



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 1

Módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a

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

Autora:

Daniela Correderas Ferrer

Tutores:

MsC. Aylin Estrada Velazco

MsC. Sahilyn Delgado Pimentel

Ing. Walter Daniel Camejo López

JUNIO DE 2018

“La Información es poder”

-Bill Gates



DECLARACIÓN DE AUTORÍA

Yo, Daniela Correderas Ferrer, con carnet de identidad 95122704979, declaro ser la única autora del trabajo titulado “Módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a” y autorizo a la Facultad 1 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Declaro que todo lo anteriormente expuesto se ajusta a la verdad, y asumo la responsabilidad moral y jurídica que se derive de este juramento profesional.

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

Autor
Daniela Correderas Ferrer

Firma del Autor

Tutor
Ing. Aylin Estrada Velazco

Firma del Tutor

Tutor
Sahilyn Delgado Pimentel

Firma del Tutor

Tutor
Walter Daniel Camejo López

Firma del Tutor

AGRADECIMIENTOS

Agradezco a mi familia toda, sin ellos, este sueño no hubiese sido posible.

Agradezco a mis tutores Aylin, Sahilyn y Walter gracias por su apoyo y dedicación.

Agradezco a mis profesores de la carrera, en especial Máxora, Ponce y Hubert.

A mis amigas Dalila y Amanda, por ser las mejores amigas.

A mis compañeros de aula y amistades de la universidad.

A todos los que me apoyaron durante la realización de esta tesis.

DEDICATORIA

A mi mamita por ser el motor impulsor de mi vida, a mis hermanos por el amor y el apoyo que me han dado en cada paso.

A mi tía María y mi prima Martha María por ser personas maravillosas que me han dado tanto amor y por ser como mi segunda madre y una hermana más.

A mi familia en general gracias por su apoyo.

RESUMEN

Para localizar y procesar la gran cantidad de información que se genera diariamente en Internet de fuentes confiables de una manera más rápida y eficiente, son utilizados buscadores académicos que han incorporado la funcionalidad de búsqueda avanzada. La presente investigación describe la realización de un módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a, que permita la recuperación de información confiable de carácter científico-académico perteneciente al dominio .cu. Para dar cumplimiento al objetivo planteado en la investigación, se realiza un estudio y análisis de buscadores académicos, se identifican las características por las cuales se encuentran posicionados entre los más usados atendiendo los elementos de filtrado que contiene la búsqueda avanzada, para posteriormente definir los factores a tener en cuenta para la propuesta solución. Se describen como principales tecnologías utilizadas para el desarrollo del módulo: como marco de trabajo Symfony, lenguajes de desarrollo web PHP, CSS y JavaScript, mecanismo de rastreo Nutch, mecanismo de indexación Solr; este proceso fue guiado bajo la metodología AUP en su versión UCI. Se diseñan los artefactos ingenieriles que permiten el desarrollo de la propuesta; y como mecanismos de validación se aplicaron diferentes pruebas de software de tipo: funcional, de seguridad, integración y carga y estrés. Se aplicó la técnica de ladov para identificar el nivel de aceptación de los usuarios hacia el módulo, arrojando que cumple con los requisitos funcionales establecidos.

Palabras clave: búsqueda, herramienta, documentos académicos, Sistemas de Recuperación de Información, Web.

ÍNDICE

INTRODUCCIÓN	12
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA DEL MÓDULO DE BÚSQUEDA AVANZADA PARA EL SUBSISTEMA ACADÉMICO DE LA PLATAFORMA C.U.B.A	17
1.1 RECUPERACIÓN DE INFORMACIÓN	17
1.1.1 <i>Búsqueda y recuperación de información</i>	17
1.1.2 <i>Sistemas de Recuperación de Información</i>	18
1.1.3 <i>Buscadores académicos</i>	21
1.1.4 <i>Análisis de soluciones existentes</i>	22
1.2 TECNOLOGÍAS Y HERRAMIENTAS.....	25
1.2.1 <i>Tecnologías del lado del Cliente</i>	25
1.2.2 <i>Tecnologías del lado del Servidor</i>	26
1.2.3 <i>Entorno de Desarrollo Integrado</i>	27
1.2.4 <i>Sistema de Gestión de Base de Datos</i>	27
1.2.5 <i>Servidor web</i>	27
1.2.6 <i>Mecanismo para la recolección de información</i>	28
1.2.7 <i>Mecanismo para la indexación de la información</i>	28
1.2.8 <i>Herramientas de prueba</i>	28
1.2.9 <i>Lenguaje de modelado</i>	29
1.2.10 <i>Herramienta CASE</i>	29
1.3 METODOLOGÍA DE DESARROLLO	29
1.4 MARCO DE TRABAJO	30
1.5 CONCLUSIONES DEL CAPÍTULO	30
CAPÍTULO 2. MÓDULO DE BÚSQUEDA AVANZADA PARA EL SUBSISTEMA ACADÉMICO DE LA PLATAFORMA C.U.B.A	32
2.1 MODELO DEL DOMINIO	32
2.1.1 <i>Descripción</i>	32
2.1.2 <i>Diagrama de clases del modelo del dominio</i>	33
2.2 ESPECIFICACIÓN DE LOS REQUISITOS DEL SOFTWARE	34

2.2.1	<i>Requisitos funcionales</i>	34
2.2.2	<i>Requisitos no funcionales</i>	35
2.3	PROPUESTA DEL SISTEMA	36
2.4	MODELO DE CASOS DE USO DEL SISTEMA	37
2.4.1	<i>Diagrama de Casos de uso del sistema</i>	38
2.4.2	<i>Especificación de casos de uso</i>	38
2.5	ARQUITECTURA DEL MÓDULO DE BÚSQUEDA AVANZADA	40
2.6	ESTILO ARQUITECTÓNICO.....	40
2.7	PATRONES DE DISEÑO	42
2.8	MODELO DE DISEÑO.....	44
2.8.1	<i>Diagrama de clases del diseño con estereotipos web</i>	44
2.9	DIAGRAMA DE INTERACCIÓN.....	45
2.9.1	<i>Diagramas de secuencia</i>	45
2.10	DISEÑO DE LA BASE DE DATOS.....	45
2.11	DIAGRAMA DE DESPLIEGUE	46
2.12	CONCLUSIONES DEL CAPÍTULO	47
CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL MÓDULO DE BÚSQUEDA AVANZADA PARA EL SUBSISTEMA ACADÉMICO DE LA PLATAFORMA C.U.B.A.		48
3.1	MODELO DE COMPONENTES	48
3.1.1	<i>Diagrama de componentes</i>	48
3.2	ESTÁNDARES DE CODIFICACIÓN UTILIZADOS.....	51
3.3	VALIDACIÓN DEL MÓDULO	52
3.3.1	<i>Pruebas funcionales</i>	52
3.3.2	<i>Pruebas de integración</i>	56
3.3.3	<i>Pruebas de seguridad</i>	56
3.3.4	<i>Pruebas de carga y estrés</i>	58
3.3.5	<i>Aceptación por usuarios</i>	60
3.4	CONCLUSIONES DEL CAPÍTULO	63
CONCLUSIONES		64

RECOMENDACIONES.....	65
BIBLIOGRAFÍA.....	66
ANEXOS.....	70

ÍNDICE DE TABLAS

TABLA 1 COMPARACIÓN DE SISTEMAS HOMÓLOGOS	24
TABLA 2 CLASES DEL MODELO DE DOMINIO	32
TABLA 3 REQUISITOS FUNCIONALES DEL SISTEMA	34
TABLA 4 DESCRIPCIÓN DEL CU 2. REALIZAR BÚSQUEDA ACADÉMICA AVANZADA	38
TABLA 5 VARIABLES EMPLEADAS EN EL DISEÑO DEL CASO DE PRUEBA BASADO EN EL CU “REALIZAR BÚSQUEDA ACADÉMICA AVANZADA”	52
TABLA 6 CASO DE PRUEBA CORRESPONDIENTE AL CU “REALIZAR BÚSQUEDA ACADÉMICA AVANZADA”	54
TABLA 7 RESULTADOS POR ITERACIÓN.....	55
TABLA 8 VULNERABILIDADES DEL SISTEMA	57
TABLA 9 RESULTADO DE LA PRUEBA CON 1000 USUARIOS.....	59
TABLA 10 RESULTADO DE LA PRUEBA CON 1500 USUARIOS.....	59
TABLA 11 CUADRO LÓGICO IADOV	60
TABLA 12 NIVEL DE SATISFACCIÓN.....	62
TABLA 13 VARIABLES CORRESPONDIENTES AL CASO DE PRUEBA: “INTEGRAR NUTCH-SOLR”.	70
TABLA 14 CASO DE PRUEBA: “INTEGRAR NUTCH-SOLR”.	70

ÍNDICE DE IMÁGENES

FIGURA 1 DIAGRAMA DE CLASES DEL MODELO DE DOMINIO.....	33
FIGURA 2 CASO DE USO INICIALIZADO POR EL USUARIO	38
FIGURA 3 ARQUITECTURA DEL MÓDULO PROPUESTO.....	40
FIGURA 4 ARQUITECTURA MODELO VISTA CONTROLADOR	42
FIGURA 5 DIAGRAMA DE CLASES DE DISEÑO CON ESTEREOTIPOS WEB DEL CASO DE USO “REALIZAR BÚSQUEDA ACADÉMICA AVANZADA”	44
FIGURA 6 DIAGRAMA DE SECUENCIA PARA EL CASO DE USO “REALIZAR BÚSQUEDA ACADÉMICA AVANZADA”	45
FIGURA 7 DIAGRAMA DE DISEÑO DE LA BASE DE DATOS	46
FIGURA 8 DIAGRAMA DE DESPLIEGUE.....	46
FIGURA 9 DIAGRAMA DE COMPONENTES DEL MÓDULO DE BÚSQUEDA AVANZADA	49
FIGURA 10 DIAGRAMA DE COMPONENTES DE LA APLICACIÓN WEB	50
FIGURA 11 COMPORTAMIENTO DE LAS ALERTAS	58

INTRODUCCIÓN

El progreso social de la humanidad y el constante desarrollo de las tecnologías han propiciado una mejor forma de gestionar la información, de forma que sea más asequible. Internet más conocida como red de redes es uno de los mayores logros que han contribuido a facilitar el acceso a un gran volumen de información. “Internet es en primer lugar un sistema de comunicación y en segundo lugar una gran enciclopedia” (Santamaría, 2005). Posee sitios y páginas web con datos, dinámicos y volátiles: que permite su modificación en cualquier momento. Proporciona información en diferentes formatos y soportes como textos, imágenes, videos, sonidos, multimedia, entre otros.

Internet es una plataforma que permite la transferencia de información de forma inmediata, lo que ha posibilitado el auge de medios virtuales de comunicación y formatos publicitarios de forma vertiginosa. El creciente flujo de datos ha provocado la necesidad de desarrollar Sistemas de Recuperación de Información (SRI) que posibiliten que la información sea localizada y procesada de modo más rápida y eficiente.

Los SRI poseen herramientas web como los directorios, que son contenedores virtuales en los que se almacenan archivos y otros subdirectorios, atendiendo a su contenido, a su propósito o a cualquier criterio que decida el usuario. Los buscadores como SRI, son útiles para realizar búsquedas sobre temas o sitios específicos y diseñados para agilizar el proceso de búsqueda de información. Poseen su propio motor de búsqueda que se encarga de ir de un sitio a otro obteniendo direcciones y almacenarlas en grandes bases de datos (Varlotta, 2012). Los metabuscadores son otro tipo de herramienta web que se encargan de trasladarse de un motor de búsqueda a buscadores o bases de datos. Devuelven una serie de resultados de la búsqueda o un listado de enlaces para acceder a los resultados individuales de cada buscador (Varela, 2011).

Hoy día una gran cantidad de personas tienen acceso a tecnologías y herramientas que le permiten colocar un artículo o crear un sitio web sin una verificación de que la información que se muestra sea verídica. Para el sector científico y académico, algunas de estas fuentes de información no son confiables y como solución a este problema algunos buscadores han incorporado un motor de búsqueda académico.

Los buscadores académicos son sistemas de información que registran sitios web vinculados a entornos profesionales que contienen: artículos de revistas, investigaciones, ponencias presentadas a congresos, tesis y tesinas, libros y páginas web de instituciones académicas (Leal, 2007). Actualmente son usados por gran parte de la población con el objetivo de encontrar información fidedigna y de cierto rigor científico. El

buscador académico es el encargado de filtrar información admisible y fiable, extraída de diferentes fuentes confiables como editoriales académicas, sociedades profesionales, universidades y otras organizaciones científicas.

Diariamente nuevas informaciones y publicaciones son indexadas por los buscadores académicos; lo que provoca un constante crecimiento que hace que las búsquedas sean excesivas, sin embargo, una gran cantidad de los datos recolectados termina siendo información innecesaria que propicia pérdida de tiempo a los internautas. Como respuesta a esta situación algunos de los buscadores académicos han incorporado la funcionalidad de búsqueda avanzada que permite a los usuarios limitar la búsqueda basándose en diferentes criterios como: título, autor, materia, serie, título de la revista, idioma, tipo de material y fecha de publicación. Estos criterios de búsqueda son metadatos, y proporcionan una información mínima necesaria para identificar un recurso. Hacen posible la búsqueda de información en múltiples lugares de forma simultánea; facilitan la difusión de versiones digitales de un único objeto y permiten establecer restricciones de uso, condiciones de licenciamiento e informan sobre los derechos de autor (Benjumea, 2008).

Existen buscadores académicos, que poseen restricciones de acceso para los usuarios cubanos y muchas de las revistas que indexan necesitan de pago para acceder a ellas, por lo que no siempre el internauta encuentra la información que busca. Actualmente nuestro país para lograr la soberanía tecnológica con el empleo de tecnologías libres se encuentra inmerso en el proceso de informatización de la sociedad, como parte de dicho proceso se han tomado medidas en cuanto a la conectividad de la población a la internet e intranet. Estas situaciones, unidas a la demanda por parte de los centros de estudios del acceso dinámico a información confiable de carácter académico, plantean la necesidad de crear un buscador propio, fácil de utilizar y capaz de responder a la necesidad de los usuarios en el menor plazo de tiempo posible.

Bajo esta premisa la Universidad de las Ciencias Informáticas (UCI) y en particular el Centro de Ideoinformática, desarrolló en el año 2010 el buscador Orion. Con su creación se facilitó la recuperación de información publicada en el dominio .cu, una alternativa para las personas que hoy solo tienen acceso a la intranet nacional. El 13 de julio de 2015 se lanzó la plataforma de Contenidos Unificados para Búsqueda Avanzada, conocida por sus siglas c.u.b.a; que posee como base el motor de búsqueda Orion. La plataforma proporciona servicios como la búsqueda web, de imágenes, documentos, búsqueda avanzada y académica; se enlaza a servicios como son: la cartelera cultural La Papeleta, la plataforma de blogs Reflejos, la enciclopedia EcuRed, el sitio noticioso Cubadebate y la aplicación de mapas interactivos Andariego.

Hoy día en nuestro país existen alrededor de 27 universidades que gradúan anualmente una gran cantidad de profesionales, los que generan una serie de documentación, artículos, tesis y tesinas de carácter científico con un fuerte impacto académico. Además, en Cuba existen 208 revistas científicas que recogen resultados académicos y científicos de contenidos versátiles; relevantes para los profesionales y se encuentran publicadas en la red .cu (Tocororo, 2017). En esta situación juega un papel fundamental la plataforma c.u.b.a que contiene la funcionalidad de búsqueda académica, la cual permite el acceso a documentos y artículos de fuentes confiables que se encuentran en dicho dominio. Debido al cúmulo de información académica que se genera, los usuarios que consultan la búsqueda académica no siempre obtienen los resultados de manera rápida y eficiente. Esta búsqueda se ve limitada pues no contiene varios campos de búsqueda que permitan hacer combinaciones, de tal forma que uno o más términos sean buscados en diferentes índices simultáneamente. Además, no permite excluir un término específico o indicar sobre qué información se quiere realizar dicha búsqueda provocando que el usuario consulte conceptos y bibliografías que no cumplen con sus objetivos, por lo que el logro de los mismos no se realiza con la calidad y en el tiempo requeridos. El subsistema académico con el que se cuenta actualmente, no posee una opción que permita elegir el idioma en que se desea que aparezcan las páginas de búsqueda, por lo cual todos los resultados se encuentran en el idioma español, limitando el área de investigación de los usuarios. La búsqueda académica de la plataforma c.u.b.a no permite restringir los resultados de las búsquedas complejas. Esta situación puede ocasionar que el usuario demore horas consultando páginas, lo que provoca una pérdida del interés y en algunos casos que consulten otros buscadores.

