

Universidad de las Ciencias Informáticas

**Facultad 4**



**Módulo para la búsqueda y recuperación de información en el Repositorio de Recursos Educativos RHODA V3.0**

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

**Autor:**

Jeimel Cosme Hernández Moran

**Tutor:**

Ing. Leonardo Rodríguez González

**La Habana, junio 2014**

**“Año 56 de la Revolución”**

## Declaración de Autoría

# Módulo para la búsqueda y recuperación de información en RHODA V3.0

Declaro que soy el **único** autor del trabajo “**Módulo para la búsqueda y recuperación de información en el Repositorio de Recursos Educativos RHODA V3.0**” y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo el presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del 2014

---

**Autor:**

**Jeimel Cosme Hernández Moran**

---

**Tutor:**

**Ing. Leonardo Rodríguez González**

*Dedicatoria*

*A todos aquellos que han hecho posible y han contribuido con su desarrollo.*

*A mi familia.*

*A mis amigos.*

*Agradezco:*

*A mi familia por apoyarme en mis decisiones y por convertirme en la persona que soy hoy en día.*

*A mis padres (Ana y Roberto) que han hecho hasta lo imposible para enseñarme como hacer las cosas en la vida, las primeras personas que confiaron en mí, quienes constituyen mi guía, mi desvelo.*

*A mi abuela (María) y a mi madrina (Alicia) por ser las abuelitas más lindas del mundo siempre demostrándome su afecto y cariño.*

*A mis hermanos de sangre (Jeissy, Jeidel, Jeimara) que siempre estuvieron presente dándome todo lo que estuviera a su alcance para cumplir con las metas propuestas.*

*A mi madre blanca (Tania) y a mi hermano (Carlos Ariel) que aunque la distancia nos separa en estos momentos siempre me brindaron su apoyo y cariño.*

*A mis amistades que actualmente están fuera de Cuba (Omarrito, Milena), siempre me apoyaron a pesar de la distancia que nos separa.*

*A la incalculable y valiosa ayuda de mi tutor Leonardo Rodriguez, por su entrega total y desinteresada, por haber estado guiándome para obtener estos resultados satisfactorios.*

*A mis socios de la cueva Alejandro, José Ángel, Orelvis, Yoandrys, Yordanis, Raul Noa, Raul Infante, Yosvani, Marcos, Omar y Edelfo.*

*A mis amistades Rosalina, Dainier, Gloria, Liema, Pavel, Edwin, Jose Alberto, Carlos, Dariel, Maday y la vaquita de mi vida con las que he compartido maravillosas experiencias y de las que me llevo los mejores recuerdos. A mis compañeros de grupo, por el tiempo que hemos compartido, las risas y los estudios.*

*A los maestros y profesores que han hecho posible mi educación, los responsables de que tantos años de esfuerzos obtengan sus frutos, especialmente al profesor Fiol que siempre me apoyó cuando lo necesitaba.*

*A todos aquellos que aportaron su granito de arena y siempre confiaron en mí*

*A todos ¡Muchas Gracias!*

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

Los Repositorios de recursos educativos son la infraestructura clave para el almacenamiento, localización y recuperación de recursos educativos. Varias iniciativas existen a nivel mundial (Agrega, Merlot y Connexions,) y algunos centros de la educación superior en Cuba también poseen sistemas con características similares a estos. La Universidad de las Ciencias Informáticas cuenta con el repositorio RHODA, cuyo objetivo fundamental radica en almacenar y gestionar los objetos de aprendizaje (OA). Debido a que otros productos del centro Tecnologías y Herramientas para la Formación (FORTES) manejan recursos que no son comprendidos como OA, se hace necesaria la implementación de la versión 3.0 del repositorio RHODA, siguiendo una nueva filosofía de desarrollo y haciendo uso de las recientes tecnologías con el fin de optimizar los procesos que se realizan en este sistema. El presente trabajo tiene como objetivo el desarrollo de un módulo para la realización de búsquedas y recuperación de información en el repositorio RHODA 3.0. Para ello se analizan los referentes teóricos actuales relacionados con el proceso de recuperación de la información. El proceso de desarrollo es guiado por la metodología de desarrollo *Rational Unified Process* (RUP), PHP, JavaScript y CSS como lenguajes de programación, como *framework* de desarrollo Symfony 2.3.7 y la librería Lucene para la realización de búsquedas. Se emplean técnicas como la búsqueda aproximada y la exacta, además del uso del índice invertido para el proceso de indexación, ordenando los resultados mediante el modelo vectorial. Finalmente el sistema es validado mediante la ejecución de un conjunto de pruebas.

**Palabras clave:** recuperación de información, recursos educativos, repositorios.

|   |    |
|---|----|
| Introducción .....  | 1  |
| Capítulo 1: Fundamentación teórica .....  | 7  |
| Introducción.....   | 7  |
| 1.1 Recuperación de información .....   | 7  |
| 1.2 Modelos de recuperación de información .....                                      | 8  |
| 1.2.1 Sistema de recuperación de información de búsqueda exacta .....                 | 8  |
| 1.2.2 Búsqueda aproximada .....   | 9  |
| 1.3 Repositorio de recursos educativos .....  | 10 |
| 1.3.1 Sistemas similares.....   | 11 |
| 1.3.2 Estándares y especificaciones de contenidos e interoperabilidad.....            | 15 |
| 1.4 Metodología de desarrollo de software, tecnologías y herramientas a utilizar..... | 17 |
| 1.4.1 Metodología de desarrollo de software .....                                     | 18 |
| 1.4.2 Lenguajes de programación.....  | 18 |
| 1.4.3 Lenguajes de programación para el lado del servidor .....                       | 19 |
| 1.4.4 Lenguajes de programación para el lado del cliente. ....                        | 20 |
| 1.4.5 Frameworks para el desarrollo de aplicaciones web .....                         | 21 |
| 1.4.6 Sistema gestor de base de datos.....  | 23 |
| 1.4.7 ORM.....  | 23 |
| 1.5 Librerías para realizar búsquedas .....   | 24 |
| 1.6 Herramientas de software para el desarrollo de la solución .....                  | 25 |
| 1.6.1 Servidor web.....   | 26 |
| Conclusiones parciales.....   | 27 |
| Capítulo 2: Propuesta de solución. Análisis y diseño .....                            | 28 |
| Introducción.....   | 28 |
| 2.1 Propuesta de solución .....   | 28 |
| 2.2 Modelo conceptual .....   | 31 |
| 2.3 Especificación de requisitos.....   | 32 |
| 2.3.1 Requisitos funcionales .....  | 32 |
| 2.3.2 Requisitos no funcionales: .....  | 33 |
| 2.4 Modelo de casos de uso del sistema .....  | 35 |
| 2.4.1 Descripción de casos de uso .....   | 37 |
| 2.5 Modelo del análisis .....   | 40 |
| 2.5.1 Diagramas de clases del análisis.....   | 40 |
| 2.5.2 Diagramas de colaboración .....   | 42 |

