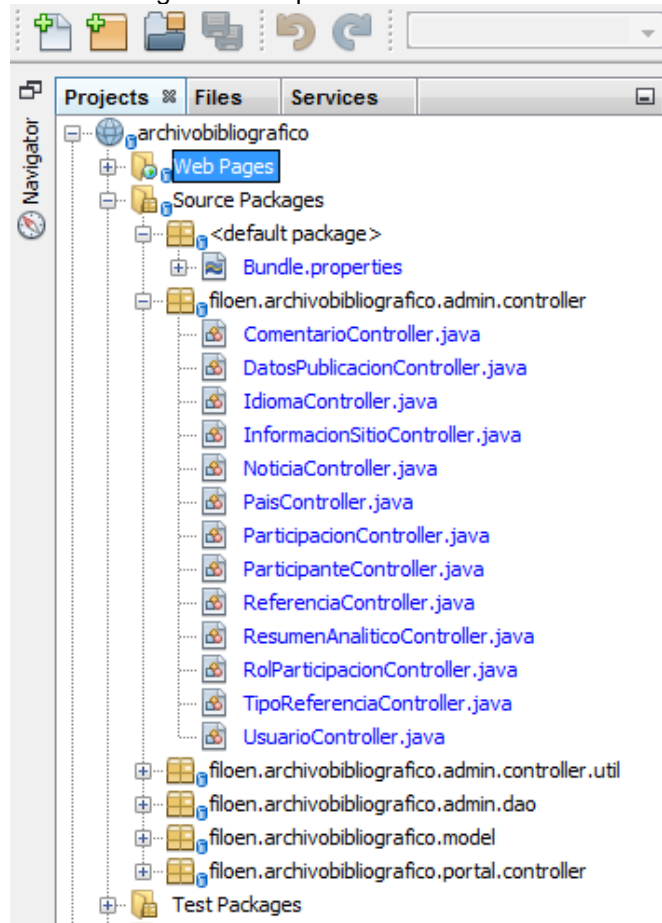


6.2.3. Clases de lógica de negocio (ManagedBean)

Los paquetes `admin.controller` u `portal.controller` contienen todas las clases requeridas para la administración del *backend* y del *frontend* de la herramienta. Estas clases contienen las funciones más comunes que lograron asociadas a:

- La validación de datos de un componente
- Manejo de eventos activados por un componente
- Determinar el procesamiento a realizar y determinar la siguiente página a la que se debe navegar.

Figura 33 Paquete admin.controller



De acuerdo con esto, la tabla 17 presenta la definición de las diferentes clases asociadas a dichos paquetes, las cuales se encargan de interactuar con los campos que se han definido en la capa de vista mediante JSF.

Tabla 17 Lista y Funciones Paquete admin.controller y portal.controller

Filoen.archivobibliografico.admin.controller	Función
ComentarioController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Comentarios
DatosPublicacionController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de DatosPublicacion
IdiomaController.java	Clase de la capa controlador del

	módulo de administración asociado a la gestión de Idioma
InformacionSitioController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de InformacionSitio
NoticiaController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Noticia
PaisController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Pais
ParticipacionController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Participacio
ParticipanteController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Participante
ReferenciaController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Referencia
ResumenAnaliticoController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de ResumenAnalitico
RolParticipacionController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de RolParticipacion
TipoReferenciaController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de TipoReferencia
UsuarioController.java	Clase de la capa controlador del módulo de administración asociado a la gestión de Usuario
Filoen.archivobibliografico.model	Función
InicioBean.java	Clase de la capa controlador del módulo de navegación del portal asociado a los eventos de inicio de sesión
NoticiasBean.java	Clase de la capa controlador del módulo de navegación del portal asociado a las noticias

ReferenciaBean.java	Clase de la capa controlador del módulo de navegación del portal asociado al acceso a la colección digital
Filoen.archivobibliografico.admin.controller.util	Función
JsfUtil.java	Clase para conversiones de tipo asociadas a la presentación valores en JSF
InactividadControl	Clase para el control de la Inactividad en el sistema
LoginAuthentication.java	Clase para el control del manejo de sesiones
PaginationHelper.java	Clase asociada a la paginación de estructuras de datos
DatosPublicacionHelper	Clase para conversiones de tipo y validaciones asociadas a los Datos de publicación de las Referencias

7. PRUEBAS

Una vez implementado el sitio web Archivo Digital Daniel Herrera, se realizaron pruebas de funcionalidad, con el fin de encontrar carencias o errores y de esta forma poder mejorar el entorno.