Teniendo en cuenta lo anteriormente planteado se identifica como **problema de investigación** ¿cómo mejorar el proceso de recuperación de documentos académicos en la plataforma c.u.b.a? Como **objeto de estudio** se plantea el proceso de recuperación de la información académica en buscadores, centrando el **campo de acción** la búsqueda avanzada en el subsistema académico de la plataforma c.u.b.a.

Se traza como **objetivo general** desarrollar un módulo de búsqueda avanzada para mejorar el proceso de recuperación de documentos académicos en la plataforma c.u.b.a.

Para alcanzar el objetivo general se definen los siguientes **objetivos específicos**:

1. Elaborar el marco teórico de la investigación a partir de un análisis del estado del arte y de las revisiones bibliográficas de los principales conceptos relacionados con el proceso de recuperación de la información en buscadores, metodologías y herramientas a utilizar en el campo de acción.

2. Diseñar el módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.
3. Desarrollar el módulo de búsqueda avanzada de contenidos académicos para la plataforma c.u.b.a.
4. Validar del correcto funcionamiento del módulo de búsqueda avanzada desarrollado.

Según el problema planteado, se propone como **idea a defender**: el desarrollo de un módulo de búsqueda avanzada para el subsistema académico, mejorará la recuperación de documentos académicos en la plataforma c.u.b.a.

Para el cumplimiento de los objetivos específicos se definen las siguientes **tareas de la investigación**:

1. Análisis de referentes teóricos.
2. Definición de las herramientas, tecnologías y metodología a utilizar para el desarrollo de la propuesta de solución.
3. Construcción de los artefactos asociados a la metodología de desarrollo de software seleccionado.
4. Definición de requisitos de la solución y de la arquitectura de software.
5. Implementación de los requerimientos identificados.
6. Selección de métodos de validación de la solución implementada y aplicación de pruebas a la solución implementada.

Para el desarrollo de la investigación fueron empleados varios **métodos científicos**, tanto teóricos como empíricos, los cuales facilitaron el progreso y correcto avance.

Métodos teóricos:

- **Analítico Sintético:** Fue utilizado en el análisis de la documentación relacionada con el proceso de recuperación de la información, para extraer los elementos más importantes relacionados con el campo de acción.
- **Modelación:** Fue empleado en la modelación del proceso con el objetivo de lograr una mejor visibilidad y entendimiento de cómo se debe desarrollar la solución a la problemática existente, a través de diagramas.

- Histórico-lógico: Utilizado para realizar el estudio, la evolución, comportamiento y las tendencias actuales de las tecnologías y herramientas a utilizar en el desarrollo del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.

Métodos empíricos

- Entrevista: Usado durante el intercambio con el cliente para adquirir información sobre el negocio y los requisitos que se deben cumplir en el desarrollo del sistema.
- Observación: Empleado en la visualización de las opciones de búsqueda avanzada en buscadores académicos similares, identificando requisitos básicos para el desarrollo del módulo de búsqueda especializada para el subsistema académico de la plataforma c.u.b.a.
- Encuesta: Utilizado para la recopilación de información en la aceptación por los usuarios, por el método Técnica de ladov.

La estructura del trabajo de diploma consta de índice, resumen, introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y anexos.

Capítulo 1. Fundamentación teórica del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a: Se realiza la búsqueda y análisis de la información relacionada con los procesos de recuperación de la información mediante metabuscadores, así como las tendencias actuales y principales autores relacionados. Se presentan las características más significativas de los elementos que componen la base tecnológica necesaria para el desarrollo de la solución.

Capítulo 2. Módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a: Se exponen las características del sistema, incluyendo los requisitos funcionales y no funcionales, patrones de diseño y arquitectura utilizados; algunos de los artefactos que requiere la metodología de desarrollo utilizada.

Capítulo 3. Evaluación del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a: Se presentan los diseños de casos de prueba a utilizar en la validación del sistema y se analizan los resultados de las pruebas realizadas que permiten evaluar la calidad de la propuesta de solución.

Capítulo 1. Fundamentación teórica del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a

En este capítulo se exponen criterios valorativos sobre los sistemas de recuperación de información, los conceptos asociados al dominio de la investigación y se realiza un análisis del estado del arte que la precede. Se representan los motores de búsquedas similares con sus características, ventajas y desventajas, y con los algoritmos fundamentales que implementan. Se establece la base tecnológica que será empleada, y se despliegan las características que presenta hasta el momento el buscador c.u.b.a en cuanto a la recuperación de información.

1.1 Recuperación de información

La información constituye hoy la base del desarrollo de las esferas de la sociedad. Compartir la información a través de la red es elemental, permite interactuar de forma más rápida y tener acceso a un gran cúmulo de ella a partir del uso de herramientas que facilitan su recuperación. Para un correcto uso de la información esta debe gestionarse de manera efectiva.

1.1.1 Búsqueda y recuperación de información

La búsqueda de información es un conjunto de operaciones que tiene como objetivo poner al alcance de los investigadores la información que dé respuesta a sus preguntas (Fanego, 2007). La gestión de la información es un proceso que incluye operaciones como extracción, manipulación, tratamiento, depuración, conservación, acceso y colaboración de la información adquirida a través de diferentes fuentes.

Para explicar qué es la recuperación de información (RI) se han seleccionado los siguientes conceptos (Giordanino, 2010).

Salton(1983): “es un campo relacionado con la estructura, análisis, organización, almacenamiento, búsqueda y recuperación de información”.

Baeza-Yates (1999): “la Recuperación de Información trata con la representación, el almacenamiento, la organización y el acceso a ítems de información”.

Croft (1987): “el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la resolución del problema planteado. En estas tareas desempeñan

un papel fundamental los lenguajes documentales, las técnicas de resumen, la descripción del objeto documental”.

Korfhage (1997): “la localización y presentación a un usuario de información relevante a una necesidad de información expresada como una pregunta”.

La RI produce como salida un conjunto de documentos cuyo contenido satisface potencialmente una necesidad, no obstante, no se da la información deseada por el usuario, sino que indica los documentos que sean significativamente relevantes. En el proceso de RI se necesitan diferenciar los documentos relevantes y rechazar todos los documentos irrelevantes. Actualmente la Web posee un gran tamaño, se estima que presenta decenas de exabytes de información y crece permanentemente. Ante semejante volumen de información, existen mecanismos que utilizan técnicas o modelos que ayudan en la realización del proceso de recuperación, algunos de estos modelos son: booleano, probabilístico y vectorial.

En la recuperación de información se emplean ordenadores, con bases de datos u otros sistemas automáticos o semiautomáticos de procesamiento de la información que gestionan información de cualquier tipo, dígase textos, videos, imágenes; mediante el uso de información textual. Es desarrollada en un contexto donde los usuarios sienten la necesidad de descubrir entidades que satisfagan más sus necesidades de búsqueda; teniendo en cuenta que características y necesidades poseen, por lo cual un solo sistema no las satisface a todas.

1.1.2 Sistemas de Recuperación de Información

Se puede definir qué es un sistema de recuperación de varias maneras, según Salton es definido como un software que trata la representación, almacenamiento, organización y acceso de elementos de información (Varela, 2011). Es un software capaz de almacenar, recuperar y mantener información (Kuna, y otros, 2013). Los SRI: poseen la capacidad para gestionar información, entendiéndose por gestión a su recuperación, almacenamiento y mantenimiento para diversos fines según el contexto de su aplicación (Kuna, y otros, 2014). En función de los objetivos de la investigación, la autora decide guiar la investigación por el concepto brindado por Kowalski en 1998, se refiere a los sistemas como herramientas de recuperación de información que permiten el acceso a la información alojada en la web, así como los servicios que se brindan. Su relevancia en el mundo de la navegación por internet está determinada debido a que son esencialmente rastreadores de información que luego es almacenada y posteriormente accesible a través de consultas y

filtros y ordenada según algoritmos matemáticos usados para calcular la relevancia de los resultados obtenidos (Rodríguez, y otros, 2016).

Generalmente los SRI comparten como características comunes de su arquitectura las siguientes:

- La **Interfaz**: es la manera que tiene el usuario de interactuar con la interfaz del sistema, al introducir consultas en un lenguaje preestablecido y el sistema muestra las respuestas.
- El **Sistema de Formulación de Consultas**: es el procesamiento que convierte las consultas realizadas en lenguaje natural a consultas entendibles por los sistemas informáticos.
- El **Mecanismo de evaluación de consultas**: devuelve de acuerdo con un criterio de relevancia los documentos que satisfagan a la consulta introducida, obtenidos por la comparación de documentos representados en el sistema de información con la consulta procesada.

Entre los SRI se encuentran: los buscadores, los directorios y los metabuscadores, que se describen brevemente a continuación:

Buscadores o motores de búsqueda: son sistemas de recuperación de la información que permiten la búsqueda de información sin tener en cuenta una jerarquía. Recorren la web de manera automática a través de las bases de datos. Una de las ventajas que poseen es que comprueban si los documentos referenciados han tenido alguna modificación. Su manera de buscar es mediante una consulta introducida por un usuario. Uno de los inconvenientes que poseen los buscadores son: con el crecimiento de la web algunos buscadores no son capaces de indexar toda la información, lo que provoca que se quede fuera de los índices de los buscadores y por tanto que no sea accesibles para los usuarios. Existen nuevas tendencias de indexar que hacen más probable que refleje como resultado de la búsqueda un sitio comercial, que sitios de fuentes confiables. No se puede obviar que los buscadores son más fáciles de usar, la carga y las respuestas son rápidas, una fiabilidad y precisión en los resultados y proporcionan la información más actualizada y organizada (Fernández, 2013).

Según su arquitectura los buscadores pueden ser de dos tipos: de propósito general y de propósito especial. Los primeros son los que abarcan toda la web. Actualmente los más conocidos son Google, Yahoo!, Ask, Altivista, Excite, Lycos y Bing. Los buscadores de propósito especial se centran en documentos que pertenecen a un dominio específico como puede ser una temática determinada o documentos de alguna

organización. Ejemplo de buscadores es Medical World Search buscador encargado de la búsqueda y descripción de técnicas médicas (Giordanino, 2010).

Los componentes principales de los motores de búsqueda son: el *crawler* que es el encargado de navegar por toda la web para la recuperación de documentos y de sus enlaces, el generador de índice con la función de incluir dichos documentos en la base de datos, el índice que es como tal la base de datos donde el buscador almacena la información que es recuperada por el *crawler*. El motor de búsqueda que recoge las consultas que el usuario ha introducido en un campo de texto donde las preprocesa y las contrasta con los datos contenidos en el índice y la interfaz (Giordanino, 2010). En esta categoría se encuentra la plataforma c.u.b.a desarrollado en la UCI.

Directorios: son herramientas que organizan según una jerarquía, que va de las categorías más específicas a las más generales. Los documentos que se poseen son clasificados por humanos que pueden ser los propios autores. Permiten además la búsqueda por palabras claves, puede reducir la búsqueda a categorías particularmente relevantes, lo cual equivale a que los documentos que se obtengan sean también relevantes (Martínez, 2004). Los directorios más usados en la actualidad son Terra, The Library Channel, looksmart y el Open Directory Project DMOZ, y existen periódicos que utilizan sus ediciones digitales con formato de directorios un ejemplo de ellos son El Mundo y El País. Los directorios no utilizan como mecanismos para recorrer la red una araña u otro mecanismo automático para la búsqueda de informaciones, sino que son operados por humanos. Si se desea localizar información acerca de un tema en general, es efectivo usar directorios, pues proporciona documentos relevantes sobre el tema y habrá enlaces a otros con información relacionada (Giordanino, 2010).

Metabuscadore: son motores de búsqueda, que no poseen una base de datos propia por donde indexar documentos. Realizan un proceso de selección y elaboración de la lista de resultado muy complejo, pero su interfaz permite consultar varios motores de búsqueda de forma simultánea, es decir, la petición enviada por el usuario es enviada a diferentes buscadores y después le muestra los resultados y eliminan los duplicados. Los metabuscadores también emplean a los buscadores especializados lo que posibilita mejorar los resultados, evitando la información irrelevante que se obtenga de otros buscadores. Al tener diferentes fuentes tiene menor variabilidad en los resultados ya que se favorece de buscadores que ofrecen resultados más estables (comparativo, 2009). A pesar de lo anterior, para los metabuscadores la selección del buscador correcto no es nada sencillo, pues una vez que son seleccionados, el problema es el número de

documentos a devolver y cómo combinar los resultados de diferentes buscadores. Para obtener los resultados se tiene en cuenta sus características y forma de evaluación en una lista ordenada por relevancia. Existen varios metabuscadores, como son: Zapmeta, Zuula, Soovle, Metacrawler, Yippy, Entireweb, excite, entre otros (Giordanino, 2010).

Los metabuscadores poseen una arquitectura compuesta por cinco componentes:

- **Interfaz de usuario:** es donde se recogen las consultas introducidas por el usuario y los resultados de la búsqueda.
- **Selector de la base de datos:** es el encargado de seleccionar los buscadores que darán la respuesta a la consulta introducida por el usuario.
- **Selector de documentos:** su objetivo es tratar de obtener el mayor número de documentos relevantes, evitando recuperar los no relevantes.
- **Emisor de consultas:** se encarga de establecer la conexión con el buscador y enviarle la consulta, y obtener los resultados.
- **Agrupador de resultados:** agrupa los resultados de los diferentes buscadores y los combina en una lista, usando algún tipo de criterio para establecer el orden de la lista (Varela, 2011).

1.1.3 Buscadores académicos

Dada la cantidad de información que circula en Internet, es difícil saber con certeza que la información que algunos buscadores proporcionan es fidedigna. Debido a lo anterior se crearon los buscadores académicos que permiten indexar fuentes confiables como lo son libros, revistas científicas, tesis, páginas web de instituciones académicas e investigaciones. Constituyen una herramienta para ayudar a estudiantes, profesores e investigadores a obtener información relevante y de carácter científico. Los documentos son producidos por universidades, centros de investigación, editoriales académicas, bibliotecas y repositorios digitales (Leal, 2007).

Estos buscadores incorporan periódicamente novedosos materiales, por lo cual poseen un gran volumen de información, debido a esto la mayoría ha incorporado una búsqueda avanzada que facilita al usuario el uso de estos buscadores. Recopilan parcialmente los mismos documentos que las bases de datos académicas comerciales, pero en proporciones dispares, según las especialidades. Entre los tipos de

documentos que son recuperados por estos buscadores académicos están páginas web y documentos de toda clase publicados en sitios de instituciones académicas o científicas; artículos de publicaciones científicas, ya se trate de publicaciones abiertas o de pago. También recuperan trabajos académicos, tales como tesis doctorales o tesis de licenciatura, documentos depositados en repositorios científicos, patentes, libros por acuerdos con editoriales comerciales o a través de acuerdos con bibliotecas. Además, poseen interfaces fáciles de interactuar, proporcionan información en varios idiomas (Leal, 2007). Actualmente existen muchos buscadores académicos como son: SciELO, Dialnet, World Wide Science, Google Scholar, Scholarpedia, Springer Link, Refseek, entre otros.