|  |  |    |
|--|--|----|
| 2.6  | Patrones de diseño utilizados .....                  | 43 |
| 2.6.1                                      | Patrones de diseño GRASP.....                        | 43 |
| 2.6.2                                      | Patrones de diseño GoF .....                         | 45 |
| 2.6.3                                      | Patrón arquitectónico Modelo Vista Controlador ..... | 46 |
| 2.7  | Modelo del diseño .....                              | 47 |
| 2.7.1                                      | Diagrama de clases del diseño .....                  | 47 |
|  | Conclusiones parciales.....                          | 49 |
| Capítulo 3: Implementación y pruebas ..... |  | 50 |
|  | Introducción.....                                    | 50 |
| 3.2  | Modelo de datos.....                                 | 50 |
| 3.3  | Modelo de implementación.....                        | 52 |
| 3.3.1                                      | Diagrama de componentes .....                        | 52 |
| 3.4  | Pruebas.....   | 54 |
| 3.4.1                                      | Pruebas de Caja blanca .....                         | 54 |
| 3.4.2                                      | Pruebas de Caja negra .....                          | 56 |
| 3.5  | Resultados de las pruebas .....                      | 61 |
|  | Conclusiones parciales.....                          | 62 |
| Conclusiones generales.....                |  | 63 |
| Glosario de términos.....                  |  | 64 |
| Bibliografía referenciada .....            |  | 66 |

## Introducción

El progresivo desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) y el gran cúmulo de información en formato digital, han sido consideradas las razones principales para que el hombre, en el transcurso de su evolución, gestionara progresivamente numerosas vías que ayuden, flexibilicen y al mismo tiempo conviertan su trabajo en un proceso más eficiente.

Esta evolución ha facilitado la existencia de diferentes medios de representación de la escritura, llegando hasta nuestros días, donde la información se representa digitalmente. Siendo posible su almacenamiento y su distribución masiva en forma simple y rápida. La digitalización abrió nuevos horizontes en las formas en que el hombre puede tratar con la información que produce.

En este contexto, cobra singular importancia la recuperación de información, dentro del campo asociado a las Ciencias de la Información. La cual es definida por Ricardo Baeza-Yates y otros autores como "... la representación, el almacenamiento, la organización y el acceso a elementos de información" (1), la misma ha presentado un desarrollo ascendente motivado por el avance de Internet y la creciente necesidad de realizar búsquedas en la Web.

Actualmente la vinculación entre la educación y la Web mediante el empleo de las redes de comunicación (Internet), constituyen uno de los elementos de mayor capacidad de información a distancia. Esta tecnología se ha convertido en un medio eficaz para el proceso de enseñanza-aprendizaje, surgiendo y desarrollándose de esta manera el término *e-learning*.

Esta novedosa modalidad educativa se describe como un conjunto de tecnologías, aplicaciones y servicios orientados a facilitar la enseñanza y el aprendizaje a través de Internet/Intranet, que facilitan el acceso a la información y la comunicación con otros participantes (2). La misma genera un acercamiento al plano de la enseñanza de los distintos individuos, que tenían como limitante estar en contacto continuo con los tradicionales procesos de formación.

La aceptación del *e-learning* aumenta cada día más, reportando un incremento de los recursos digitales con fines educativos, dichos recursos se clasifican en recursos educativos abiertos y recursos educativos cerrados. De manera general, según la valoración del autor los recursos educativos son elementos digitales destinados para la enseñanza, el aprendizaje y la investigación, compuestos por materiales o técnicas empleadas para dar soporte al acceso de conocimiento según las necesidades educativas.

## Módulo para la búsqueda y recuperación de información en RHODA v3.0

Con el objetivo de establecer compatibilidad y reutilización entre las herramientas que gestionan dichos recursos, específicamente los OA; se realiza una estandarización de los contenidos educativos, surgiendo así estándares con diversas funciones: de catalogación (*Learning Object Metadata (LOM)*, *Dublin Core*, *IMS-Learning Resource Meta-data Specification (IMS-MD)*), de empaquetamiento (*IMS-Content Packaging (IMS-CP)*) y de secuenciación (*SCORM Sequencing and Navigation*). Además se encuentran las arquitecturas que permiten la transferencia de datos entre los sistemas, ejemplo: *Simple Query Interface (SQI)*, *Open Archives Initiative Protocol Metadata Harvesting (OAI-PMH)*, entre otros (3), las cuales definen una serie de funcionalidades que deben poseer los repositorios, destacando la búsqueda y recuperación de los recursos educativos.

Para lograr un amplio acceso y reutilización de los recursos es necesario almacenarlos en contenedores conocidos como repositorios, ya sea de recursos educativos o de OA. Estos repositorios según los autores Guzmán y Peñalvo se pueden clasificar en dos tipos, los que almacenan recursos con sus metadatos y los que solo almacenan metadatos (4).

En un repositorio los recursos educativos están indexados a través de sus metadatos, el uso de los mismos permite facilitar el proceso de búsqueda, los hace accesibles y recuperables propiciando una gestión automatizada del objeto. Aunque esta es una forma bastante aplicable, existe otra que se centra en el objeto de información del OA, la cual los contempla como: elementos digitales que conforman su contenido (5), permitiendo la indexación de estos para la recuperación de información. Este proceso se refiere a la acción de registrar ordenadamente información para elaborar un índice, con la finalidad de obtener resultados de una forma rápida y relevante en el momento de realizar la búsqueda.

Actualmente existen librerías para la realizar el proceso de indexación y búsqueda, que brindan facilidades para el desarrollador a la hora de ejecutarlo. Además poseen algoritmos y modelos de recuperación ya probados y de gran eficiencia. Entre las librerías más usadas se encuentran Lemur, Xapian, Terrier, Lucene, entre otras. Estas permiten la indexación y búsqueda de archivos de diversos formatos, tales como *Plain Text*<sup>1</sup>(TXT), *Portable Document Format*<sup>2</sup>(PDF), *Microsoft Word Document*<sup>3</sup>(DOC) y *HyperText Markup Language*(HMTL) (5).

---

<sup>1</sup> TXT (por sus siglas en inglés Plain text) es un archivo informático compuesto únicamente por texto sin formato, sólo caracteres.