7.1. PRUEBAS DE FUNCIONALIDAD

Para las pruebas de funcionalidad se utilizó el método de caja negra, el cual consiste en realizar casos de prueba que demuestren que cada función del sistema es operativa. El método recibe su nombre debido a que se evalúan las respuestas de una entrada en una funcionalidad específica del sistema sin conocer cómo funciona el código internamente.

7.1.1. Guía del procedimiento:

Para realizar las pruebas de funcionalidad se tuvieron en cuenta las siguientes actividades:

- Hacer una revisión de las funcionalidades del sistema.
- Basados en la revisión, identificar las funcionalidades más relevantes del sistema.
- Diseñar los casos de prueba según las funcionalidades identificadas.
- Llevar a cabo los casos de prueba utilizando datos de entrada.
- Revisar si las salidas cumplen con la funcionalidad esperada.

Se escogieron las siguientes funciones en el sitio para llevar a cabo las pruebas:

- Autenticación
- Crear Colección
- Crear Noticia

Los casos de prueba se muestran en tablas donde se especifica el nombre del caso, la descripción del procedimiento que se realiza, los valores de entrada, las respuesta a estos valores y el resultado correspondiente. Cuando se obtiene un resultado esperado, el caso es marcado con O.K. Cuando se obtiene un resultado no esperado, el caso es marcado con N.E (No esperado).

Tabla 18 Caso Prueba Autenticación

Caso Prueba	Descripción	Valores entrada	Respuesta	Resultado
Autenticación Correcta	Se ingresan valores correctos de autenticación de usuarios que se encuentren registrados.	Usuario: cahucadi Contraseña: cahucadi	<p>Inicio de sesión valido.</p> <p>Bienvenido, Carlos Carreño</p> <p>Cerrar Sesión Editar Perfil Administrar el Portal</p> <p>ARCHIVO DIGITAL </p>	O.K
Autenticación Correcta	Se ingresan valores correctos de autenticación de usuarios que se encuentren registrados.	Usuario: yolman Contraseña: yolman	<p>Inicio de sesión valido</p> <p>Bienvenido, Yolman Carrillo</p> <p>Cerrar Sesión Editar Perfil</p> <p>ARCHIVO DIGITAL </p>	O.K
Autenticación Incorrecta	Se ingresan valores no válidos.	Usuario: (nulo) Contraseña: (nulo)	Debe digitar su contraseña. Debe digitar su Usuario	O.K

Autenticación Incorrecta	Se ingresan valores no válidos.	Usuario: Diego Contraseña: qwertyu	Usuario o Contraseña incorrectos. Verifique su usuario y contraseña	O.K
---------------------------------	---------------------------------	---------------------------------------	---------------------------------------------------------------------	-----

Tabla 19 Caso Prueba Crear Colección

Caso Prueba	Descripción	Valores entrada	Respuesta	Resultado
Crear colección datos validos	Se ingresan todos los datos que requiere crear una colección correctamente.	Tipo Referencia: libro Título: Epistemología Año:1999 Formato: Impreso Descripción: Libro siglo pasado Palabras clave : libro, Daniel Estado Edición :Editado Url:esquema://máquina/directorio/archivo Idioma : Español	Colección creada exitosamente	OK N.E Se muestra la colección creada pero falta el mensaje.
Crear Referencia Datos Inválidos	Se ingresan datos inválidos en el campo correo. Se deja el campo Usuario vacío.	Tipo Referencia: capitulo libro Título: Referencia nueva Año:2009 Formato: Impreso Descripción: parte libro Palabras clave : XXXX Estado Edición :Editado Url:esquema://máquina/directorio/archivo Idioma : Español	El campo Palabras Clave es requerido.	O.K El campo palabras clave respondió correctamente.

Tabla 20 Caso de Prueba Comentario

Caso Prueba	Descripción	Valores entrada	Respuesta	Resultado

Crear Comentario Datos Validos	Se autentica el usuario administrador luego en colección digital hacer un comentario	Excelente comentario	Comentario creado exitosamente	O.K
Crear Comentario Datos Inválidos	Se autentica el usuario registrado luego en colección digital hacer un comentario	Muy buen comentario	Comentario creado exitosamente	O.K

Las pruebas de funcionamiento realizadas y mostradas en las tablas de casos aplicados al funcionamiento del sitio arrojaron resultados favorables, se puede concluir que cumplen con las funcionalidades esperadas y también permitieron realizar ajustes finales.