1.1.4 Análisis de soluciones existentes

Según el Centro de Recursos para el Aprendizaje y la Investigación en la actualidad la utilización de los buscadores académicos ha incrementado siendo más utilizados los siguientes:

SciELO: es una biblioteca virtual de revistas científicas brasileñas en formato electrónico. La biblioteca funciona con la metodología SciELO. La metodología prepara, almacena, divulga y evalúa publicaciones científicas. Fue desarrollado como una investigación experimental sobre la publicación electrónica y como una investigación operativa con el objetivo de establecer y ampliar la publicación electrónica en Brasil, América Latina y el Caribe para mejorar el control, divulgación y evaluación de la literatura científica. Como producto principal cuenta con normas, guías, manuales, programas de computación y procedimientos operacionales para la preparación de publicaciones periodísticas-científicas en formato electrónico. La interfaz de búsqueda avanzada de SciELO permite añadir los campos de búsqueda que el usuario necesite, buscar en todos los índices usando los operadores booleanos (AND, OR AND NOT) y la combinación de los operadores con un índice que determine el usuario entre el año de publicación, autor, financiador, periódico, resumen y título (SciELO, 1998).

DIALNET: es un buscador que recopila y proporciona acceso a documentos publicados en España, en idioma español o en cualquier otro. Creado en la Universidad de Rioja que integra entre sus contenidos los artículos de revistas, capítulos de monografías colectivas, tesis doctorales, libros, entre otros. Posee una gran base de datos de artículos científicos hispanos accesibles de modo gratuito en Internet. Como funcionalidades incorporadas al buscador presenta: mostrar información sobre disponibilidad de las revistas en la Biblioteca Nacional y búsqueda avanzada. Esta búsqueda permite obtener cualquier documento indexado acotándolo por el título, resumen, autores o años de publicación y posibilidad de marcar y

seleccionar documentos, enviar las selecciones a una cuenta de correo, importarlas a gestores bibliográficos, entre otras. Para ser uso de sus servicios no es necesario que el usuario esté registrado, sus usuarios pueden seleccionar sus revistas favoritas para recibir correos electrónicos o alertas informativas. Se han creado actualizaciones como son DIALNET Plus que proporciona una serie de servicios de valor añadido (Dialnet, 2017).

World Wide Science: es un buscador científico internacional que permite acceder a la información científica y técnica, la recuperación de referencias bibliográficas de base de datos científicas distribuidas por diferentes organismos. Ha sido creado de manera conjunta por la Biblioteca Británica y el Departamento de Energía de Estados Unidos en conjunto con otras organizaciones del ámbito de la ciencia y tecnología de Alemania, Australia, Brasil, Canadá, Dinamarca, Francia, Japón y Países Bajos. Fue puesto en público en junio del 2008 por la firma Global Science Online Gateway; en este momento era representado por 38 países. Es sencillo y flexible en su utilización por parte del usuario. A través de una consulta el usuario puede obtener de bases de datos y portales científicos, datos de referencia desconocidos e inaccesibles para los buscadores habituales, presentando estos resultados listados con indicaciones de sus fuentes. En la actualidad lo integran 56 países aportando recursos informativos a el portal. Posee una búsqueda avanzada acotada por un registro completo, título, autor y el rango de fecha; se encuentra para varios idiomas (Johnson, 2017).

Google Scholar: su función es similar al de la búsqueda web regular de Google, restringiéndose a los resultados de carácter científico. Entre los servicios que presenta está la búsqueda en diversas fuentes desde un mismo sitio, acceso a documentos a texto completo, búsqueda de la producción bibliográfica de un autor, de una revista o sobre un tema, conocer las citaciones de las publicaciones, entre otros. El rastreador del buscador toma registros de sitios donde la información es de libre acceso o de base datos comerciales. Ordena sus resultados por un determinado orden de relevancia, cuyos resultados ofrecen información del documento o recurso, artículos relacionados, diferentes versiones del documento. El buscador presenta como dificultades la presencia de resultados duplicados y la falta de transparencia en el contenido de la base de datos, pues no informa de qué fuentes indexa. Presenta una búsqueda avanzada mediante una interfaz que contiene como campos de búsqueda todas las palabras, frase exacta, al menos una de las palabras, autor, dónde fue publicado y por fecha (Google, 2005).

Springer Link: es una editorial europea que publica en el idioma de cada país sobre todo en Alemania y en Holanda para médicos y profesionales de los sectores de automoción, el transporte y la sanidad. La plataforma de búsqueda en línea de Springer permite acceder a los libros, revistas, obras de referencia, protocolos y bases de datos de ciencia, tecnología y medicina. Proporciona un acceso rápido y preciso a la investigación académica de Springer. Posee un diseño amplio y sencillo, lo que hace más fácil la navegación por sus distintas opciones. Proporciona una búsqueda avanzada mediante los campos de: todas las palabras, frase, con al menos una palabra, sin alguna palabra, título, autor/ editor y fecha de publicación (SpringerLink, 2017).

En el ámbito nacional teniendo en cuenta el bloqueo económico y comercial que impone a nuestro país el gobierno de los Estados Unidos, se han realizado varios esfuerzos por lograr la soberanía tecnológica y para se trabaja en la creación de un buscador propio. En el año 2007 fue lanzado el buscador 2x3 que solo presentaba los servicios de noticias, directorio y el tiempo. Posteriormente en el 2011 se crea el buscador Orion, en la Universidad de Ciencias Informáticas, con nuevas funcionalidades, pero no poseía la opción de búsqueda académica por lo cual no cubría las expectativas del sector académico. En la busca de crear un buscador más completo en el 2015 se lanza la plataforma c.u.b.a, un sistema de recuperación, basado en el motor de búsqueda de Orion, compuesto por tres componentes principales que son el subsistema de recolección, subsistema de indexación y la visualización. El componente de recolección rastrea la información que se encuentra en la red, en el caso de c.u.b.a se emplea Nutch por sus ventajas en el proceso de rastreo y almacenamiento de información. Como componente de indexación se emplea Solr que se encarga de recepcionar toda la información rastreada en la red y luego la procesa y almacena (Rodríguez, y otros, 2016). Cuenta con una búsqueda académica muy básica, por lo que no satisface la necesidad existente ya que la posibilidad de que los resultados esperados de un criterio de búsqueda introducido por el usuario no sean los mejores.

Con el estudio de las herramientas homólogas se analizó mediante qué parámetros es guiada esta búsqueda avanzada y cuáles son los elementos de filtrados más usados y esenciales para que el resultado sea más preciso. La Tabla 1 muestra una comparación entre los buscadores estudiados teniendo en cuenta los criterios que cada uno utiliza en la búsqueda avanzada.

Tabla 1 Comparación de sistemas homólogos

Buscadores Académicos	Autor	Titulo	Fecha de publicación	Frase	Una palabra	Sin la palabra	Resumen
Google Scholar	X	X	X	X	X	X	
SciELO	X	X	X				X
Springer Link	X	X	X	X	X	X	
Dialnet	X	X	X			X	X
World Wide Science	X	X	X				
Plataforma c.u.b.a							

Como resultado del estudio de las funcionalidades de los buscadores académicos profundizando en la búsqueda avanzada, se llegó a la conclusión de que poseen características comunes que propician el éxito entre los usuarios más exigentes. La tabla comparativa permitió definir los filtros de búsqueda a utilizar en el módulo propuesto, como es el caso del autor, título, fecha de publicación y frase.

La plataforma c.u.b.a no cuenta con un mecanismo de búsqueda avanzada para el subsistema académico que permita filtrar las búsquedas por diferentes parámetros. Por lo que se necesita la realización de un módulo que permita la búsqueda avanzada en la plataforma. El estudio de los buscadores académicos internacionales, arrojó que generalmente presentan una búsqueda avanzada, debido a que facilita la búsqueda de artículos y revistas de carácter académico. Sin embargo, mediante estos buscadores es poco probable obtener artículos y documentos publicados en la red cubana, ya que los buscadores internacionales generalmente dan mejor posicionamiento a revistas y documentos más conocidas en el ámbito internacional. Este posicionamiento es obtenido a través de algoritmos matemáticos que calculan el nivel de relevancia de los resultados, pero en algunas ocasiones las revistas pagan por un buen posicionamiento. Además, el código fuente de estos buscadores no puede ser utilizado debido a que son privativos lo que dificulta la integración con la plataforma c.u.b.a.

1.2 Tecnologías y herramientas

Se realiza un estudio y análisis de las herramientas y tecnologías existentes para el desarrollo del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a; con el objetivo de garantizar la correcta integración y compatibilidad con la plataforma.

1.2.1 Tecnologías del lado del Cliente

HTML (*Hiper Text Markup Language*) 5.0: es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Es un conjunto o serie de etiquetas incluidas en archivos de texto que definen la

estructura de un documento www y sus vínculos con otros documentos. Los navegadores www leen archivos de texto e interpretan sus etiquetas para determinar cómo desplegar la página web. Facilita la transformación de un archivo de texto inicial en una página web con diferentes tipos y tamaños de letras, con imágenes, animaciones, formularios. No requiere de un programa especial para crear la página lo cual permite que se creen desde cualquier ordenador o sistema operativo (Ferrer, y otros, 2017).

CSS (*Cascading Style Sheets*) 3.0: permite controlar el aspecto o presentación de los documentos electrónicos definidos con HTML. Separa el contenido y la presentación para crear páginas web complejas. Designa la función de cada elemento en la página: párrafo, titular, texto destacado, tabla, lista de elementos, entre otros. Utilizando CSS se establecen estilos con poco esfuerzo y sin ensuciar el código HTML. Es muy flexible y ofrece varias opciones para realizar la misma tarea. Permite definir varios estilos para diferentes medios o dispositivos: pantallas, impresoras, móviles y proyectores. Otras características que lo definen es la simplicidad, combinación de lenguajes alternativos, accesibilidad y mantenibilidad fácil (Perez, 2008).

JavaScript 1.8.5: es un lenguaje de programación interpretado simple, no hace falta tener criterio de programación, maneja objetos dentro de una página web y sobre el mismo puede definir diferentes eventos. Permite gestionar diferencias de implementación entre distintos navegadores. Es dinámico, pues responde a eventos en tiempo real, además está asociado a un valor no a la variable como tal. Fue desarrollado a partir del lenguaje Java el funciona para crear aplicaciones de todo tipo, pero JavaScript solo funciona dentro de una página HTML (Navarrete, 2006).

1.2.2 Tecnologías del lado del Servidor

PHP (*Hypertext Preprocessor*) 5.2.1: es un lenguaje de programación de propósito general de código abierto del lado del servidor diseñado para el desarrollo web de contenido dinámico. Es fácil de aprender y es invisible al navegador web y al cliente. Tiene una gran capacidad de conexión con los motores de base de datos como son MySQL y PostgreSQL. PHP es libre, por lo que es de fácil acceso para todos, permite aplicar técnicas de programación orientada a objetos. Su sintaxis es similar a la del C y es completamente expandible.

Tiene una gran velocidad de ejecución, utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables. Provee diferentes niveles de seguridad, además corre en cualquier plataforma utilizando el mismo código fuente. Presenta una gran variedad de módulos y es libre.

Debido a lo planteado anteriormente se decide usar este lenguaje del lado del servidor para el desarrollo de la interfaz web de la herramienta propuesta (Mariño, 2008).

1.2.3 Entorno de Desarrollo Integrado

NetBeans 8.2: es un entorno de desarrollo para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero sirve para cualquier otro lenguaje de programación. Es de código abierto y gratuito, el código fuente está disponible para su reutilización. Incluye un editor para varios lenguajes, editor de perfiles y un detector de errores. Proporciona aplicaciones de muestra en forma de plantillas de proyectos, las tecnologías que soporta, así como amplia variedad de plugin para todas las formas de desarrollo.

Posee varias versiones para los distintos sistemas operativos del mercado; su estructura modular le proporciona estabilidad y posibilidad de ser extendido gradualmente por desarrollos comunitarios, permitiendo agregar continuamente nuevas funcionalidades (Gimeno, y otros, 2011). Al ser gratuito, libre y potente se escogió para el desarrollo de la solución de la propuesta de solución

1.2.4 Sistema de Gestión de Base de Datos

MongoDB 3.4: posee un código abierto y está escrito en C++, orientado al almacenamiento de datos en documentos al estilo JSON con esquemas dinámicos. Conserva los índices de todos los atributos y es más flexible para la agregación y procesamiento de datos. Sus colecciones son análogas a las tablas en una base de dato relacional, cada colección contiene varios documentos que pueden ser grandes.

Posee un diseño conceptual del esquema de lo que los campos de los documentos de una colección representarán y como estarán estructurados. No requiere que en una colección todos los documentos tengan la misma estructura. Con este sistema se logra un mejor manejo de la información frente a un volumen considerable de datos (10gen, 2011).

1.2.5 Servidor web

Apache 2.2: es un servidor HTTP de código abierto, que implementa el protocolo HTTP/1.12. Su objetivo es proporcionar un servidor seguro, eficiente y extensible que proporcione servicios HTTP en sincronía con los estándares HTTP actuales. Es altamente configurable, posee una base de datos de autenticación y negociado de contenido, es multiplataforma. Puede ser adaptado a diferentes entornos y necesidades, con sus diferentes módulos de apoyo y con la API (*Application Programming Interface*) de programación de

módulos, para el desarrollo de módulos específicos. Es extensible debido a que es modular; se desarrollaron diversas extensiones entre las que se destaca PHP, un lenguaje del lado del servidor (Kabir, 2002).

1.2.6 Mecanismo para la recolección de información

Nutch 1.9: es un programa desarrollado en el lenguaje Java, su arquitectura es muy flexible, permitiendo realizarle mejoras por parte de los usuarios a través de plugins. Es independiente del servidor de indexación y permite la recopilación de muchos tipos de documentos. Los parámetros utilizados por Nutch para el rastreo y la indexación pueden ser configurables mediante ficheros escritos en XML. Mantiene internamente un ranking que define el orden de rastreo de las urls. Como mecanismo de rastreo posee *parsers*, que se encargan de descomponer las páginas web y analizar cada uno de los recursos que la componen o tienen relación. Se escoge Nutch como mecanismo de recolección de información porque facilita la incorporación de nuevos plugins y es el mecanismo de rastreo utilizado en la plataforma c.u.b.a lo que facilitará la integración entre ambos sistemas. (Nieto, 2009)

1.2.7 Mecanismo para la indexación de la información

Solr 6.3.0: es una plataforma de búsqueda de código abierto, permite la búsqueda de texto completo, resaltado de resultados y manejo de documentos. Permite realizar búsquedas distribuidas y replicación de índices. Fácil de configurar y usar. Una de sus principales características es su API estilo REST, además permite importar datos desde una base de datos. (Foundation-Tika, 2014) Es un mecanismo de indexación que utiliza un modelo de recuperación de información híbrido, se basa en un modelo booleano y un modelo vectorial para establecer un subconjunto de documentos relevantes. Es capaz de manejar complejos criterios de búsqueda, corrige la ortografía en las consultas, permite realizar resultados, configurar la relevancia de términos, entre otras funcionalidades. Debido a esto se empleará como mecanismo de indexación, también es el utilizado en la plataforma c.u.b.a y es el servidor índice empleado por Nutch. (Seta, 2010)

1.2.8 Herramientas de prueba

JMeter 2.12: es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el desempeño de una variedad de servicios, con énfasis en aplicaciones web. JMeter puede ser usado como una herramienta de pruebas unitarias para conexiones de bases de datos con JDBC, FTP, LDAP, Servicios web, JMS, HTTP y conexiones TCP genéricas. (Foundation-Jmeter, 2015)

Acunetix 9.5: comprueba los sistemas en busca de múltiples vulnerabilidades que un atacante podría aprovechar para obtener acceso a los sistemas y datos. Acunetix puede utilizarse para realizar escaneos de vulnerabilidades en aplicaciones web y para introducir pruebas de acceso frente a los problemas identificados. La herramienta provee sugerencias para mitigar las vulnerabilidades identificadas y puede utilizarse para incrementar la seguridad de servidores web o de las aplicaciones que se analizan. (Acunetix, 2015)

1.2.9 Lenguaje de modelado

Lenguaje Unificado de Modelado 2.1: es el lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema creando una idea inicial de como quedara estructurado el proyecto. Sirve para el modelado del negocio y sistema de software. Cuenta con un conjunto de notaciones y diagramas para modelar sistemas orientados a objetos y puede utilizarse en las diferentes etapas del ciclo de vida de del desarrollo de sistemas y es independiente del proceso o metodología de desarrollo y del lenguaje de implementación (Gonzalez, y otros, 2012).