## Módulo para la búsqueda y recuperación de información en RHODA v3.0

En el campo de la recuperación de la información, Cuba ha ido evolucionando de forma satisfactoria, logrando la incorporación de nuevos métodos y herramientas de apoyo a la enseñanza, sumados a los procedimientos educativos tradicionales en el ámbito de la Educación Superior. Varias instituciones han comenzado a crear sus propios repositorios, algunos ejemplos que evidencian este proceso son: el Repositorio de la Facultad de Matemática y Computación basado en E-Prints<sup>4</sup>, el Repositorio de Recursos Educativos de la Universidad Virtual de la Salud (RREUVS) de la red telemática Infomed; el Repositorio de la Universidad Agraria de la Habana (RUNAH) y el Repositorio de Recursos Digitales de la Universidad de la Habana.

La Universidad de las Ciencias Informáticas (UCI), como institución de avanzada en el uso de las TIC en su proceso docente-educativo ha incorporado los OA en función de elevar la calidad de los recursos educativos que se producen. Dicha universidad en su organización productiva posee varios proyectos de desarrollo de productos y servicios informáticos, entre los cuales se encuentra el Repositorio de Recursos Educativos RHODA, cuyo objetivo fundamental radica en almacenar y gestionar los recursos educativos mediante la utilización de estándares y cuenta con una amplia gama de funcionalidades que hacen del repositorio una eficaz herramienta web para la gestión de OA.

Para la recuperación de información asociada a los recursos educativos, RHODA en su versión 2.2 cuenta con un módulo para la realización de búsquedas. Este módulo fue desarrollado en el marco de trabajo Symfony 1.3.8 y de conjunto con la utilización de la librería Lucene, permite la ejecución de diferentes tipos de búsquedas tales como: general, avanzada, por metadatos, metadatos recolectados y federadas.

En la actualidad Symfony 1.3.8 no presenta soporte técnico, lo que limita las opciones de mantenimiento para esta versión del *framework*. De igual modo, debido a que otros productos del centro FORTES manejan algunos recursos que no son comprendidos como OA, se hace necesario que el repositorio RHODA en su versión 3.0 sea capaz de gestionarlos y almacenarlos, por lo que se requiere que el repositorio generalice su concepción inicial como repositorio de objetos de aprendizaje para convertirse en un Repositorio de Recursos Educativos. Actualmente dicho proyecto se encuentra en proceso de actualización de las tecnologías definidas, por tal motivo, se hizo necesario realizar una nueva concepción

---

<sup>2</sup> PDF (por sus siglas en inglés portable document format) es un formato de almacenamiento de documentos digitales independiente de plataformas de software o hardware.

<sup>3</sup> DOC(por sus siglas en inglés Microsoft Word Document) es un software destinado al procesamiento de textos

<sup>4</sup> E-Prints: software para el desarrollo y gestión de repositorios digitales.

## Módulo para la búsqueda y recuperación de información en RHODA v3.0

del módulo de búsqueda y recuperación de información, por lo que no solo se tendrá en cuenta el manejo de OA, sino también todos aquellos recursos educativos que se encuentren almacenados en el repositorio, eliminando de este modo los problemas de incompatibilidad existentes con el nuevo marco de trabajo(Xalix), así como las limitantes relacionadas con los recursos antes mencionados.

Por lo anteriormente descrito se plantea el siguiente **problema a resolver**: ¿Cómo facilitar el proceso de búsqueda y recuperación de la información en el repositorio RHODA 3.0?

Como **objeto de estudio**: Los procesos de búsqueda y recuperación en repositorios digitales.

Se define como **objetivo general**: Desarrollar un módulo que facilite la búsqueda y recuperación de los recursos educativos en RHODA 3.0.

Como **campo de acción** queda definido por consiguiente: el proceso de búsqueda y recuperación de información en el repositorio RHODA 3.0.

En correspondencia con el problema a resolver en el presente trabajo y como respuesta al mismo se proyecta la siguiente **idea a defender**: El desarrollo del módulo facilitará el proceso de búsqueda y recuperación de los recursos educativos en el repositorio RHODA 3.0.

De acuerdo al objetivo general se desglosan los siguientes **objetivos específicos**:

- Elaborar el marco teórico de la investigación.
- Describir la propuesta de solución a través del análisis y diseño de las funcionalidades definidas.
- Implementar el módulo para la realización de búsquedas y recuperación de información.
- Aplicar pruebas funcionales al módulo desarrollado.

Para dar cumplimiento a los objetivos propuestos se proponen las siguientes **tareas de investigación**:

- Revisión bibliográfica relacionada con el objeto de estudio.
- Análisis del proceso de búsqueda en los diferentes sistemas similares.
- Selección de las herramientas y tecnologías a utilizar.
- Descripción a partir de los diagramas del análisis y el diseño las funcionalidades del módulo.
- Selección de los métodos y técnica de pruebas a utilizar para garantizar el correcto funcionamiento del módulo a implementar.

## Módulo para la búsqueda y recuperación de información en RHODA v3.0

- Validación del sistema a través de la ejecución de pruebas.
- Corrección de los errores detectados durante la ejecución de las pruebas.

Se recurrió a la utilización de métodos científicos de investigación para cumplir con las metas que se persiguen en el trabajo y brindarle un carácter objetivo y científico al mismo:

Los **métodos teóricos** utilizados son:

- **Analítico-Sintético:** Permite de manera profunda la recopilación y estudio de temas relacionados con la búsqueda y recuperación de información, permitió generar una propuesta de análisis y diseño adecuada a la situación planteada y las tecnologías estudiadas.
- **Histórico-Lógico:** Para el estudio del desarrollo y evolución de sistemas similares existentes que posibiliten la búsqueda y recuperación de los recursos educativos.

Entre los **métodos empíricos** usados se encuentran:

- **Entrevistas:** Permite obtener información acerca del sistema de búsqueda y recuperación presente en la versión 2.2 de RHODA , mediante varias visitas a los usuarios y al personal que intervienen directa o indirectamente con dicho proyecto, recopilando la información necesaria de los requisitos funcionales del sistema a desarrollar.

### Estructura del Documento

La presente investigación se compone de 3 capítulos. Cada uno de ellos aborda diferentes temáticas las cuales están estructurados de la siguiente manera:

**Capítulo I.** Describe el marco teórico de la investigación. Se realiza un análisis de los elementos esenciales de los procesos de búsqueda en los repositorios, que constituyen el fundamento teórico del problema a resolver. Se hace referencia a los sistemas de búsquedas y recuperación de información, las técnicas que emplean, cuáles son sus principales funciones y la terminología correspondiente.