8. CONCLUSIONES

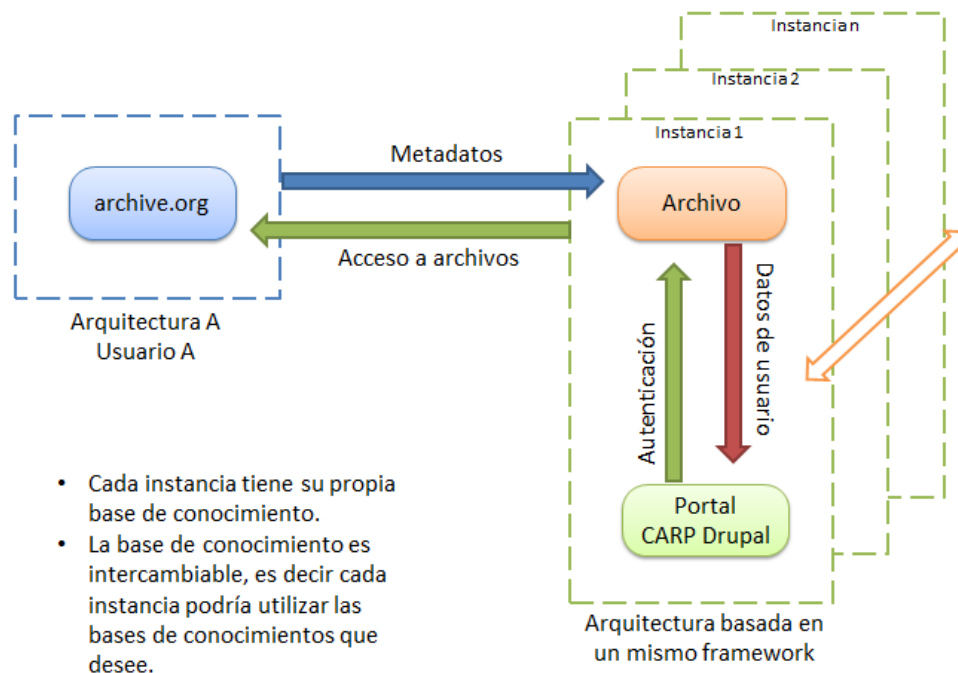
- Se detectó la ausencia de información en línea y sitios web dedicados a pensadores colombianos como Daniel Herrera Restrepo y esto permitió hacer realidad la propuesta de implementación.
- Se logró recopilar una colección importante de escritos propuestos por el Autor del Archivo para su preservación en formato digital.
- La clasificación de documentos realizada para el proyecto Archivo digital para filósofo colombiano permitió establecer una clasificación mínima y fijar las bases para próximos proyectos de archivos digitales.
- La implementación de espacios como el de comentarios de usuarios registrados en las colecciones permite una interacción entre el autor y la comunidad.

9. RECOMENDACIONES

- Algunas recomendaciones para promover el buen desempeño de la herramienta, el flujo de visitantes en el sitio web y mostrar el legado del filósofo:
- Agregar un foro incentivara al usuario al debate sobre temas en fenomenología y ayuda a consolidar una comunidad virtual en fenomenología
- Mediante actualizaciones sobre actividades y eventos relacionados con la fenomenología y la filosofía en Colombia se incentiva al usuario a visitar contantemente la página, esto es posible realizarlo mediante el módulo de noticias.

- Como siguiente fase del proceso del desarrollo, la herramienta deberá incorporar las características definidas por el grupo de investigación en el marco del proyecto “Modelo de gestión del conocimiento en actividades de grupos de investigación”. De acuerdo con esto se deberá busca que en la siguiente fase de desarrollo sea desarrollada de forma que pueda ser presentada como un servicio que se encuentra articulado a la arquitectura propuesta en el proyecto.

Figura 34 Arquitectura propuesta para la gestión de archivo en el grupo de investigación



Fuente: (Carreño, C; Gamboa, S, 2014)

BIBLIOGRAFÍA

Arlandiz, J., (n.d) “Sistemas de Reconocimiento Óptico de Caracteres”, URL= <<http://www.iti.es/media/about/docs/tic/13/articulo2.pdf>>. Consultado el 8 de mayo de 2013.