1.2.10 Herramienta CASE

Visual Paradigm 8.0: es una herramienta CASE multiplataforma, que soporta el ciclo de desarrollo de software: análisis, diseño, implementación y pruebas. Utiliza UML como lenguaje de modelado, permite realizar ingeniería directa e inversa en diferentes lenguajes. En él se puede representar todos los tipos de diagramas UML para las fases que propone RUP, la captura de requisitos, análisis, diseño e implementación.

Permite la interoperabilidad con otras aplicaciones, tiene distintas versiones como son: Enterprise, Professional, Standard, Modeler, Personal y Community. Facilita licencias especiales para fines académicos. Debido a lo abordado se escoge Visual Paradigm para realizar el proceso de modelado (Paradigm, 2017).

1.3 Metodología de desarrollo

AUP-UCI: es una adaptación de la metodología de desarrollo AUP (Proceso Unificado Ágil). Fue creada en la UCI con el objetivo de que se ajustara al ciclo de vida de los proyectos de la universidad. Emplea técnicas ágiles en el desarrollo de aplicaciones de software de negocio. Define que el ciclo de vida de un proyecto consta de tres fases: la inicial se encarga de realizar un estudio de la organización del cliente, estimaciones

de tiempo, esfuerzo, costo y decidir si se ejecuta o no el proyecto. La segunda fase ejecución se realizan las actividades requeridas para el desarrollo del software, los ajustes de los planes del proyecto considerando los requisitos y la arquitectura, implementar y liberar el producto. Como fase final se realiza el cierre donde se analizan los resultados del proyecto y las actividades formales del cierre del proyecto. Se escoge como metodología ya que con la adaptación realizada se logra estandarizar el proceso de desarrollo de software en la UCI, cumpliendo con las buenas practicas que define CMMI-DEV v1.3. (Sánchez, 2015)

1.4 Marco de trabajo

Symfony 2.8: diseñado para optimizar el desarrollo de las aplicaciones web. Separa la lógica de negocio, la del servidor y la presentación de la aplicación web. Proporciona herramientas y clases para reducir el tiempo de desarrollo de una aplicación web compleja. Automatiza tareas comunes, lo cual permite al desarrollador enfocarse en los aspectos específicos de la aplicación. Es compatible con la mayoría de los gestores de bases de datos.

Fue creado para aprovechar las características de PHP; es uno de los marcos de trabajo PHP con mejor rendimiento. Su arquitectura interna esta desacoplada, lo que permite reemplazar o eliminar las partes que no se ajustan a un proyecto determinado. Es el marco de trabajo que más idea incorpora de sus semejantes, incluyendo aquellos que no están programados con PHP (Perez, 2013).

Doctrine: es un mapeador de objetos relacional programado en PHP, multiplataforma. Puede generar clases a partir de base de datos creadas. No necesita generar esquemas complejos en XML como otros *frameworks*. Es una comunidad grande que integra marcos diferentes como son: Symfony, Zend Framework, Flow y otros.

Proporciona facilidad de búsqueda pues permite realizarlas basada en cualquier campo de una tabla. Posee su propio lenguaje Doctrine Query Language siendo una de sus características más importante, le permite manejar las interacciones con la base de datos. Está diseñado para extraer objetos, entiende las relaciones por lo que evita que algunas operaciones se hagan a mano y es portable con diferentes bases de datos (Garcia, y otros, 2016).

1.5 Conclusiones del capítulo

En este capítulo se profundizó en los elementos teóricos de la investigación que sustentan la propuesta de solución del problema que se plantea, se pudo llegar a las siguientes conclusiones:

- El estudio y análisis de buscadores académicos más utilizados del mundo que proporcionan la funcionalidad de búsqueda avanzada, permitió identificar requerimientos necesarios para tener en cuenta en la propuesta de solución.
- El estudio de los buscadores nacionales, arrojó que no existe actualmente un subsistema académico que presente una búsqueda avanzada y en el ámbito internacional no se ajustan a la necesidad de la plataforma c.u.b.a.
- La selección de la metodología, tecnologías y herramientas basadas en software libre, permitió obtener una base tecnológica que permitirá el desarrollo de la solución.

Capítulo 2. Módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a

En este capítulo se abordan los aspectos fundamentales relacionados con el diseño del módulo a desarrollar. Para la representación de las clases conceptuales del problema a resolver se utiliza el diagrama de modelo del dominio. Se generan los artefactos relacionados a la especificación de requerimientos funcionales y no funcionales vinculados al software y la especificación de los casos de uso del sistema. Se muestran los principales artefactos de ingeniería de software adecuados a los casos de uso más críticos. Como parte del diseño se definen los estilos y patrones de arquitectura.

2.1 Modelo del Dominio

Un modelo de dominio es una representación de las clases conceptuales del mundo real, no de componentes de software (Craig, 2003). Permite modelar clases conceptuales significativas en un determinado problema.

2.1.1 Descripción La modelación del dominio es una de las herramientas más importantes para garantizar la comprensión y descripción de las clases, conceptos y sus relaciones. En la Tabla 2, se muestran conceptos relacionados con la propuesta de solución:

Tabla 2 Clases del modelo de dominio

Conceptos	Descripción
Usuario	Persona que introduce los criterios de búsqueda en la aplicación web.
Indexador	Componente del motor de búsqueda que almacena los resultados del rastreo en forma de índice.
Rastreador	Componente del motor de búsqueda que accede a la información publicada en la red y procesa los documentos encontrados.
Motor de búsqueda	Sistema de recuperación de información compuesto por una aplicación web, indexador y rastreador.
Interfaz web	Vista mediante la cual se muestra el resultado de la búsqueda.

Intranet	Red informática donde están los documentos.
Documento	Tipo de contenido electrónico publicado en la web.

2.1.2 Diagrama de clases del modelo del dominio

El diagrama representa los objetos relacionados con los principales conceptos que se manejan en el desarrollo del módulo de búsqueda avanzada del subsistema académico para la plataforma c.u.b.a. Esta representación permite una mejor comprensión de los conceptos que se tratan. En los diagramas del modelo de dominio se emplea un vocabulario común para comprender el contexto en que se encuentra el sistema. (Pressman, 2010)

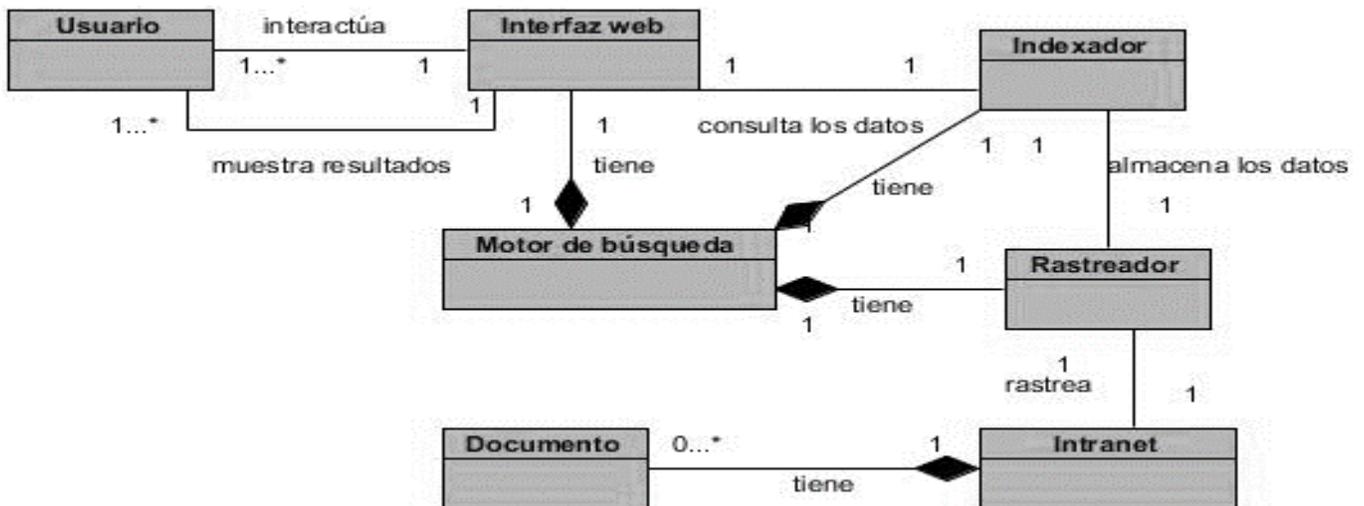


Figura 1 Diagrama de clases del modelo de dominio

La Figura 1 muestra la relación entre los conceptos que intervienen en la investigación que se presenta. Se puede observar que en la Intranet se encuentran los documentos que son recopilados y procesados por el rastreador y seguidamente almacenados en el indexador. Como muestra el diagrama el motor de búsqueda está compuesto por la interfaz, el rastreador y el indexador. La interfaz consulta en el indexador los datos que son introducidos por el usuario a través de un campo de texto, obteniendo resultados acordes a la petición del cliente que son mostrados en la interfaz web.

2.2 Especificación de los Requisitos del Software

Los requisitos del software expresan las necesidades objetivas que presenta el cliente, permiten establecer las funciones, características, restricciones en el funcionamiento y los procesos de desarrollo del software. (Jesús, 2015) Los requisitos identificados para la búsqueda avanzada del subsistema académico de la plataforma cubano c.u.b.a se relacionan a continuación.

2.2.1 Requisitos funcionales

Tabla 3 Requisitos funcionales del sistema

Nº	Nombre	Descripción	Prioridad	Complejidad
RF1	Realizar búsqueda académica avanzada por autor.	El sistema debe permitir filtrar la búsqueda por el nombre del autor.	Alta.	Media.
RF2	Realizar búsqueda académica avanzada por fecha de publicación.	El sistema debe permitir filtrar la búsqueda por la fecha de publicación.	Alta.	Alta.
RF3	Realizar búsqueda académica avanzada por título.	El sistema debe permitir filtrar la búsqueda por el título.	Alta.	Media.
RF4	Realizar búsqueda académica avanzada por frase.	El sistema debe permitir filtrar la búsqueda de una frase exacta	Media.	Media.
RF5	Realizar búsqueda académica avanzada por una palabra.	El sistema debe permitir filtrar la búsqueda por una palabra.	Media.	Alta.
RF6	Realizar búsqueda académica avanzada por todas las palabras.	El sistema debe permitir filtrar la búsqueda por todas las palabras.	Media.	Alta.
RF7	Realizar búsqueda académica avanzada sin las palabras.	El sistema debe permitir filtrar la búsqueda en artículos sin la palabra.	Media.	Alta.
RF8	Realizar búsqueda académica avanzada por formato.	El sistema debe permitir filtrar la búsqueda por un formato específico.	Media.	Baja.
RF9	Realizar búsqueda académica avanzada por revista.	El sistema debe permitir filtrar la búsqueda por la revista a la que pertenece.	Alta.	Media.

RF10	Realizar búsqueda académica avanzada por idioma.	El sistema debe permitir filtrar la búsqueda por el idioma.	Media.	Baja.
RF11	Mostrar resultados de la búsqueda académica avanzada.	El sistema debe mostrar los resultados de la búsqueda avanzada.	Alta.	Media.

2.2.2 Requisitos no funcionales

Requerimientos de software

RNF 1: Se requiere la instalación del servidor web Apache y PHP para poder visualizar la interfaz web.

RNF 2: Se requiere la instalación de la Máquina Virtual Java para el correcto funcionamiento del rastreador.

RNF 3: Los dispositivos clientes deben contar con navegadores web que soporten HTML 5.0 y CSS 3.0.

Requerimientos de hardware

RNF 4: Para servidor de rastreo se necesita 4 GB RAM, CPU de 4 núcleos y como mínimo 60 GB de Disco Duro.

RNF 5: Se necesita para el servidor de índice: 4 GB RAM, CPU de 4 núcleos y 1 GB de almacenamiento.

Requerimientos de diseño e implementación

RNF 6: Utilizar como lenguaje de programación para la interfaz web PHP en su versión 5.5 o mayor.

RNF 7: Para el *plugin* de componente de rastreo se deberá utilizar Java en su versión 1.8 como lenguaje de programación.

RNF 8: Se deberá utilizar para el desarrollo de la aplicación web Symfony en su versión 2.8 como marco de trabajo.

RNF 9: Para el componente de rastreo se deberá utilizar la herramienta Nutch 1.9 en su versión o superior.

RNF 10: Para el componente de indexación se deberá utilizar la herramienta Solr en su versión 6.3 o superior.

Requerimientos de interfaz de usuario

RNF 11: En el formulario de búsqueda avanzada se brindará una descripción textual para cada campo.

RNF 12: La interfaz deberá ser de fácil acceso y manejo para los usuarios.

RNF 13: En la página de resultados siempre estará visible el formulario de búsqueda avanzada y el paginador, para resoluciones superiores a 800x600.

Requerimientos de seguridad

RNF 14: Los campos de entrada deben ser válidos.

RNF 15: El formulario de búsqueda avanzada debe estar protegido por los mecanismos de control de acceso que establece la plataforma c.u.b.a.

RNF 16: Integridad: el sistema debe permitir la realización de salvadas periódicas de la información.

Requerimientos de usabilidad

RNF 17: El módulo de búsqueda avanzada debe poder ser transferido de un ambiente a otro o permitir ser reemplazado por nuevas versiones.

RNF 18: Se requiere el uso de herramientas y recursos de software libre, las cuales se podrán usar, modificar y distribuir libremente.

Requerimientos de eficiencia

RNF 19: Debe permitir que al menos 1000 usuarios interactúen de manera concurrente.

RNF 20: Debe poder responder alrededor de 4000 peticiones en 5 segundos como máximo.

2.3 Propuesta del sistema

El sistema debe permitir a los usuarios realizar la búsqueda avanzada del subsistema académico, mediante filtros que le permitirán realizar una búsqueda más acertada y eficiente, atendiendo a los siguientes criterios:

Autor: Permitirá obtener como resultado los artículos pertenecientes a un autor en específico.

Título: Este servicio permitirá realizar la búsqueda de un artículo por el título que tenga.

Fecha de publicación: Se obtendrán los resultados que pertenezcan a un año en específico o a partir de ese año.

Frase: El sistema devolverá los artículos donde se encuentre la frase exacta.

Por una palabra: Este servicio permitirá obtener como resultado los artículos que incluyan la palabra en alguna parte.

Por todas las palabras: El sistema analizará cada palabra y devolverá los resultados en correspondencia con las mismas.

Sin las palabras: Permitirá obtener los artículos en los que no estén incluidas las palabras que el usuario introduzca.

Formato: El sistema devolverá los artículos con el formato deseado.

Revista: Se obtendrá los artículos pertenecientes a una revista en específico.

Idioma: La búsqueda se podrá realizar en varios idiomas, dependiendo del que el usuario escoja.

El módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a que se propone, pretende responder a las peticiones introducidas por los usuarios en forma de consultas, a través de campos de textos llamados filtros, como los mencionados anteriormente que permitan acotar la búsqueda solicitada por los usuarios de los artículos académicos publicados en la red cubana. Por cada petición debe generar un mensaje de respuesta que contenga la información relevante y pertinente. El término relevante se define como información significativa o importante con respecto a la necesidad de información planteada por el usuario. En el caso de pertinencia se refiere a calidad y a documentos que añaden nueva información a la investigación realizada por el usuario y que resulte útil en el trabajo que ha propiciado la pregunta. El módulo permite mejorar el proceso de recuperación de información carácter académico en la plataforma c.u.b.a.