**Capítulo II.** Se realiza el análisis y diseño de la aplicación. Se exponen además, los artefactos generados durante la realización de los flujos de trabajo atendiendo la metodología utilizada.

*Módulo para la búsqueda y recuperación de información en RHODA  v3.0*

**Capítulo III.** En este capítulo se explica cómo se construyó la herramienta, así como su configuración. Se describen, además, las pruebas realizadas con el objetivo de verificar el correcto funcionamiento de la solución propuesta.

## **Capítulo 1: Fundamentación teórica**

### **Introducción**

Los Repositorios de Recursos Educativos son vistos como potenciadores de innovaciones pedagógicas, introduciendo nuevas alternativas de enseñanza y aprendizaje, estos facilitan la reutilización de recursos educativos, permitiendo la automatización completa de los procesos de búsqueda y recuperación de los mismos. Para la reutilización de estos recursos en distintas plataformas y escenarios han surgido estándares y especificaciones, los cuales a través de la catalogación, el almacenamiento y la interoperabilidad posibilitan erradicar de manera eficiente los problemas existentes de incompatibilidad en diferentes repositorios.

### **1.1 Recuperación de información**

Un sistema de recuperación de información (SRI) permite el procesamiento de una colección de documentos buscando entre estos aquellos que contengan el término asociado a la consulta del usuario (6). Este se encarga tanto de la recuperación de documentos como de su almacenamiento y organización, al igual como ocurre en los sistemas de gestión de bases de datos (SGBD). Hay que tener en cuenta que en los SGBD, los objetos que se tratan son estructuras de datos conocidas, sin embargo en los SRI la unidad u objeto de tratamiento es básicamente un documento de texto estructurado o no estructurados. En un SRI, un dato de consulta puede ser el documento completo, una parte del documento (puede ser una estructura del documento si este está estructurado o simplemente un párrafo del documento), o un metadato de la cabecera del documento, si este posee una cabecera estructurada (7). Si posee una estructura, se puede consultar en ciertos campos que representan las distintas secciones del documento. En cualquiera de estos campos se puede tener un tipo de datos, no previamente especificados y en el caso de datos de tipo texto pueden ser de longitud variable.

Un sistema de recuperación de datos simplemente comprueba si un dato existe en un fichero, mientras en los SRI se seleccionan aquellos documentos que coinciden parcialmente (búsqueda aproximada) o totalmente (búsqueda exacta) con los términos de la consulta del usuario, creándose una lista de documentos que en el caso de la búsqueda aproximada se puede ordenar según un índice de relevancia, es decir, aquellos que satisfacen la necesidad de información de un usuario determinado. Uno de los problemas más importantes en los SRI es cómo discernir si los documentos tratados son relevantes o no (8).

En la actualidad la recuperación de información ha desarrollado estrategias y modelos para mejorar y aumentar las funcionalidades de los SRI. Estas abarcan tópicos como filtrados y ruteos, clasificación, agrupamiento, sumarización y modelos de recuperación. Estos últimos emplean estructuras físicas y técnicas que dan soporte al proceso de acceso a la información de manera eficiente.

## 1.2 Modelos de recuperación de información

El término modelo tiene un significado conocido dentro del campo de la recuperación de información, mediante él se obtienen las representaciones de los documentos y las consultas, la estrategia para evaluar la relevancia de un documento respecto a una consulta y los métodos para establecer la importancia (orden) de los documentos de salida (8). De forma general, se puede considerar un modelo de recuperación como un método para representar tanto documentos como consultas en los SRI y comparar la similitud de esas representaciones.

Los SRI clásicos utilizan técnicas de búsqueda booleanas y de reconocimiento de patrones. Estas técnicas llamadas de búsqueda exacta son técnicas muy restrictivas, ya que no tienen en cuenta las ambigüedades que aparecen en el lenguaje natural y generalmente los usuarios tienen problemas en la construcción de las consultas.

Debido a los problemas que plantean los sistemas de búsqueda exacta se han desarrollado técnicas de búsqueda basadas en la información estadística, las denominadas técnicas de búsqueda aproximada. El sistema de búsqueda aproximada no hace un emparejamiento exacto y permite que el usuario especifique la importancia de cada uno de los términos. La realización de las consultas resulta más compleja en estos sistemas. A partir de lo expresado anteriormente se presentan diferentes clasificaciones de los modelos de recuperación de información.

### 1.2.1 Sistema de recuperación de información de búsqueda exacta

En los SRI de búsqueda exacta se utiliza una correspondencia exacta entre los términos de las consultas y de los documentos. Cuando se realiza correspondencia entre una consulta y un documento, el sistema le asigna un valor de relevancia de '1' al documento, mientras que si no se realiza el emparejamiento tiene relevancia '0'. Así, este le devuelve al usuario una lista con todos los documentos con relevancia '1' (7). En este sistema el primer documento de la lista no tiene por qué ser más interesante para el usuario, simplemente es el primero que ha encontrado relacionado con la consulta. Es decir, todos los documentos de la lista tienen la misma

relevancia, no se tiene en cuenta la frecuencia de aparición, ni el orden o importancia de los términos de la consulta.

Destacan dos modelos de sistemas de recuperación de información por búsqueda exacta:

- **Búsqueda de patrones.**

Los SRI por búsqueda de patrones utilizan técnicas de reconocimiento de patrones. En estos la consulta puede ser una colección de palabras, cadenas con comodines o expresiones regulares. El sistema intenta buscar los documentos que contengan el patrón de consulta. Este sistema de búsqueda no requiere índices, puede utilizar los documentos de la colección directamente. No suele ser muy útil en colecciones grandes por ser muy lento, pero es muy útil en colecciones de documentos que se modifican frecuentemente (9).

- **Indexación booleana.**

Se trata de uno de los modelos de recuperación de información más simples que se conocen. Se fundamenta en el álgebra de Boole y en la teoría de conjuntos. Este modelo crea una expresión booleana para formalizar la consulta y utiliza los operadores booleanos *AND*, *OR* y *NOT*. Dependiendo de los operadores booleanos que unan las palabras a buscar, se recuperarán unos documentos u otros (10).

### 1.2.2 Búsqueda aproximada

Los estudios sobre el modelo booleano dieron lugar a que estos se ampliaran para no desestimar un documento porque en él no aparecieran todos los términos de la consulta. De esta manera nacieron los modelos de búsqueda aproximada, entre los cuales hay que destacar el modelo vectorial conocido por ser uno de los más utilizados en los SRI.