Çivici, Ç. (n.d.). *PRIME FACES USER ' S GUIDE* (p. 497).

DAO Patrón. (n.d.). Retrieved from <http://www.codefutures.com/java-dao/>

FACADE. (n.d.). Retrieved from <http://design-patterns-with-uml.blogspot.com.ar/20>

Jac. (n.d.). Guía para escribir documentos HTML.

MVC. (n.d.). Retrieved from <http://www.lab.inf.uc3m.es/~a0080802/RAI/mvc.html>

Pavón Maestras, J. (2013). *Java EE – Enterprise Beans (EJB)*. Madrid: Universidad Complutense Madrid.

Pérez, J. E. (2009). *Introducción a JavaScript* (p. 140).

Pérez, J. E. (2009). *Introducción a JavaScript* (p. 140). Disponible en <http://books.google.com/books?hl=en&lr=&id=jCXw02NaYvEC&oi=fnd&pg=PP5&dq=JavaScript&ots=U0pc3-KpQW&sig=auHce3I8INjFvJ4AAqdVXIjPNAo>

Shi, F. (n.d.). *A Brief History of Java*.

Sierra, A. M. (2008). *Programador certificado Java 2: curso práctico* (Segunda Ed., Vol. Segunda Ed, p. 610). Mexico DF: Alfa Omega.

Carreño, C; Gamboa C. (2014). Modelo de gestión del conocimiento en actividades de grupos de investigación. Universidad Industrial de Santander. Bucaramanga.

Sierra, A. M. (2008). Programador certificado Java 2: curso práctico (Segunda Ed., Vol. Segunda Ed, p. 610). Mexico DF: Alfa Omega. Disponible en <http://orton.catie.ac.cr/cgi-bin/wxis.exe/?IsisScript=SUV.xis&method=post&formato=2&cantidad=1&expresion=mfn=010087>

<http://www.thinkepi.net/repositorios-digitales-un-concepto-multiples- visiones#sthash.eIXMrjyb.dpuf>

<http://epf.eclipse.org>

ANEXOS
ANEXO A. DICCIONARIO DE DATOS

comentario

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id	Entero(11)	NO	PRI	(NULO)	auto_incremento	
id_referencia	Entero(11)	NO	MUL	(NULO)		
id_usuario	varchar(50)	NO	MUL	(NULO)		
comentario	Texto	NO		(NULO)		
fecha	datetime	NO		(NULO)		
estado	char(1)	NO		A		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_comentario_ref	referencia	`id_referencia`	`id_referencia`	,
FK_comentario_usu	usuario	`id_usuario`	`id_usuario`	

datos_publicacion

Campos

Campo	Tipo	Nulo	Llave	Defecto	Extra	Comentario
id_publicacion	Entero(11)	NO	PRI	(NULO)	auto_increment	
referencia_bibliografica	varchar(50)	SI		(NULO)		
editorial	varchar(50)	SI		(NULO)		
numero_de_paginas	varchar(50)	SI		(NULO)		
isbn	varchar(30)	SI		(NULO)		
numero_edicion	varchar(30)	SI		(NULO)		
rango_paginas	varchar(30)	SI		(NULO)		
issn	varchar(50)	SI		(NULO)		
serie_coleccion	varchar(50)	SI		(NULO)		
volumen	varchar(30)	SI		(NULO)		
temporada	varchar(50)	SI		(NULO)		

periodo_temporado	varchar(50)	SI		(NULO)		
numero_publicacion	varchar(30)	SI		(NULO)		
publicado_libro_revista	varchar(50)	SI		(NULO)		
medio_publicacion	varchar(50)	SI		(NULO)		
nivel_circulacion	varchar(50)	SI		(NULO)		
id_pais	Entero(3)	SI	MUL	(NULO)		
ciudad	varchar(50)	SI		(NULO)		
id_referencia	Entero(11)	NO	UNI	(NULO)		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_datos_publicacion_pais	pais	`id_pais`	`id_pais`	,
FK_datos_publicacion_referencia	referencia	`id_referencia`	`id_referencia`	

idioma

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_idioma	Entero(2)	NO	PRI	(NULO)	auto_increment	
nombre_idioma	varchar(30)	NO		(NULO)		

informacion_sitio

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_info	varchar(50)	NO	PRI	(NULO)		
descripcion	Texto	SI		(NULO)		
id_usuario	varchar(50)	SI	MUL	(NULO)		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_Informacion	usuario	`id_usuario`	`id_usuario`	