2.4 Modelo de Casos de uso del sistema

Un caso de uso representa una unidad discreta de interacción entre un usuario y el sistema. El modelo de casos de uso describe las funcionalidades propuestas del sistema a desarrollar. Permite el acuerdo entre desarrolladores y clientes, y proporciona la entrada fundamental para el análisis, el diseño y las pruebas. (Pressman, 2010)

2.4.1 Diagrama de Casos de uso del sistema

A continuación, se encuentra el diagrama de caso de uso, correspondiente a el actor, para un mejor entendimiento.

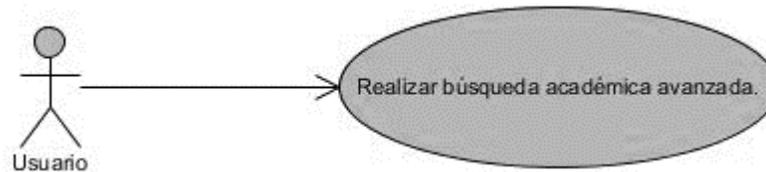


Figura 2 Caso de uso inicializado por el usuario

En el diagrama se encuentra representado 1 caso de uso y el actor que interactúa con él. A continuación, se presenta el caso de uso y una breve descripción.

- **CU1 Realizar búsqueda académica avanzada:** Engloba todas las maneras de realizar una búsqueda avanzada por parte del usuario.

2.4.2 Especificación de casos de uso

A continuación, se presenta la descripción del caso de uso CU Realizar búsqueda avanzada. El objetivo fundamental de la especificación de los casos de uso es un mejor entendimiento de las funcionalidades que abarcan.

Tabla 4 Descripción del CU 2. Realizar búsqueda académica avanzada

Objetivo	Realizar la búsqueda académica avanzada.
Actores	Usuario.
Resumen	El usuario introduce los parámetros que quiere consultar en los campos de texto del formulario de la búsqueda avanzada.
Complejidad	Alta.
Prioridad	Alta.
Precondiciones	Debe existir un índice con los documentos académicos buscados en la red que contenga los metadatos a seleccionar.
Postcondiciones	Se ha introducido la consulta al formulario y se devolvieron resultados correspondientes al autor.
Flujo de eventos	
Flujo básico <Búsqueda avanzada>	
Actor	Sistema

	Presiona el botón "Búsqueda avanzada"	
		Muestra un formulario con diferentes criterios para filtrar la búsqueda: -autor -titulo -revista -formato -idioma -fecha -palabras -sin palabras -algunas palabras -frase
	Introduce los criterios de búsqueda	
		Muestra los resultados filtrados por los criterios.
		Termina el CU.
Flujos alternos		
Introducir datos incorrectos en formulario de búsqueda avanzada		
	Actor	Sistema
	Presiona el botón "Búsqueda avanzada"	
		Muestra un formulario con diferentes criterios para filtrar la búsqueda.
	Introduce los datos incorrectos	
		Muestra un mensaje indicado que no existen resultados para el criterio de búsqueda introducido.
Flujos alternos		
Dejar los campos vacíos en formulario de búsqueda avanzada		
	Actor	Sistema
	Presiona el botón "Búsqueda avanzada"	
		Muestra un formulario con diferentes criterios para filtrar la búsqueda.
	No introduce ningún criterio de búsqueda avanzada	
		Muestra un mensaje indicado que no existen resultados porque existen campos vacíos.
Relaciones	CU	

	Incluidos	
	CU extendidos	
Requisitos funcionales	no	
Asuntos pendientes		

2.5 Arquitectura del módulo de búsqueda avanzada

La arquitectura es un elemento primordial en el éxito o el fracaso de un proyecto, proporciona una visión global del módulo a desarrollar, es una vista estructural de alto nivel que define el estilo arquitectónico o su combinación para la solución de un determinado problema.

El módulo de búsqueda avanzada a desarrollar debe tener 2 componentes. Un primer componente sería la Aplicación web que contiene la interfaz donde el usuario interactúa con el módulo y un segundo componente el mecanismo de Indexación. Solr es el indexador, y contiene toda la información de los documentos almacenados en la base de datos a través del *Core* academia. Entre la aplicación web y el indexador existe un flujo de información realizado mediante el protocolo HTTPS.

A continuación, se muestra en la Figura 3 la arquitectura descrita anteriormente:



Figura 3 Arquitectura del módulo propuesto

2.6 Estilo arquitectónico

Los estilos arquitectónicos expresan un esquema de organización estructural para sistema de software, proveen un conjunto de tipos de elementos predefinidos, especifica sus responsabilidades e incluye reglas y guías para organizar las relaciones entre ellos (Rodríguez, y otros, 2016). El estilo arquitectónico que presenta Symfony es Modelo-Vista-Controlador conocido por sus siglas(MVC) separa la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones web. Facilita la mantenibilidad y escalabilidad

del sistema, de forma simple y sencilla, a la vez que permite “no mezclar lenguajes de programación en el mismo código”. (Rodríguez, y otros, 2016) Permite dividir las aplicaciones en tres capas:

- **Vista:** se refiere al diseño, visualización de la información, estilos, estructura visual y colores. Es la capa con la que interactúan los usuarios. Esta capa se evidencia a través de la biblioteca jQuery, en conjunto con los archivos CSS, HTML y JavaScript. Esta capa está presente en el módulo en las clases **base.html.twig**, **inicio.html.twig** y **result.html.twig**; la clase inicio y *result* contienen las vistas que muestran el formulario y los resultados y ambas heredan de la clase base
- **Modelo:** responsable de la conexión a la base de datos y la manipulación de los datos. Se divide en dos subcapas, la de negocio y la de acceso a datos. La primera contiene clases gestoras encargadas del manejo de la lógica del negocio. En el caso de la subcapa de acceso a datos es la encargada de establecer la conexión con la base de datos y se encuentra presente en la clase **SearchService.php** donde se encuentra implementado el método que consulta la Query de solr.
- **Controlador:** se procesa y muestra los datos obtenidos por el modelado. Gestiona la interacción entre el usuario y los datos. Pertenece a esta capa el Controlador Frontal, mediante los archivos app.php y app_dev.php empleados para entornos de producción y entornos de desarrollo respectivamente. Los archivos de enrutamiento también forman parte de esta capa y las plantillas Twig, responsables de transformar cada archivo Twig en HTML. Y como controlador del flujo de información que se recibe y envía forma parte la clase Controller. En esta capa se encuentran las clases **DefaultController.php** y **ServiceQuery.php** donde se encuentran implementado los servicios.

La Figura 4 muestra el funcionamiento de la arquitectura incluyendo el estilo arquitectónico presente en el componente: Aplicación web para una mejor comprensión, ajustada al módulo a desarrollar:

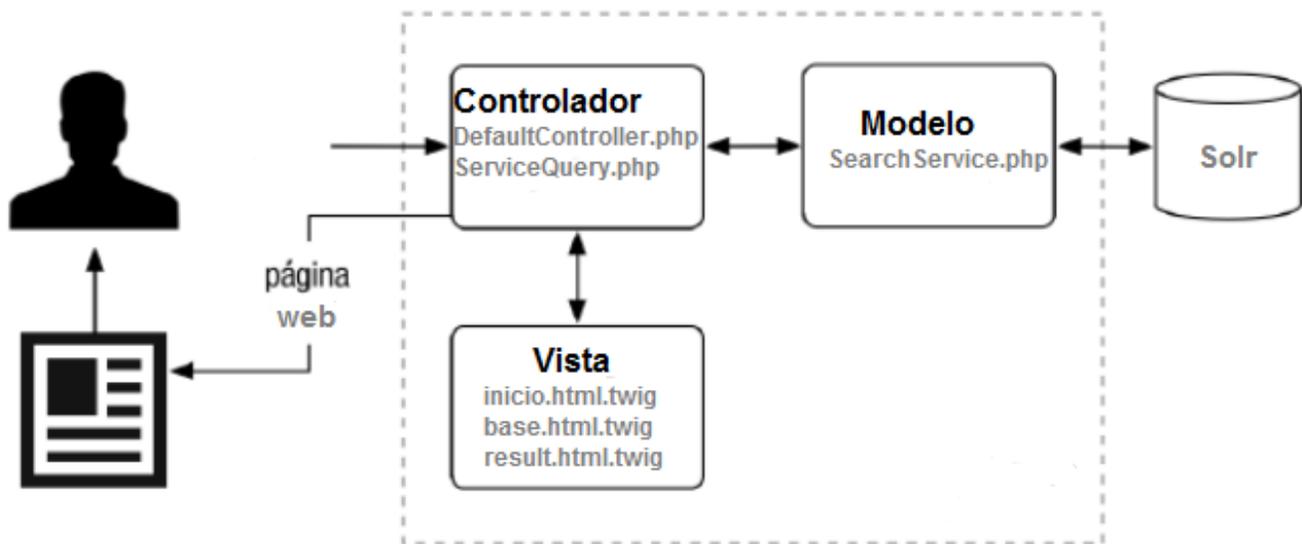


Figura 4 Arquitectura Modelo Vista Controlador

En la figura anterior el cliente envía una petición, que es interceptada por el controlador encargado de realizar las validaciones, procesamiento de datos y lógica de negocio asociadas a la petición. El controlador envía datos al modelado para realizar la consulta al Solr y obtiene los resultados para finalmente enviarlos a la vista para ser mostrado nuevamente al cliente a través de una respuesta.

2.7 Patrones de diseño

Un patrón es una solución a un problema en un contexto particular, permite entender cómo adaptar una solución a la variante particular del problema donde se quiere aplicar. Los patrones facilitan la reutilización de diseños y arquitecturas de software que han tenido éxito. Facilitan la reestructuración de un sistema tanto si fue concebido o no con patrones en mente. (Pressman, 2010)

En el diseño del módulo de búsqueda avanzada para el subsistema académico se emplearon algunos patrones GRASP (Patrones Generales de Software para Asignación de Responsabilidades), que describen principios fundamentales de la asignación de responsabilidades a objetos: (Hamon, 2014)

Experto: Cuando se trabaja con Symfony es uno de los patrones que más se utilizan, con la inclusión de la librería Doctrine que permite mapear la base de datos. Symfony utiliza esta librería para generar clases con todas las funcionalidades comunes de las entidades, encapsular toda la lógica de los datos y realizar su

capa de abstracción en el modelo. Estas clases de abstracción de datos poseen un grupo de funcionalidades que están relacionadas con la entidad que representan y contiene la información necesaria de la tabla. Este patrón se evidencia en la clase **SearchService.php** donde se hace las consultas al Solr.

Controlador: El patrón controlador es el intermediario entre la interfaz y el algoritmo que la implementa, recibe los datos del usuario y los envía a las clases según el método llamado. La presencia del patrón se evidencia en la clase **DefaultController.php** donde se delega las funcionalidades. Se encarga de atender y ofrecer respuesta a cada una de las peticiones realizadas por el usuario mediante la interfaz web.

Bajo Acoplamiento: Es una medida de la fuerza en las conexiones de una clase con otras, la recurrencia y las conexiones a ella. Una clase con bajo acoplamiento no depende de muchas clases. Su utilización se evidencia en que las clases controladoras del sistema no se relacionan entre sí, lo que disminuye las dependencias entre las mismas.

Se emplearon algunos patrones GOF, que describen las formas comunes en que diferentes tipos de objetos, pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases y la formación de estructuras de mayor complejidad. Además, permiten crear grupos de objetos para ayudar a realizar tareas complejas. Se emplearon los siguientes (Craig, 2003):

Estructurales:

Decorator (Decorador): Permite añadir responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades. En el módulo se observa este patrón en la vista **base.html.twig** de ella heredan el resto de las páginas de la aplicación. Esta vista contiene los elementos que son comunes para el diseño y estructura del módulo.

Otros:

Registry: Este patrón es útil para los desarrolladores en la Programación Orientada a Objetos. Es un medio sencillo y eficiente de compartir datos y objetos en la aplicación, sin la necesidad de preocuparse por conservar numerosos parámetros o utilizar variables globales. Se aplica en la clase de configuración (*config.yml*) que es la encargada de guardar las variables globales del módulo, como *drivers* de conexión a la base de datos y rutas bases del proyecto.

2.8 Modelo de diseño

El modelo de diseño se encarga de describir la realización de los casos de uso del sistema, y se utiliza como medio de abstracción del modelo de implementación y el código fuente del software. Transmite una comprensión en profundidad de los aspectos relacionados con los requerimientos no funcionales y las restricciones concernientes a los lenguajes de programación (Sommerville, 2011).

2.8.1 Diagrama de clases del diseño con estereotipos web

A continuación, se muestra el diagrama de clases del diseño con estereotipos web, pertenecientes al caso de uso CU Realizar búsqueda académica avanzada:

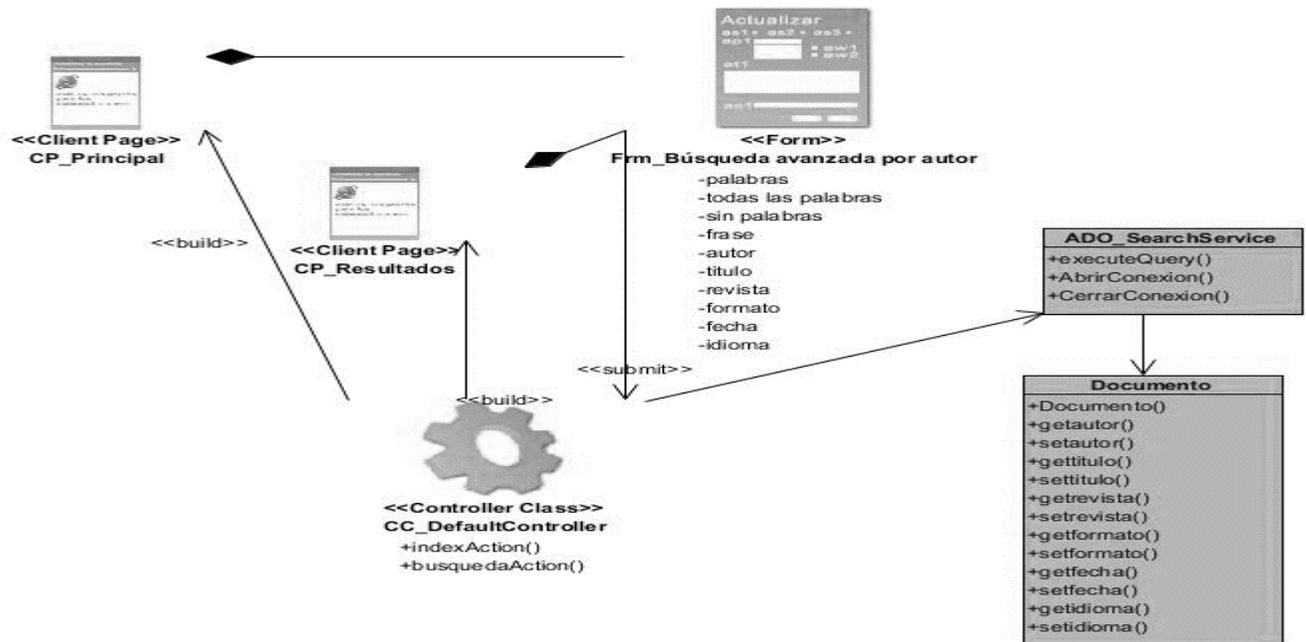


Figura 5 Diagrama de clases de diseño con estereotipos web del caso de uso "Realizar búsqueda académica avanzada".