- **Modelo vectorial**

También conocido como modelo de espacio vectorial, está basado en el modelo booleano, pero mejorado, de manera que se asigna a cada término de la consulta un peso que puede ser cualquier valor positivo (binario, entero o real). Dentro de este modelo los documentos son representados utilizando un vector en el que se recogen las relaciones existentes entre el documento y sus características. La consulta también se representa como un vector por lo que este modelo resulta perfecto para realizar la comparación entre documentos y consultas (6).

Para obtener las características que ayudan a la formación del vector, se utilizan las ocurrencias encontradas de algunas palabras significativas dentro del texto. Con estos datos se realiza la representación vectorial que será usada en las consultas para recuperar la información de determinados recursos educativos.

### 1.3 Repositorio de recursos educativos

En la gestión de contenidos en los entornos *e-learning* y la utilización de recursos educativos intervienen varios procesos, tales como la producción y el control de la calidad de los recursos, la carga en los LMS <sup>5</sup> y el almacenamiento en los repositorios digitales. Estos últimos actualmente se han convertido en parte de los servicios que ofrecen organizaciones e instituciones educacionales. Además de almacenar, los repositorios digitales permiten compartir a través de Internet materiales educativos, como vídeos, animaciones, imágenes, documentos y libros digitales (8). En el mundo digital, estos se clasifican en dos grandes grupos: Institucionales y Temáticos:

Los repositorios Institucionales, almacenan toda la información generada en una institución. Las universidades generan un elevado número de información digital y necesitan garantizar su persistencia, es por ello que han adoptado el uso de estas herramientas, con el fin de que profesores y estudiantes, puedan almacenar, localizar y recuperar artículos de diversos temas. Los repositorios Temáticos, almacenan datos de uno o varios temas específicos, ya sea de una rama de la ciencia o una disciplina (4).

El análisis de los tipos de repositorios digitales existentes le permite al autor seleccionar los repositorios institucionales como referencia para su investigación, pues a partir de ellos es posible almacenar y divulgar recursos educativos que se depositen.

Los repositorios de recursos educativos son una distinción especial de los repositorios digitales, y se distinguen por las características de los recursos que almacenan. Diferentes autores se refieren a estos como: un catálogo electrónico/digital que facilita las búsquedas en internet de objetos digitales para el aprendizaje (11).

---

<sup>5</sup> LMS (por sus siglas en inglés Learning Management System) son sistemas enfocados al área educativa que automatizan la administración.

Sin embargo, otros autores como Graeme , aseguran que son bases de datos, con búsquedas que alojan recursos digitales y/o metadatos que pueden ser utilizados para el aprendizaje mediado (12).

Las anteriores definiciones, comparten ideas en común: los repositorios de recursos educativos almacenan recursos digitales, facilitando su búsqueda y utilización en el aprendizaje, son sistemas especializados en el almacenamiento de recursos educativos, accesibles y operables por usuarios a través de Internet/Intranet, con funciones encaminadas a la clasificación, localización, reutilización, recuperación y mantenimiento de los mismos, que permiten compartir estos con otras herramientas de un entorno *e-learning*.

### 1.3.1 Sistemas similares

Como parte del estudio del arte se hizo necesario un análisis de la documentación y organizaciones existentes a nivel nacional e internacional. Para la selección de los repositorios se tuvo en cuenta que fueran de acceso abierto y que los contenidos fueran destinados a la enseñanza y el aprendizaje. En el análisis realizado a cada repositorio se buscó reflejar el cumplimiento de las características deseables. Además, de cada uno se describen: los destinatarios, el tipo de recurso educativo, el estándar de metadatos para el empaquetamiento y catalogación de los recursos que gestionan.

#### MERLOT

Es un recurso libre y abierto, diseñado principalmente para profesionales y estudiantes enmarcados en la enseñanza universitaria. Entre sus principales características se encuentran que es un repositorio centralizado que contiene solo los metadatos de los objetos y referencia la ubicación de estos en sitios remotos. Su funcionamiento es del tipo *standalone*<sup>6</sup>, y actúa como un portal de OA. Brinda servicios de búsquedas, las que de forma interna se ejecutan a través de la coincidencia con palabras clave, las que se pueden combinar con criterios como las categorías. Posee un formulario para la ejecución de búsquedas avanzadas que permiten diseñar complejos criterios de búsquedas y así satisfacer las necesidades de los usuarios sobre un contenido que se necesite y esté creado (11).

---

<sup>6</sup> Standalone: Palabra correspondiente al léxico informático para denominar a sitios con funcionamiento independiente.

Además permite que se ejecuten funciones de importar y exportar recursos digitales, específicamente comprendidos como OA. Los mismos son exportados según el formato técnico con el que se han creado, generalmente SCORM<sup>7</sup> 1.2, facilitando la creación de un sistema de metadatos para su posterior localización y catalogación.

### CAREO

Es un repositorio centralizado e independiente que da acceso abierto a los OA remotos y locales a través de los metadatos de su colección. Este repositorio consta de dos componentes la aplicación web Careo y la iniciativa Aloha<sup>8</sup>, que contempla las especificaciones IMS LOM<sup>9</sup> como base para su esquema de metadatos CanCore<sup>10</sup>.

La aplicación permite realizar búsquedas, utilizando los criterios introducidos por los usuarios y comparándolos con la información que aparece en los metadatos, este sistema de búsqueda permite ofrecer a los usuarios respuestas rápidas. Dicho repositorio puede establecer comunicación con otras aplicaciones similares utilizando las especificaciones IMS DRI<sup>11</sup> como base para la interoperabilidad. (12).

### SMETE Digital Library

Este repositorio pertenece a la *SMETE Open Federation*<sup>12</sup> y cuenta con materiales abiertos para profesores y estudiantes, principalmente orientado a la matemática, la ingeniería y la tecnología. SMETE es considerado un repositorio abierto, utiliza un perfil de metadatos basado en IEEE LOM. Para la navegación ofrece el acceso a los materiales a través de once áreas temáticas.

---

<sup>7</sup> SCORM (por sus siglas en inglés Sharable Content Object Reference Model) es un conjunto de estándares y especificaciones que permite crear objetos pedagógicos estructurados.

<sup>8</sup> Aloha (por sus siglas en inglés Advanced Learning Object Hub Application) es un servidor de metadatos que ofrece especial funcionalidad a dicho proyecto.

<sup>9</sup> LOM (por sus siglas en inglés Standard for Learning Object Metadata).

<sup>10</sup> CanCore especificación de metadatos propuesta por la iniciativa CanCore, una organización creada por el Multimedia Learning Group of Industry Canada y soportada por la universidad de Athabasca.