noticia

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_noticia	varchar(50)	NO	PRI	(NULO)		
titulo	Texto	NO		(NULO)		
descripcion	Texto	NO		(NULO)		
fecha	datetime	NO		(NULO)		
imagen	longblob	NO		(NULO)		
id_usuario	varchar(50)	NO	MUL	(NULO)		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_noticia	usuario	`id_usuario`	`id_usuario`	

pais

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_pais	Entero(3)	NO	PRI	(NULO)		
nombre_pais	varchar(50)	NO		(NULO)		

participacion

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_participacion	Entero(11)	NO	PRI	(NULO)	auto_increment	
id_referencia	Entero(11)	NO	MUL	(NULO)		
id_participante	Entero(11)	NO	MUL	(NULO)		
id_rol_participante	Entero(11)	NO	MUL	(NULO)		
comentarios_participacion	Texto	SI		(NULO)		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_participacion_participante	participante	`id_participante`	`id_participante`	,
FK_participacion_referencia	referencia	`id_referencia`	`id_referencia`	,

FK_participacion_rol	rol_participacion	`id_rol_participante`	`id_rol`
----------------------	-------------------	-----------------------	----------

participante

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_participante	Entero(11)	NO	PRI	(NULO)	auto_increment	
nombres	varchar(30)	SI		(NULO)		
apellidos	varchar(30)	SI		(NULO)		
id_pais	Entero(11)	NO	MUL	(NULO)		Nacionalidad"Pais de Origen"

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_participante_pais	pais	`id_pais`	`id_pais`	

referencia

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_referencia	Entero(11)	NO	PRI	(NULO)	auto_increment	
titulo_referencia	varchar(100)	NO		(NULO)		
agno_referencia	Entero(4)	NO		(NULO)		
formato_referencia	varchar(50)	NO		(NULO)		
descripcion_referencia	Texto	NO		(NULO)		
palabras_clave_referencia	varchar(100)	NO		(NULO)		
estado_edicion_referencia	varchar(7)	NO	MUL	Editado		
url_recurso	Texto	SI		(NULO)		
id_tipo_referencia	Entero(2)	NO	MUL	(NULO)		
id_idioma	Entero(2)	NO	MUL	(NULO)		
id_usuario_creacion	varchar(50)	SI	MUL	(NULO)		
estado_referencia	char(1)	SI		A		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info
FK_referencia_idioma	idioma	`id_idioma`	`id_idioma`	,

FK_referencia_tipo	tipo_referencia	`id_tipo_referencia`	`id_tipo_referencia`	,
FK_referencia_usuario	usuario	`id_usuario_creacion`	`id_usuario`	

resumen_analitico

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_resumen	Entero(11)	NO	PRI	(NULO)	auto_increment	
fecha_resumen	date	SI		(NULO)		
conclusiones_resumen	Texto	SI		(NULO)		
contenido_resumen	Texto	SI		(NULO)		
metodologia_resumen	Texto	SI		(NULO)		
id_usuario_creacion	varchar(50)	SI	MUL	(NULO)		
id_referencia	Entero(11)	NO	UNI	(NULO)		

Relaciones Llave Foránea

FK Id	Tabla Referencia	Columna de origen	Columna Referenciada	Extra Info

FK_resumen_analitico_referencia	referencia	`id_referencia`	`id_referencia`	,
FK_resumen_analitico_usuario	usuario	`id_usuario_creacion`	`id_usuario`	

rol_participacion

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_rol	Entero(2)	NO	PRI	(NULO)	auto_increment	
nombre_rol	varchar(50)	NO		(NULO)		

tipo_referencia

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario

id_tipo_referencia	Entero(3)	NO	PRI	(NULO)	auto_increment	
nombre_tipo_referencia	varchar(50)	NO		(NULO)		Descripcion del tipo

usuario

Campos

Campo	Tipo	Nulo	Llave	Por defecto	Extra	Comentario
id_usuario	varchar(50)	NO	PRI	(NULO)		
contrasena	varchar(20)	NO		(NULO)		
nombres	varchar(30)	NO		(NULO)		
apellidos	varchar(30)	NO		(NULO)		
avatar	longblob	SI		(NULO)		
genero	char(1)	NO		(NULO)		
fecha_nacimiento	date	NO		(NULO)		
correo	varchar(50)	NO		(NULO)		
tipo	char(1)	SI		U		
estado	char(1)	SI		A		