Descripción del diagrama de clase de diseño del CU Realizar búsqueda académica avanzada:

Se presenta una página principal que tiene un enlace con el formulario para introducir los criterios de la búsqueda avanzada, el formulario envía los datos a la clase DefaultController. La clase controladora

construye los resultados a través del método `busquedaAction` que son mostrados en la página principal. Estos resultados son consultados en la base de datos a través del método `executeQuery`.

2.9 Diagrama de interacción

Los diagramas de interacción son empleados para modelar los comportamientos dinámicos que caracterizan un sistema informático. Representan un conjunto de objetos, clases, sus relaciones y los mensajes que se pueden enviar entre ellos.

2.9.1 Diagramas de secuencia

Como ejemplo de los diagramas de secuencia se presenta el perteneciente a el CU Realizar búsqueda académica avanzada.

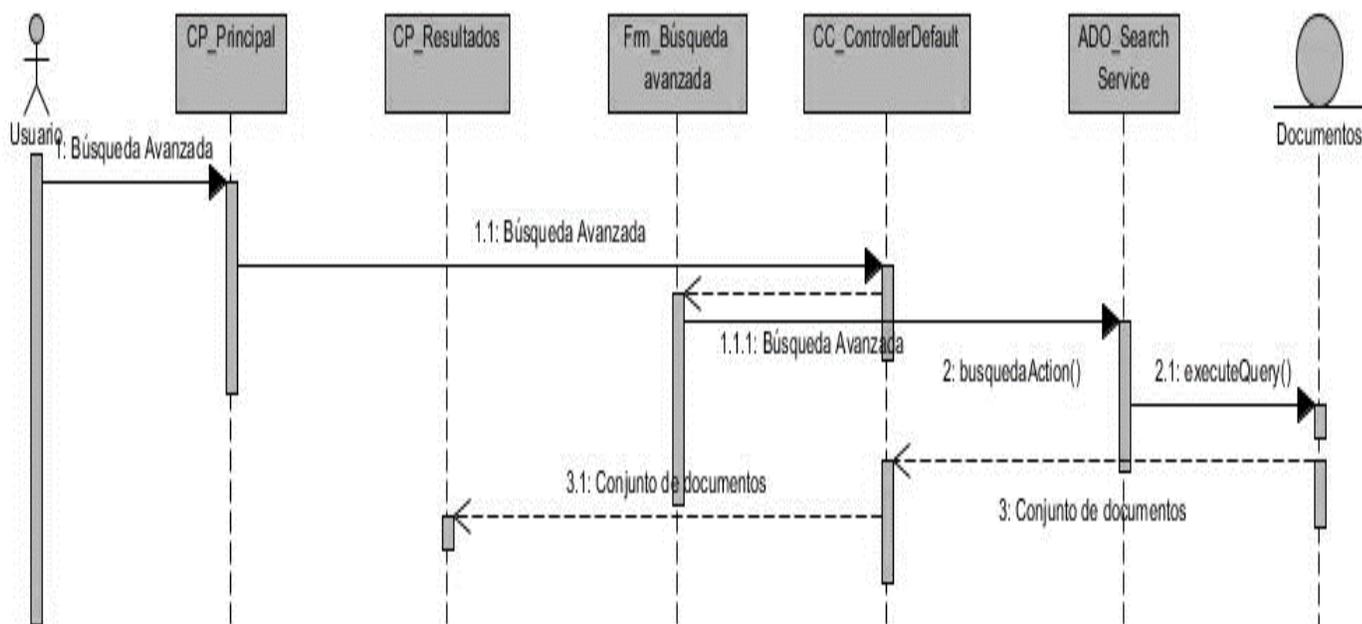


Figura 6 Diagrama de secuencia para el caso de uso "Realizar búsqueda académica avanzada"

2.10 Diseño de la Base de Datos

El diseño de Base de Datos se basa en la identificación de los objetos primarios que el módulo va a procesar, la composición y los atributos. Se encuentran almacenados los objetos, la relación entre ellos y los procesos que los transforman. (Blázquez, 2014)

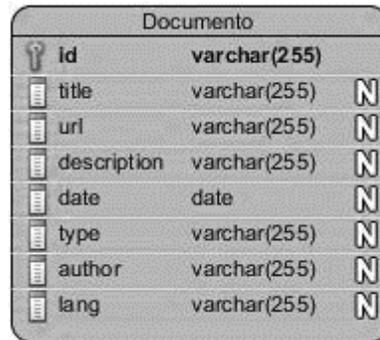


Figura 7 Diagrama de diseño de la base de datos

El modelo de datos representado anteriormente es un documento en Solr y sus atributos, que son metadatos que le fueron extraídos a un documento en el proceso de rastreo como es el caso del id, título, la url, fecha, descripción, tipo, autor, idioma.

2.11 Diagrama de Despliegue

El modelo de despliegue es un tipo de diagrama de UML que se utiliza para modelar la disposición física de los artefactos de software en nodos. Modela la arquitectura en tiempo de ejecución del sistema.

En la figura 8 se muestra la distribución física de los componentes del sistema propuesto para su despliegue. Se considera mejor que cada uno de ellos se encuentren en servidores independientes con el objetivo de utilizar al máximo las características de software y hardware.

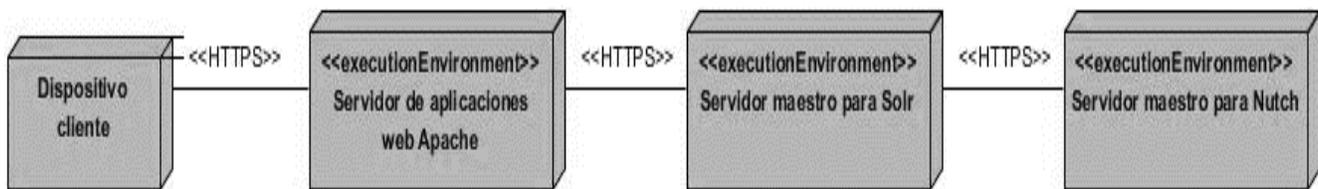


Figura 8 Diagrama de despliegue

Representa como primer nodo el Dispositivo-Cliente que es el utilizado por el usuario para realizar la búsqueda avanzada, a través del protocolo HTTPS por el puerto 443. En el caso del Servidor de aplicaciones web Apache se atienden y ofrecen las respuestas a cada una de las solicitudes del cliente. Se observan en los otros dos nodos el Servidor maestro para Solr y el Servidor maestro para Nutch; cada uno de ellos realiza el intercambio de mensaje a través del protocolo HTTPS a través del puerto 8983.

2.12 Conclusiones del capítulo

En este capítulo se han abordado los elementos del análisis y diseño de la búsqueda avanzada del subsistema académico para la plataforma c.u.b.a, arribando a las siguientes conclusiones:

- La elaboración del modelo de dominio y su descripción permitió una mayor comprensión del flujo de trabajo del módulo que se propone.
- La identificación de los requisitos funcionales, permitió agrupar las funcionalidades del sistema en caso de uso, que contribuyeron a una mayor comprensión de los principales procesos de la búsqueda avanzada.
- La identificación de los requisitos no funcionales posibilitó definir las condiciones y características del sistema a desarrollar.
- La elaboración de los diagramas de clase del diseño y los diagramas de colaboración propició la comprensión de la distribución y asignación de responsabilidades de cada una de las clases involucradas en los casos de uso analizados.

Capítulo 3. Implementación y validación del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.

La fase de implementación del sistema comprende la materialización, en forma de código, de los artefactos, descripciones y arquitectura propuestos en la etapa de análisis y diseño; con el objetivo de conformar el sistema final requerido por el cliente. El módulo que se construye debe ser sometido a pruebas que corroboren la correspondencia entre el producto y los requisitos funcionales definidos anteriormente. Esta es la etapa de validación donde pueden realizarse los diferentes tipos de pruebas en función de los objetivos.

3.1 Modelo de componentes

3.1.1 Diagrama de componentes

El diagrama de componentes es un gráfico que muestra los componentes de un sistema de software conectados por las relaciones de dependencias lógicas entre cada uno de ellos. Provee una vista arquitectónica de alto nivel del sistema, ayudando a los desarrolladores a visualizar el camino de la implementación. La realización del diagrama posibilita tomar decisiones respecto a las tareas de implementación y los requisitos (Cruz, y otros, 2012). A continuación, se muestra el diagrama de componentes del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a:

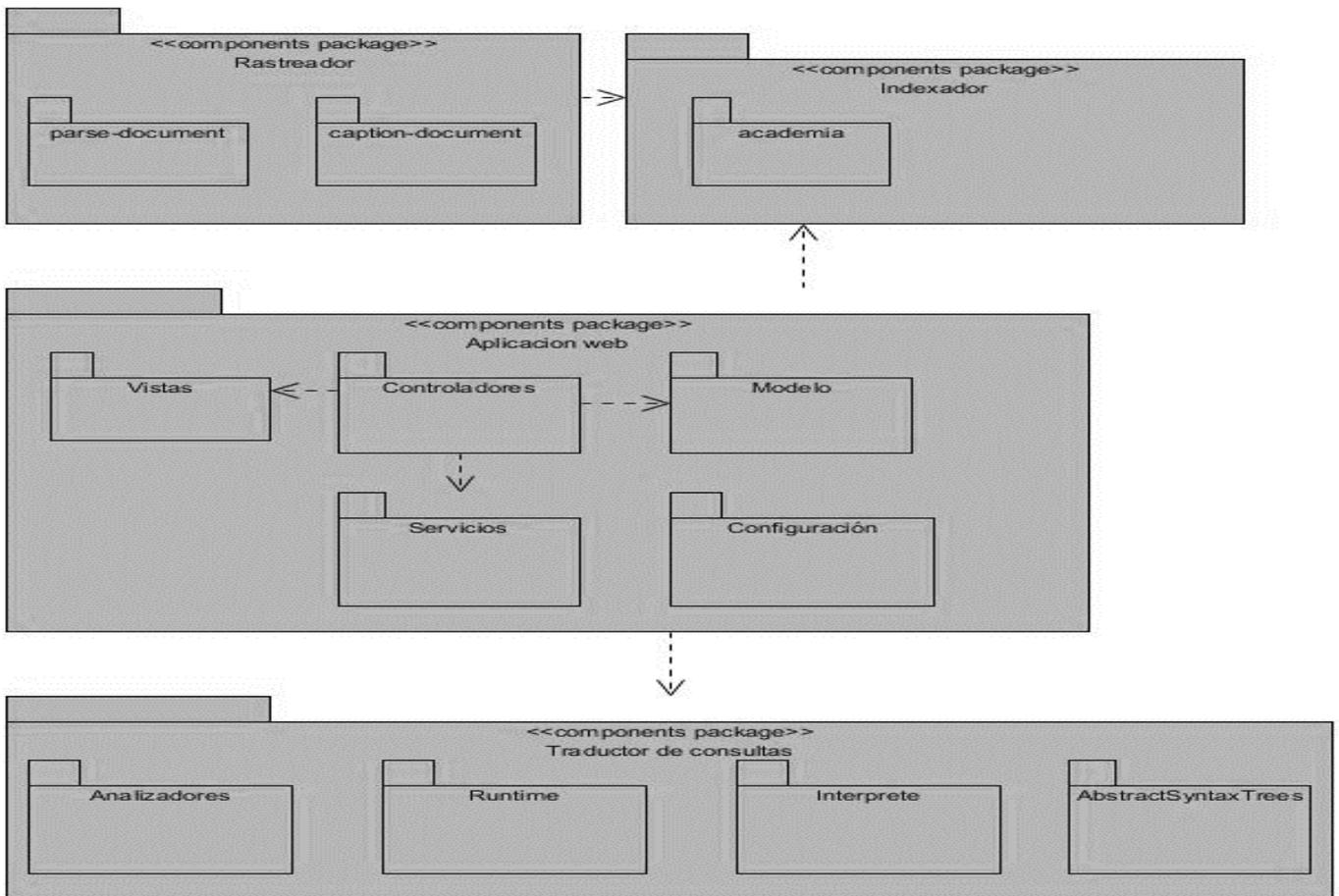


Figura 9 Diagrama de componentes del módulo de búsqueda avanzada

El diagrama mostrado anteriormente está compuesto por cuatro paquetes, que representan los componentes del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a:

- **Rastreador:** Implementa la lógica de la búsqueda de documentos académicos en el dominio .cu.
- **Indexador:** Contiene los componentes de configuración que permitirán la indexación de documentos y la comunicación con el rastreador y la aplicación web.
- **Aplicación web:** Incluye paquetes que contienen los controladores, vistas, servicios y configuración de la aplicación web.
- **Traductor de consultas:** Contiene los componentes del sistema encargados de transformar el criterio de búsqueda introducido por el usuario en una consulta para el indexador.

Como muestra la siguiente imagen, los componentes físicos del paquete Aplicación web están divididos en cinco subpaquetes: Vistas, Controladores, Servicios, Modelo y Configuración.

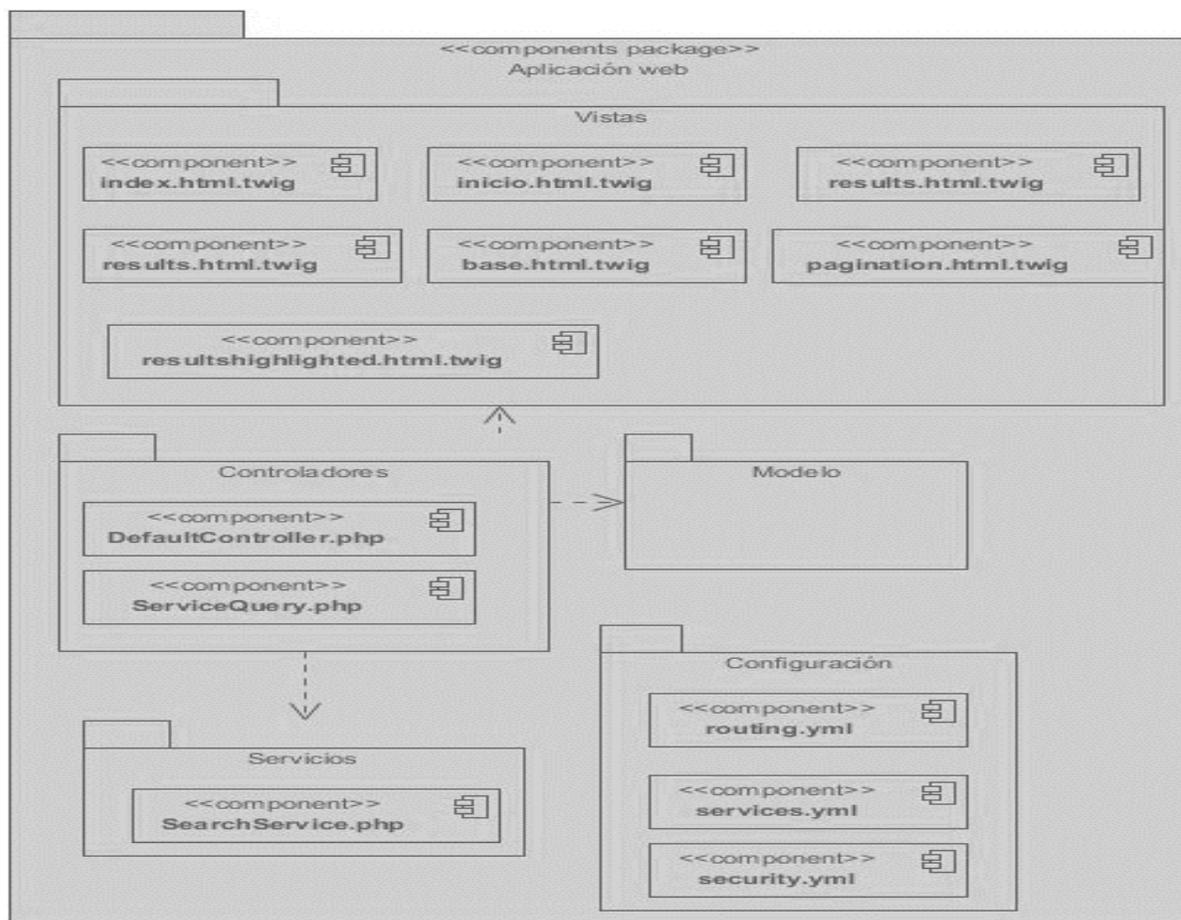


Figura 10 Diagrama de componentes de la Aplicación web

Dentro del paquete **Vistas** se incluyen los componentes donde se muestra la opción de búsqueda avanzada, el formulario de la búsqueda avanzada y los resultados de dicha búsqueda, en los siguientes componentes: **index.html.twig**, **inicio.html.twig**, **results.html.twig** y **base.html.twig**.