<sup>11</sup> IMS DRI (por sus siglas en inglés IMS Digital Repositories Interoperability). trata de abordar la interoperabilidad de los repositorios abarcando cada una de las funciones que estos deben cumplir un durante su uso: búsqueda, representación, recolección, envío, almacenaje, solicitud, entrega y aviso

<sup>12</sup> SMETE Open Federation es una biblioteca en línea y portal de servicios por parte de la Federación SMETE abierto para profesores y estudiantes.

Ofrece una búsqueda por palabra clave y avanzada. Realiza búsquedas federadas sobre los Repositorios MERLOT y NSDL<sup>13</sup> (13).

### **Agrega**

Es un proyecto de software que se ejecuta a cargo de Red.es<sup>14</sup>, el Ministerio de Educación y Ciencia, el Ministerio de Industria, Turismo y Comercio, y los Consejos de las Comunidades Autónomas en España, es un repositorio federado conformado por uno central y otros con carácter autonómico (dotado de autonomía legislativa y competencias ejecutivas, así como de la facultad de administrarse mediante sus propios representantes). El repositorio almacena tanto los OA como los metadatos de este, utilizando como esquema para la catalogación LOM-<sup>15</sup>ES y SCORM para el empaquetado.

El repositorio implementa la búsqueda de contenidos entre repositorios usando el protocolo SQI, mientras que los mecanismos de búsqueda de los nodos se basa en la indexación de un subconjunto de los metadatos que vienen descritos en el manifiesto de un OA, una vez indexados son procesados con la herramienta Lucene, que es la que se usa posteriormente para llevar a cabo las búsquedas (2).

### **Repositorio de Recursos Educativos de la Universidad Virtual de la Salud**

EL Repositorio de Recursos Educativos de la Universidad Virtual de la Salud (RREUVS) es un proyecto emprendido por la red telemática de la salud en Cuba: Infomed, desarrollado con el fin de crear, almacenar, catalogar y consultar recursos educativos publicados en la RREUVS. El repositorio cuenta además con un grupo importante de funcionalidades que se traducen en su correcto y eficiente funcionamiento.

Este repositorio permite a sus usuarios realizar un grupo importante de acciones, al igual que los repositorios antes mencionados registrarse es gratis y no se adquieren obligaciones por esto. Es del tipo centralizado, su funcionamiento de forma independiente y no almacena los recursos educativos, solo los metadatos y una referencia a la web del autor donde se encuentra

---

<sup>13</sup> NSDL (por sus siglas en inglés National Science Digital Library) proporcionando una manera de ver y descubrir cómo los conceptos científicos se relacionan entre sí.

<sup>14</sup> Red.es es una Entidad Pública Empresarial adscrita al Ministerio de Industria, Turismo y Comercio a través de la Secretaría de Estado de Telecomunicaciones.

<sup>15</sup> LOM-ES describe un perfil de aplicación que adapta al sistema educativo español el estándar LOM, este perfil ha sido realizado en el ámbito de la entidad de normalización AENOR.

físicamente. El *Dublin Core Metadata Elements Set* <sup>16</sup> es el modelo de metadatos usado, el cual provee una serie de elementos que permiten la descripción de los recursos, lo que permite una rápida localización de los mismos (14).

La ejecución de la búsqueda de los recursos se puede realizar del tipo simple o avanzada. El tipo avanzado de búsqueda permite construir complejos criterios de búsqueda, incluyendo que una vez definido el criterio permite guardarlo, de manera que si se necesita realizar una búsqueda semejante, no es necesario volver a construir el criterio, solo seleccionarlo de la lista de búsquedas guardadas (14).

### **Repositorio de Objetos de Aprendizaje RHODA**

RHODA en su versión 2.2 es un Repositorio de Objetos de Aprendizaje, cuyo objetivo fundamental radica en almacenar y gestionar los OA, facilitando la reutilización de los mismos y apoyar el proceso de capacitación existente en cualquier institución; es una aplicación Web modular y multiplataforma, que dispone de un lugar común, accesible a través de un navegador, donde los usuarios pueden almacenar, recuperar y consultar recursos destinados a lograr la enseñanza-aprendizaje. Utiliza el estándar SCORM1.2 y SCORM 2004 para gestionar los OA, y cuenta con una amplia gama de funcionalidades (3). Entre estas se encuentra la ejecución de búsquedas (general, avanzada, por metadatos, por metadatos recolectados y federada) basándose en los metadatos del estándar LOM, que hacen del repositorio una eficaz herramienta web para la gestión de OA.

Después del análisis de varios sistemas similares se pudo constatar según las tendencias actuales que los repositorios de recursos educativos están encaminados al manejo de diferentes tipos recursos, de los cuales permiten la reusabilidad, accesibilidad, durabilidad e interoperabilidad. Dichos repositorios aportan una serie de elementos significativos para la futura propuesta de solución tales como: el empleo de los diferentes estándares y especificaciones y se pudo analizar la forma en que se realizan los distintos tipos de búsquedas (simple, avanzada, por metadatos, entre otras) y la visualización de la información a través de las mismas.

---

<sup>16</sup> El estándar The Dublin Core Metadata Element Set de ISO recoge el conjunto de metadatos definidos por la iniciativa Dublin Core, un foro abierto dedicado al desarrollo de estándares de metadatos de propósito general enfocado principalmente a la localización y catalogación de recursos.

### 1.3.2 Estándares y especificaciones de contenidos e interoperabilidad

En la actualidad las personas utilizan términos como estándares y especificaciones indistintamente, cabe destacar que aunque no son términos equivalentes poseen aplicaciones semejantes (15). Según el diccionario de la Real Academia de la Lengua Española un estándar es “lo que sirve como tipo, norma, modelo, patrón o referencia” (16). De forma general los estándares se pueden clasificar de dos formas, los de jure u oficiales y los de facto (17).

Los estándares de jure son aquellos que han sido aprobados por alguna organización acreditada que certifica su validez (IEEE<sup>17</sup>/LTSC<sup>18</sup>, ISO<sup>19</sup>/IEC<sup>20</sup>), estos estándares en algunos casos son de cumplimiento obligatorio. Los de facto son aquellos que se usan por voluntad propia y poseen un elevado nivel de aceptación, aunque este no haya sido previamente certificado por alguna organización, este tipo de estándar inicialmente fue una especificación. Cuando un formato, tecnología o método propuesto no ha sido aprobado por alguna entidad certificadora, entonces este es una especificación (18). Cuando el uso de una especificación es extendido entonces se puede considerar como un estándar de facto.

Son varias las organizaciones que se han dedicado a la creación de especificaciones y estándares relacionados a las tecnologías aplicadas en los entornos *e-learning*. A continuación se describen los de mayor relevancia para la propuesta de solución.

#### IEEE Standard for Learning Object Metadata (LOM)

Organización: IEEE/LTSC<sup>21</sup>

LOM, es un modelo de datos que permite describir adecuadamente recursos digitales, normalmente usado en XML. Es el principal modelo de referencia para la creación de subconjuntos de metadatos como *Dublin Core*, *CanCore*, LOM -ES. Su principal objetivo es la creación de descripciones estructuras que permiten conocer el contenido del recurso sin necesidad de abrirlo. Esto facilita la búsqueda y catalogación de recursos educativos de forma rápida y eficiente (2).

---

<sup>17</sup> IEEE (por sus siglas en inglés Institute of Electrical and Electronics Engineers) es una asociación mundial de técnicos e ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas.

<sup>18</sup> LTSC (por sus siglas en inglés Learning Technology Standards Committee).

<sup>19</sup> ISO (por sus siglas en inglés International Organization for Standardization).

<sup>20</sup> IEC (por sus siglas en inglés, International Electrotechnical Commission)

<sup>21</sup> Disponible en <http://www.iee.org>.

## Simple Query Interface Specification for Learning Repositories (SQI)

Organización: CEN<sup>22</sup>

SQI ofrece una interfaz de programación de aplicaciones (API), que proporciona interoperabilidad entre los repositorios y aplicaciones de búsqueda y recuperación de recursos educativos. Dicha especificación soporta una gran variedad de tecnologías de búsqueda tanto asíncronas como síncronas, convirtiéndose en el mediador existente entre la fuente y el origen (19).

SQI define un conjunto de métodos que pueden ser agrupados por la funcionalidad a la que dan soporte (20):

- Manejo de Consultas: métodos que permiten la configuración de los parámetros de consultas: `setQueryLanguage()`, `setResultsFormat()`, `setMaxQueryResults()`, `setMaxDuration()`.
- Consultas Síncronas: los resultados de las consultas son directamente retornados por el método `synchronousQuery()`. Métodos adicionales permiten la elección de los números de resultados retornados por una consulta (`setResultsSetSize`) y además permiten conocer el número total de resultados de una consulta (`getTotalResultsCount`).
- Consultas Asíncronas: el API para este tipo de interacción define los siguientes métodos: `queryResultsListener` para ejecutar una consulta asíncrona, `setSourceLocation` para establecer la ubicación del cliente para que al terminar de realizar la consulta pueda entregar los resultados y `queryResultsListener`, este método forma parte de un servicio que implementa el cliente (que se convierte en servidor en este caso) para que le sean entregados los resultados.
- Manejo de Sesión: define las funcionalidades para crear una sesión (`createSession ()`), crear una sesión anónima o invitado (`queryResultsListener ()`) y para destruir las sesiones creadas (`queryResultsListener ()`).

Como se representa en las funciones anteriores, la especificación SQI atribuye gran importancia a los procesos de búsqueda y recuperación en los repositorios de recursos educativos. Este

---

<sup>22</sup> CEN, Comité Europeo de Normalización es la organización regional europea de estandarización. Su principal objetivo es fomentar la economía europea en el negocio global.

proceso constituye una funcionalidad básica y está involucrado en todos los demás. Está apoyado en los metadatos que caracterizan el contenido, aunque no está estrictamente sujeto a los mismos.

### **Open Archives Initiative Protocol for Metadata Harvesting (AOI-PMH)**

Organización: DLF<sup>23</sup>, CNI<sup>24</sup>, NSF<sup>25</sup>

OAI-PMH define un mecanismo para la recolección de registros que contienen los metadatos de los repositorios. Utiliza transacciones HTTP para emitir preguntas y obtener respuestas entre un servidor o archivo y un cliente o servicio recolector de metadatos. Las peticiones se emiten utilizando los métodos *GET* o *POST* del protocolo HTTP. Soporta múltiples formatos para expresar los metadatos, siempre y cuando la fuente y el destino compartan el mismo formato, no obstante el estándar Dublin Core es el que ofrece el mínimo de interoperabilidad y en el que más desarrollos existen (3).

A pesar de ser un estándar diseñado principalmente para bibliotecas, existen repositorios y redes que lo utilizan tales como: Agrega, red Ariadne, red Globe, entre otros.

El estudio realizado pone de manifiesto que se han desarrollado un número importante de especificaciones que estandarizan diferentes aspectos de un recurso educativo, tales como su empaquetamiento, sus metadatos, y protocolos para poder consultar y recuperar objetos de un repositorio, los cuales se tendrán en cuenta para la presente investigación.

### **1.4 Metodología de desarrollo de software, tecnologías y herramientas a utilizar**

Como parte del marco de trabajo definido en el centro FORTES, se establecieron un conjunto de tecnologías y herramientas con el objetivo de homogenizar el proceso de desarrollo de software. A continuación se hará referencia a las principales tecnologías y herramientas, así como la metodología de desarrollo de software a utilizar para dar cumplimiento al problema planteado en la presente investigación.

---

<sup>23</sup> Digital Library Federation

<sup>24</sup> Coalition for Networked Information

<sup>25</sup> National Science Foundation

### 1.4.1 Metodología de desarrollo de software

Las metodologías de desarrollo son un conjunto de pasos y procedimientos a seguir que permiten estructurar, planear y controlar el proceso de desarrollo de software. El uso de una metodología es un proceso que define quién debe hacer las cosas, qué debe hacerse, cómo y cuándo. Entre las más conocidas se pueden mencionar *eXtreme Programming* (XP), *Microsoft Solution Framework* (MSF) y *Rational Unified Process* (RUP) (21).

Debido a que la presente investigación constituye un módulo del proyecto RHODA, que utiliza como metodología de desarrollo RUP y con el objetivo de mantener la uniformidad en la documentación generada por parte de analistas y desarrolladores del proyecto, se decide utilizar la misma como guía de todo el proceso de desarrollo. Teniendo en cuenta que las características que esta posee, cabe destacar las siguientes: la adaptabilidad a una amplia gama de proyectos y organizaciones, presenta una guía de ayuda para llevar a cabo los proyectos de desarrollo que necesitan un seguimiento y control. Acorde a lo que establece la metodología se generan los siguientes artefactos: Modelo conceptual, Modelo de casos de usos del sistema, Modelo del análisis, Modelo del diseño, Modelo de datos y Modelo de implementación.