El paquete **Controladores** se incluyen los componentes **DefaultController.php** y **ServiceQuery.php**, los cuales contienen las clases encargadas de controlar las peticiones de los usuarios, crear las vistas a cada una de ellas y manejar los servicios del sistema.

Incluido en el paquete **Servicios**, se encuentra el componente **SearchService.php**, el cual contiene la clase encargada de realizar las peticiones a Solr a partir de la consulta traducida por el Traductor de consultas.

Modelo, es el paquete encargado de la comunicación con el componente de indexación y la representación de la información almacenada en él.

En el paquete **Configuración** se encuentran los archivos de configuración correspondientes a la aplicación web como son:

- **routing.yml**: Contiene las rutas que utilizará el sistema.
- **services.yml**: Contiene la configuración de los servicios del sistema.
- **security.yml**: Contiene la configuración de la seguridad del sistema.

3.2 Estándares de codificación utilizados

Para facilitar el trabajo conjunto, se siguen estándares de codificación que permiten la lectura rápida y simple del código, influir en la calidad del software contribuyendo a una adecuada gestión del código fuente. En el caso de la aplicación web se utiliza el estándar definido por la comunidad de desarrollo de Symfony y para la implementación de los *plugins* para Nutch se emplea el estándar definido para el lenguaje Java. Symfony2 tiene como lenguaje PHP el cual sigue los estándares definidos en los documentos PSR-1, PSR-2, PSR-3, PSR-4 (SensioLabsNetwork, 2014; PHP Framework Interop Group (psr-1), 2014; PHP Framework Interop Group (psr-2), 2014; PHP Framework Interop Group (psr-4), 2014). Entre sus elementos más comunes se encuentran (PHP-FIG, 2018):

- Añadir un espacio después de cada delimitador coma `,`.
- Añadir un único espacio a ambos lados de un operador: `=`, `==`, `&&` etc.
- En los arreglos multilínea, añada una coma al final de cada elemento, incluido el último.
- Añade un salto de línea antes de una sentencia `return`, a menos que el `return` se encuentre solo en un bloque de sentencias, y un salto después de cada llave cierre de sentencia, excepto después de la llave de cierre de clase.
- Usa notación CamelCase sin guiones bajos en variables, funciones, métodos y argumentos.
- El nombre de las clases se realiza en UpperCamelCase, es decir, que comienza por mayúscula.

3.3 Validación del módulo

La validación del módulo incluye actividades que permiten asegurar que el software desarrollado se corresponda con los requisitos establecidos por el cliente. Una de las actividades son las pruebas de validación, que tiene como objetivo evaluar la calidad, detección de las no conformidades respecto a funcionalidades de la aplicación y la correcta integración de los componentes de la arquitectura. Se aplicarán pruebas funcionales, de seguridad utilizando la herramienta Acunetix, de integración y de carga y estrés utilizando la herramienta JMeter con el objetivo de evaluar la calidad del sistema al detectar las no conformidades y darle solución.

3.3.1 Pruebas funcionales

Las pruebas funcionales se aplican sobre un software con el objetivo de validar que las funcionalidades implementadas funcionen según las especificaciones de los requisitos que se definieron. Por lo general se emplean dos métodos fundamentales: el método de Caja Negra y el método de Caja Blanca (Pressman, 2010).

El método de Caja Negra permite a los probadores enfocar su atención en el funcionamiento de la interfaz, con el análisis de datos de entrada y los de salida, y el de Caja Blanca se centra en las pruebas de código de las aplicaciones. A continuación, se muestra el diseño de casos de pruebas correspondiente a la funcionalidad “Realizar búsqueda académica avanzada”.

Tabla 5 Variables empleadas en el diseño del caso de prueba basado en el CU “Realizar búsqueda académica avanzada”

No	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Título	Campo de texto	No	Se inserta una cadena de texto de no más de 50 caracteres, puede contener valores alfanuméricos pero no caracteres extraños.

2	Autor	Campo de texto	No	Se inserta una cadena de texto de no más de 30 caracteres, no puede contener caracteres extraños.
3	Revista	Campo de texto	No	Se inserta una cadena de texto de no más de 50 caracteres, puede contener valores alfanuméricos pero no caracteres extraños.
4	Frase	Campo de texto	No	Se inserta una cadena de texto de no más de 50 caracteres, puede contener valores alfanuméricos pero no caracteres extraños.
5	Palabra	Campo de texto	No	Se inserta una cadena de texto de no más de 15 caracteres, no puede contener caracteres extraños.
6	Todas las palabras	Campo de texto	No	Se inserta una cadena de texto de no más de 50 caracteres, puede contener valores alfanuméricos pero no caracteres extraños.
7	Sin palabra	Campo de texto	No	Se inserta una cadena de texto de no más de 15 caracteres, no puede contener caracteres extraños.
8	Fecha	Campo de selección	No	Se selecciona un año específico de un documento.
9	Formato	Campo de selección	No	Se selecciona un formato específico.
10	Idioma	Campo de selección	No	Se selecciona un idioma específico.

Las celdas contienen V (indica válido), I (indica inválido), N/S (No es necesario llenar).

Tabla 6 Caso de prueba correspondiente al CU “Realizar búsqueda académica avanzada”

Escenario	Descripción	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	R/ del sistema	Flujo Central
EC1.1 Introducir datos correctamente	El usuario introduce los datos correctamente en el formulario de búsqueda avanzada.	V	V	V	V	V	V	V	N/S	N/S	N/S	El sistema realiza la consulta correctamente	El usuario rellena los campos correctamente y da clic en el botón de buscar
EC1.3 Introducir campos incorrectos	El usuario introduce uno o más valores incorrectos	I	V	V	V	V	V	V	N/S	N/S	N/S	El sistema muestra un mensaje de error “El Título no	El usuario introduce uno o más valores incorrectos y da clic

												puede conten er caract eres extrañ os".	en el botón de buscar
--	--	--	--	--	--	--	--	--	--	--	--	-----------------------------------------------------------	--------------------------------

Resultados de las pruebas funcionales

Para la identificación de las no conformidades presentes en el sistema se diseñaron 3 casos de prueba que se ejecutaron en 2 iteraciones donde se encontraron 10 no conformidades de las cuales todas fueron resueltas y en la segunda iteración y final no se detectaron no conformidades. En la Tabla 7 se muestran los resultados obtenidos en cada iteración de prueba al módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.

Tabla 7 Resultados por iteración

No conformidades	1ra Iteración	2da Iteración
Detectadas	10	0
Resueltas	10	0

Entre las no conformidades identificadas, los principales errores detectados se corresponden a:

- El sistema muestra títulos que no corresponden con el introducido debido al empleo de las conjunciones y preposiciones.
- El sistema muestra las revistas al introducir el enlace.
- El sistema muestra mensajes de error con datos sobre las variables.
- Errores de ortografía.

Las no conformidades de ortografía estaban relacionadas con problemas de acentuación, para resolver se corrigió el código fuente de los errores ortográficos. Para corregir las no conformidades con el campo título

se aplicó el stopword y para solucionar el problema con el campo revista se realizó una modificación en su código para obtener solo el nombre de las revistas.

3.3.2 Pruebas de integración

Las pruebas de integración son definidas para verificar el correcto ensamblaje entre los distintos módulos que conforman un sistema informático. Las mismas validan que estos componentes realmente funcionan juntos, son llamados correctamente y, además, transfieren los datos correctos en el tiempo preciso y por las vías de comunicación establecidas (Sommerville, 2005).

Una vez realizadas las pruebas funcionales a cada componente interno de manera independiente, y verificado que las funcionalidades implementadas se corresponden de acuerdo a los requisitos funcionales y no funcionales que se establecieron; se pudo comprobar el correcto funcionamiento de los componentes mediante el estudio del flujo de datos entre ellos.

Para la realización de las pruebas de integración se llevaron a cabo diferentes acciones, a continuación, se mencionan las fundamentales:

- Verificación de la conexión de Nutch y Solr para indexar los metadatos extraídos de los documentos.
- Comprobar el enlace entre Solr y la interfaz de resultados para verificar si se muestran todos los documentos almacenados en Solr.

La ejecución de las pruebas de integración permitió verificar el trabajo conjunto de los componentes del sistema. Para detectar incoherencias en el funcionamiento de la aplicación se verificó la correcta integración entre los componentes del módulo. Se puede concluir que existe una correcta integración entre la interfaz web, el mecanismo de indexación Solr y el mecanismo de rastreo Nutch, no encontrándose ninguna no conformidad.

3.3.3 Pruebas de seguridad

La seguridad informática comprende la puesta en práctica de un conjunto de medidas preventivas y reactivas en los sistemas informáticos y tecnológicos, que posibilitan la protección de la información, persiguiendo como objetivo principal la integridad, confidencial y disponibilidad de la misma. Terminadas las pruebas es posible medir y cuantificar los riesgos a los cuales se ven expuestos aplicativos de la infraestructura interna y externa (Pressman, 1988)

Al módulo desarrollado se le realizaron una serie de pruebas de seguridad mediante el software Acunetix, las cuales se presentan a continuación:

- Ataques de inyección.
- Cross-Site Scripting (XSS).
- Falsificación de petición (CSRF).
- Detección de ficheros y directorios.

Resultados de las pruebas de seguridad

Se aplicó el nivel 2 de pruebas de seguridad, en una primera iteración se detectaron 20 alertas en total, ninguna de clasificación alta, 5 de clasificación media, 5 de clasificación baja y 10 informativas. En la segunda iteración el software no arrojó alertas existentes en el módulo. A continuación, se muestra una tabla con los tipos de vulnerabilidades encontradas en el entorno y en la Figura 11 se puede apreciar el comportamiento de las alertas encontradas en las pruebas funcionales ejecutadas.

Tabla 8 Vulnerabilidades del sistema

Tipo	Descripción	Recomendaciones
Software	La versión de PHP instalada contiene vulnerabilidades que han sido solucionadas en versiones más recientes.	Actualizar a una versión superior.
Revelación de la versión del servidor web en las páginas de error.	Las páginas de error contienen la versión del servidor web y una lista de los módulos habilitados, lo que puede conducir a la realización de ataques.	Esconder la cabecera Server.

Ausencia del encabezado X-Frame-Options	permite a un atacante realizar ataques Clickjacking	La mejor forma de añadir esta cabecera sería añadiendo unas líneas de código al archivos functions.php o especificar en los encabezados de respuesta HTTP el campo X-Frame-Options.
------------------------------------------------	-----------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

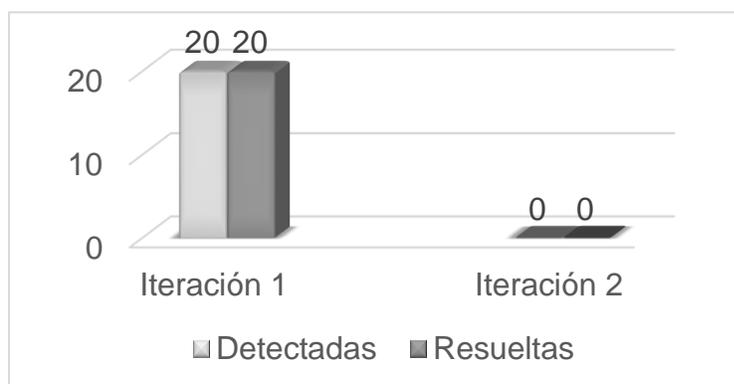


Figura 11 Comportamiento de las alertas

Las vulnerabilidades que se presentaron fueron resueltas en el servidor. Los resultados obtenidos en las pruebas de seguridad, fueron satisfactorios, llegando a la conclusión que la herramienta desarrollada es segura y está en condiciones de ser usada por los usuarios.

3.3.4 Pruebas de carga y estrés

Las pruebas de carga permiten probar el funcionamiento del software bajo condiciones extremas, las pruebas de estrés están diseñadas para enfrentar el programa en condiciones anormales. Para el módulo de búsqueda avanzada para el subsistema académico es necesario realizar dichas pruebas pues resulta necesario comprobar el rendimiento del sistema soportando una cantidad máxima de usuarios. Para la realización de las pruebas se utiliza la herramienta JMeter (Sommerville, 2011).

En la prueba realizada se definieron 2 casos que simulaban el acceso de 1000 y 1500 usuarios respectivamente. Se realizaron en una maquina Core i7 a 2.40 Ghz de velocidad de procesador, una memoria RAM de 8.00Gb.

Para un mejor entendimiento de los componentes se explica cada parámetro que la compone:

- **#Muestras:** cantidad de hilos utilizados para la URL.
- **Media:** tiempo promedio en milisegundos para un conjunto de resultados.
- **Min.:** tiempo mínimo que demora un hilo en acceder a una página.
- **Max.:** tiempo máximo que demora un hilo en acceder a una página.
- **Rendimiento:** rendimiento medido en los requerimientos por segundo / minuto / hora.
- **Kb/sec:** rendimiento medido en Kbytes por segundo.

Resultados de las pruebas de carga y estrés:

Los valores obtenidos para 1000 hilos (usuarios) se muestran en la siguiente tabla:

Tabla 9 Resultado de la prueba con 1000 usuarios

TOTAL	#Muestras	Media	Min	Max	%Error	Rendimiento	Kb/sec
	20900	13758	60	60130	0.0	11.4/segundos	61.3

El primer caso para 1000 usuarios conectados de forma concurrente arrojó una respuesta de 20900 peticiones al servidor en un promedio de 13758 milisegundos, lo que equivale a 11.4 peticiones por segundo.

Los valores obtenidos para 1500 hilos (usuarios) se muestran en la siguiente tabla:

Tabla 10 Resultado de la prueba con 1500 usuarios

TOTAL	#Muestras	Media	Min	Max	%Error	Rendimiento	Kb/sec
	23281	18569	30	84950	0.029	15.6/segundos	79.8

Como muestra la tabla, el segundo caso para un total de 1500 usuarios conectados se forma concurrente arrojó una respuesta de 23281 peticiones al servidor en un promedio de 18569 milisegundos, lo que equivale a 15.6 peticiones por segundo.

Los resultados satisfactorios de la aplicación de estas pruebas permiten que se llegue a la conclusión de que el sistema brinda un correcto funcionamiento ante una elevada carga en condiciones anormales. De esta forma se da cumplimiento a los RNF 19 y 20.

3.3.5 Aceptación por usuarios

Para la evaluación de satisfacción de los usuarios ante el módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a, se utilizó la Técnica de ladov. La técnica de ladov constituye una vía indirecta para el estudio de satisfacción, ya que los criterios se fundamentan por tres preguntas cerradas, que se intercalan dentro de un cuestionario y se relacionan a través de la Tabla 12 que se muestra a continuación denominada según la técnica el “Cuadro Lógico de ladov” (López, y otros, 2002).

Para realizar las encuestas se seleccionaron un total de 20 personas para realizar la búsqueda académica a través de la búsqueda avanzada.

Tabla 11 Cuadro Lógico ladov

	¿Considera usted que el proceso de búsqueda académica mejoró luego de integrarse el módulo de búsqueda avanzada en la plataforma c.u.b.a?								
	Sí			No sé			No		
	¿Utilizaría la aplicación desarrollada para ejecutar la búsqueda de artículos científicos-académicos en la red cubana?								
¿Le satisfacen los resultados brindados por la aplicación en relación a los documentos buscados?	Sí	No sé	No	Sí	No sé	No	Sí	No sé	No
Me gusta mucho	1	2	6	2	2	6	6	6	6

Me gusta más de lo que me disgusta	2	2	3	2	3	3	6	3	6
Me da lo mismo	3	3	3	3	3	3	3	3	3
Me disgusta más de lo que me gusta	6	3	6	3	4	4	3	4	4
No me gusta nada	6	6	6	4	4	6	4	5	6
No sé qué decir	2	3	6	3	3	3	6	3	4

Preguntas abiertas

¿Qué elementos consideras positivos de la aplicación desarrollada?

¿Qué elementos consideras negativos de la aplicación desarrollada?

¿Propondrías añadir alguna funcionalidad a la aplicación desarrollada?

El número resultante de la interrelación de las tres preguntas indica la posición de cada encuestado en la escala de satisfacción (López, y otros, 2002). Aplicados a la muestra se obtienen:

- Clara satisfacción – 15
- Más satisfecho que insatisfecho -5
- No definida- 0
- Más insatisfecho que satisfecho -0
- Clara insatisfacción -0
- Contradictoria -0

Para obtener el índice de satisfacción grupal (ISG) se presenta una escala numérica que va desde +1 considerada como máxima satisfacción, hasta -1 como máxima insatisfacción. (López, y otros, 2002) Para obtener el ISG se trabaja con la fórmula que se muestra a continuación:

$$ISG = \frac{A(+1) + B(+0,5) + C(0) + D(-0,5) + E(-1)}{N}$$

N: cantidad de usuarios encuestados

A: cantidad de usuarios con Clara satisfacción

B: cantidad de usuarios Más satisfechos que insatisfechos

C: cantidad de usuarios No definidos

D: cantidad de usuarios Más insatisfechos que satisfechos

E: cantidad de usuarios con Clara insatisfacción

Se trabajó con los diferentes niveles de satisfacción de la siguiente forma: (López, y otros, 2002)

Tabla 12 Nivel de satisfacción

+1	Máximo de satisfacción
0.5	Más satisfecho que insatisfecho
0	No definido y contradictorio
-0.5	Más insatisfecho que satisfecho
-1	Máxima insatisfacción

El resultado del ISG = (0.88) que demuestra el grado de satisfacción alto declarado por los usuarios al hacer uso de la propuesta de solución.

3.4 Conclusiones del capítulo

En este capítulo se han abordado los elementos de la implementación del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a, las pruebas realizadas al mismo y los resultados obtenidos; arribando a las siguientes conclusiones:

- La elaboración del diagrama de componente, permitió una mejor comprensión de la estructura de los componentes del sistema implementado.
- El correcto uso de los estándares de codificación permitió que el código del sistema desarrollado fuera legible para lograr una fácil y mejor comprensión del mismo, la cual es de utilidad para el mantenimiento del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.
- El proceso de validación de software arrojó como resultado que el sistema implementado responde a los requerimientos definidos por el cliente y cuenta con la calidad necesaria para ser utilizado.

CONCLUSIONES

Una vez realizada la investigación, el diseño y la implementación de la solución, se obtuvieron resultados que permiten arribar a las siguientes conclusiones:

- El estudio de buscadores académicos con una búsqueda avanzada permitió identificar los principales filtros de búsqueda para la implementación del módulo de búsqueda avanzada para el subsistema académico de la plataforma c.u.b.a.
- La correcta selección de la metodología y tecnologías permitió analizar, describir e implementar adecuadamente el sistema propuesto.
- La elaboración de artefactos que contribuyeron al diseño de la propuesta de solución, permitieron una mayor comprensión del sistema; garantizando la estructura base para la organización lógica del código fuente y la disminución del impacto ante futuras modificaciones en la aplicación.
- La aplicación de las pruebas de software realizadas permitió erradicar las insuficiencias detectadas en la herramienta desarrollada logrando así un producto más seguro y funcional conforme a las necesidades de los usuarios.

RECOMENDACIONES

- Integrar el módulo de búsqueda académica avanzada en el entorno real de la plataforma c.u.b.a.
- Implementar boletines que contengan los nuevos números de las revistas indexadas, teniendo en cuenta según los criterios de búsqueda las revistas de preferencia de los usuarios autenticados en la plataforma c.u.b.a.
- Adicionar nuevos filtros de búsqueda académica avanzada que permitan obtener resultados de búsquedas más exhaustivos.

BIBLIOGRAFÍA

- **10GEN.** MongoDB. [En línea] 2011. [Consulta: enero 2018] Disponible en: <http://www.mongodb.com>.
- **ACUNETIX.** Audit your website security with Acunetix Web Vulnerability Scanner. [En línea].2015. [Consulta: 3 enero 2018] Disponible en: <http://cloudnet.com.kh/blog-09032014.html>.
- **BENJUMEA, M.M.** Los metadatos. s.l. : Ministerio de Educación Nacional República de Colombia, [En línea] 2008. [Consulta: febrero 2018] Disponible en: http://aprendeenlinea.udea.edu.co/lms/men/docsoac3/0301_metadatos
- **BLÁZQUEZ, M.** Modelo entidad-relación ER. Fundamentos y Diseño de Base de Datos. 2014.
- **CARRÓN, J.M. ANDALUCIA : S.N.,** Metabuscadores: a propósito de un estudio, comparativo. Consejería de salud, [En línea] 2009. [Consulta: enero 2018] Disponible en: http://www.bvsspa.es/profesionales/sites/default/files/documentos/02_Metabuscadores_CC.pdf.
- **CRAIG, L.** UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado. s.l. : Madrid: Pearson Educación, 2003.
- **CRUZ, V. Y GUTIERREZ, D.**Diagrama de componentes. s.l. : SlideShare, [En línea] 2012. [Consulta: marzo 2018] Disponible en: <https://es.slideshare.net/uitron/diagrama-de-componentes-7551535>.
- **DIALNET, FUNDACION.** Manual de uso Dialnet. Una herramienta fundamental para la búsqueda de información de calidad Rioja : s.n., 2017.
- **FANEGO, R.M.** Técnicas Experimentales. E.U. de Ingeniería Técnica Aeronáutica, Nivel: Diplomatura. [En línea] 2007.[Consulta: febrero 2018] Disponible en: <http://ocw.upm.es/fisica-aplicada/tecnicas-experimentales>
- **FERNÁNDEZ, H.** MOTORES DE BÚSQUEDA TIPOS DE BUSCADORES. s.l. : Universidad Nacional Abierta Direcciones de Investigación y Postgrado, 2013.
- **FERRER, J., GARCIA, V. Y GARCIA, R.** Curso completo de HTML. s.l. : Free Software, [En línea] 2017. [Consulta: enero 2018] Disponible en: <http://es.tldp.org/Manuales-LuCAS/doc-curso-html/doc-curso-html.pdf>.

- **FOUNDATION-JMETER, THE APACHE SOFTWARE.** Apache Jmeter. [En línea] 2015. [Consulta: abril 2018] Disponible en: <https://jmeter.apache.org>.
- **FOUNDATION-TIKA, THE APACHE SOFTWARE.** Apache Solr. [En línea] 2014. [Consulta: abril 2018] Disponible en: <https://archive.apache.org/dist/lucene/solr/ref-guide/apache-solr-ref-guide-7.0.pdf>.
- **GARCIA, A. Y MINETTO, E.L.** Doctrine en la práctica. s.l. : Leanpub, 2016.
- **GIMENO, J.M. Y GONZÁLEZ, J.L.** Introducción a Netbeans. [En línea] 2011.[Consulta: enero 2018] Disponible en: <http://ocw.udl.cat/enginyeria-i-arquitectura/programacio-2/continguts-1/1-introduccioi81n-a-netbeans.pdf>.
- **GIORDANINO, E.P.** Sistemas de recuperación de información. Buenos Aires : Tecnicatura Superior en Bibliotecología, Instituto de Formación Técnica Superior N° 13,Ministerio de Educación., 2010.
- **GONZALEZ, L.C. Y POMPA, E.R.** Extensión de Visual Paradigm for UML para el Desarrollo Dirigido por Modelos de aplicaciones de gestión de información. s.l. : Ingeniería de software, 2012.
- **GOOGLE.** Google Académico. [En línea] 19 de noviembre de 2005. [Consulta: diciembre 2017] Disponible en: <https://scholar.google.com.cu/>.
- **HAMON, H.** Applying Design Ppterns to Symfony. Estambul : Turquía: SensioLabs, [En línea] 2014. [Consulta: diciembre 2017] Disponible en: <https://speakerdeck.com/hhamon/applying-design-patterns-to-symfony>.
- **I, UNIVERSITAT JAUME.** Herramientas Informáticas para documentación. s.l. : Facultad de Ciencias Humanas y Sociales, 2006.
- **JESÚS, L.** Portal web del Partido Comunista de Cuba. La Habana : Tesis de Diplomado, Universidad de las Ciencias Informáticas, 2015.
- **JOHNSON, L.** WorldWideScience.org. Zaragoza, [En línea] 2017. [Consulta: diciembre 2017] Disponible en: <https://cienciayficcio.wordpress.com/2009/09/04/worldwidescience-or>.
- **KABIR, M. J.** La biblia del Servidor Apache. Madrid : ANAYA Multimedia, 2002.

- **KUNA, H., REY, M. Y CORTES, J.** Generación de sistemas de recuperación de información para la gestión documental en el área de las ciencias de la computación. Parana : s.n., 2013.
- **KUNA, H., REY, M. Y MARTINI, E.** Desarrollo de un Sistema de Recuperación de Información para Publicaciones Científicas del Área de Ciencias de la Computación. s.l. : 1Departamento de Informática, Fac. de Cs. Exactas, Químicas y Naturales. Universidad Nacional de Misiones. Argentina 2Facultad de Ingeniería – Universidad Nacional de Itapúa, Paraguay, 2014.
- **LEAL, R.** Buscadores Académicos. Chile : s.n., [En línea] 2007. [Consulta: enero 2018] Disponible en: <https://es.slideshare.net/rosalealg/buscadores-acadmicos>
- **LÓPEZ, A. Y GONZÁLEZ, V.** “La técnica de ladov. Una aplicación para el estudio de la satisfacción de los alumnos por las clases de educación física”. abril 2002.
- **MARIÑO, C.** Programación en PHP5. Nivel Básico. [En línea]. 2008. [Consulta: enero 2018] Disponible en: http://administraciondesistemas.pbworks.com/f/Manual_PHP5_Basico.pdf.
- **MARTÍNEZ, F.J.** RECUPERACIÓN DE INFORMACIÓN: MODELOS, SISTEMAS Y EVALUACIÓN. Murcia : EL KIOSKO JMC, 2004.
- **NAVARRETE, T.** El lenguaje JavaScript. [En línea]. 2006. [Consulta: enero 2018] Disponible en: <http://www.dtic.upf.edu/~tnavarrete/fcsig/javascript.pdf>.
- **PARADIGM, VISUAL.** Visual Paradigm for UML. [En línea] Visual Paradigm, 2017. [Consulta: diciembre 2017] Disponible en: <http://www.visual-paradigm.com/product/vpuml/>.
- **PEREZ, E.** Desarrollo web agil con Symfony2. [En línea] 2013. [Consulta: febrero 2018] Disponible en: <http://allmastersolutions.com/shared/Desarrollo%20Agil%20-%20Symfony-2-1.pdf>.
- **PEREZ, J.** Introducción a CSS. [En línea] 2008. [Consulta: febrero 2018] Disponible en: http://www.jesusda.com/docs/ebooks/introduccion_css.pdf.
- **PHP-FIG.** PHP Standards Recommendations. [En línea] PHP Framework Interop Group, 2018. [Consulta: febrero 2018] Disponible en: <https://www.php-fig.org/psr/>.
- **PRESSMAN, R. S.** Ingeniería del Software. Un enfoque práctico. 7ª ed. México : D.F: The MacGraw-Hill Companies, 2010.

- **RODRÍGUEZ, P., DOLORES, A. Y ROJAS, S.** Arquitectura de software para el sistema de visualización médica Vismedic. s.l. : Revista Cubana de Informática Médica, 2016.
- **RODRÍGUEZ, P., VILTRES, H. Y PONS, L.A.** Componentes y funcionalidades de un sistema de recuperación de la información. La Habana : s.n., 2016. Vols. Vol. 10 No. Especial Informática 2016, mayo, 2016.
- **SÁNCHEZ, T.** Metodología de desarrollo para la Actividad productiva de la UCI v1.2. La Habana: Universidad de Ciencias Informáticas : s.n., 2015.
- **SCIELO.** SciELO. Scientific Electronic Library Online. [Apache Solr: una introducción <http://scielo.org>.
- **SETA, L. D.** Apache Solr: una introducción. [En línea] 2010. [Consulta: marzo 2018] Disponible en: <https://dosideas.com/noticias/java/913-apache-solr-una-introduccion>.
- **SOMMERVILLE, I.** Ingeniería de Software. 2005. 7ma edición.
- **SOMMERVILLE, I.** Ingeniería de Software. México : Pearson Educación de México, 2011. ISBN: 84-7829-074-5.
- **SPRINGERLINK.** SpringerLink. [En línea] 2017. [Consulta: enero 2018] Disponible en: <https://link.springer.com>.
- **TOCORORO.** Tocator alfa Publicaciones Científicas Cubanas. [En línea] Universidad de Pinar del Río "Hermanos Saíz Montes de Oca", 2017.[Consulta: mayo 2018] Disponible en: <http://tocororo.upr.edu.cu/catalogo>.

ANEXOS

Anexo 1

Tabla 13 Variables correspondientes al caso de prueba: "Integrar Nutch-Solr".

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Título	Variable extraída del <i>parser</i> de Nutch.	NO	Contiene información referente al título del documento.
2	Autor	Variable extraída del <i>parser</i> de Nutch	NO	Contiene información referente al nombre del autor.
3	Revista	Variable extraída del <i>parser</i> de Nutch	NO	Contiene información referente a la revista a la que pertenece.
4	Fecha	Variable extraída del <i>parser</i> de Nutch	NO	Contiene información referente a la fecha de publicación.
5	Formato	Variable extraída del <i>parser</i> de Nutch	NO	Contiene información referente al formato del documento.
6	Idioma	Variable extraída del <i>parser</i> de Nutch	NO	Contiene información referente al idioma.

Tabla 14 Caso de prueba: "Integrar Nutch-Solr".

Escenario	Descripción	V1	V2	V3	V4	V5	V6	Respuesta del sistema	Flujo central
EC 1.1 Se indexan los metadatos	El rastreador indexa los	Instituciones	Carloss	instituciones	2016	PDF	es	Metadatos del	Montar un servidor de prueba con

del documento correctamente.	metadatos extraídos del documento correctamente en Solr.		Yordan					documento.	un documento que contenga los metadatos introducidos.
EC 1.2 No se indexan los metadatos del documento.	El rastreador no indexa los metadatos extraídos del documento.	Instituciones	Carlos Yordan	Instituciones	2016	PDF	Es	El sistema muestra el mensaje "Job Failed!"	Se identifica la url del servidor donde está el documento que se va a recuperar y se inserta la url en el semillero de Nutch. Ejecutar el comando: "bin/nutchcrawl urls -dir crawl -solr ruta de Solr -depth1 -topN 1" donde la ruta de Solr

									es la ruta del servidor de Solr.
--	--	--	--	--	--	--	--	--	----------------------------------

Anexo 2:

Preguntas de la entrevista realizada al cliente:

Nombre: Eric Bárbaro Utrera Sust.

Preguntas:

1. ¿Por qué la necesidad de crear un buscador cubano?
2. ¿Cómo funciona el motor de búsqueda de la plataforma c.u.b.a?
3. ¿Cuál es la base tecnológica que se empleó para el desarrollo de la plataforma?
4. ¿Cuáles son las deficiencias que posee actualmente el subsistema de búsqueda académica de la plataforma?