RUP es una metodología caracterizada por ser dirigida por casos de usos, centrada en la arquitectura, permite ganar control sobre el proyecto, lo cual es perfecto para manejar su complejidad y controlar su integridad, haciendo posible la reutilización a gran escala. Dentro de sus características describe un proceso iterativo incremental que se divide en 4 fases: Inicio, Elaboración, Construcción y Transición. Cada una de las etapas mencionadas se divide en iteraciones, de las cuales, cuando se realiza una, incluye nuevas funcionalidades al producto o mejora las existentes. Cada iteración trabaja sobre disciplinas diferentes, entre las cuales se encuentran Modelado de negocio, Ingeniería de requisitos, Análisis y diseño e Implementación y pruebas (22).

### 1.4.2 Lenguajes de programación

Un lenguaje de programación describe un conjunto de acciones consecutivas que un equipo de cómputo debe ejecutar. Estas acciones constituyen un conjunto de símbolos y reglas que permiten la comunicación con un computador. Las reglas permiten combinar dichos símbolos y se usan para escribir programas. Los lenguajes de programación al igual que un lenguaje

natural se componen de un léxico, una estructura (sintaxis) y un significado (semántica). El léxico es un conjunto de símbolos permitidos o vocabulario que forman parte de un lenguaje específico. Mientras que las sintaxis son las reglas que indican cómo construir frases correctas en un lenguaje, la semántica son las reglas que permiten determinar el significado de cualquier construcción del lenguaje (23), es decir, constituye el significado de las frases generadas por la sintaxis y el léxico.

### 1.4.3 Lenguajes de programación para el lado del servidor

#### Hypertext Preprocesor (PHP)

PHP es un lenguaje script que se ejecuta del lado del servidor en la Arquitectura Cliente – Servidor. Es utilizado para generar páginas web dinámicas. Su programación es segura y confiable ya que de ninguna manera en el navegador se accede al código fuente en PHP, sino solo a su resultado en HTML (24).

Es extremadamente simple para alguien con poca experiencia, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales, esto demuestra su facilidad de aprendizaje. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. Con PHP se puede procesar la información de formularios, generar páginas con contenidos dinámicos y aplicar técnicas de programación orientada a objetos.

Como se ha diseñado para su uso en la Web, incorpora gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la web. Puede generar imágenes GIF a un instante, establecer conexiones a otros servicios de red, enviar correos electrónicos, trabajar con cookies y generar documentos PDF<sup>26</sup> (24).

PHP en su versión 5.4 constituye un elemento fundamental para la realización de esta investigación, ya que mediante el empleo de las novedosas características que posee, como la integración a *framework* de desarrollo (Symfony2) logrando la incorporación de funciones anónimas y namespaces<sup>27</sup>, el empleo de nuevas etiquetas, las mejoras en los mensajes de

---

<sup>26</sup> PDF (por sus siglas en inglés portable document format) es un formato de almacenamiento de documentos digitales independiente de plataformas de software o hardware.

<sup>27</sup> Namespaces es un contenedor abstracto que agrupa de forma lógica varios símbolos e identificadores.

error de análisis y las advertencias de argumentos incompatibles y un mejor soporte a XML<sup>28</sup> (XPath y DOM), se facilita al desarrollador implementación de la propuesta de solución.

#### 1.4.4 Lenguajes de programación para el lado del cliente.

##### Lenguaje de Marcado de Hipertexto Extensible (HTML)

Es un lenguaje interpretado que utiliza etiquetas para realizar la creación de las páginas web en Internet. Dentro de sus características HTML separa los contenidos de las páginas, permite la creación de documentos y tablas de contenidos complejos (25). En su versión 5 brinda nuevas funcionalidades que facilitan el trabajo de los desarrolladores como el empleo de nuevos elementos para estructurar los contenidos de las páginas (<header>, <footer> y <nav>), los elementos de los formularios (calendarios, fechas y hora), atributos y tipo de entradas de validación automática las cuales fueron utilizadas para la elaboración de la siguiente propuesta de solución. Además, permite la integración de varios lenguajes así como la utilización de diferentes colores, tamaños y fuentes de letras que se pueden incorporar a las páginas.

##### JavaScript

Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript<sup>29</sup>, se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. (26) .Actualmente las implementaciones de JS de los navegadores, implementan la versión 5 del estándar ECMAScript cuyas últimas mejoras han sido (27):

- Rediseño de los atributos internos de las propiedades.
- Introducción de métodos estáticos que permiten:
  - Acceder a la información del prototipo.
  - Manipular las propiedades de un objeto.
  - Crear objetos de forma dinámica.
  - Impedir que un objeto sea modificado.
  - Cambios a las funciones:
  - Cambios en el objeto Date.

---

<sup>28</sup> XML (por sus siglas en inglés eXtensible Markup Language) es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.

<sup>29</sup> ECMAScript es una especificación de lenguaje de programación publicada por ECMA International

JavaScript brinda un conjunto de librerías que son de gran ayuda en el desarrollo de una aplicación web, entre las más conocidas se encuentra JQuery, librería de fácil manejo en el tratamiento de los objetos DOM.<sup>30</sup>

### **Hojas de Estilo en Cascada (CSS)**

Es un lenguaje que permite la definición de hojas de estilos diseñadas para el control de las interfaces de usuario definidas en los códigos HTML. Separa los contenidos de la presentación de las aplicaciones lo que permite el diseño de documentos bien definido y muy complejo. Permite definir los diferentes tamaños, colores y tipos de fuentes para las letras, así como la gestión de imágenes (28).

CSS en su versión 3.0, utilizada para la siguiente propuesta de solución, permite definir la vista y el estilo de cada uno de los elementos del módulo, lo que ofrece mayor facilidad para actualizar y dar soporte, menor codificación, color, tamaño y tipo de letra.

#### **1.4.5 Frameworks para el desarrollo de aplicaciones web**

El término *framework*, es una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación (29). El cual permite reducir considerablemente el tiempo de desarrollo de aplicaciones complejas y brindan estructuras muy robustas.

En los epígrafes siguientes se muestran los *frameworks* utilizados para la implementación de la propuesta de solución.

### **JQuery**

Es uno de los complementos esenciales para el desarrollo web, usado en diferentes sitios, lo que facilita el desarrollo de aplicaciones enriquecidas del lado del cliente, en Javascript, compatibles con todos los navegadores. Para el desarrollo de la propuesta de solución se hará uso de la versión 1.10.2. Dentro de sus características permite simplificar la manera de

---

<sup>30</sup> DOM (por sus siglas en inglés Document Object Model) es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML

interactuar con los documentos HTML, manipular el árbol DOM<sup>31</sup>, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX<sup>32</sup> a páginas web (30).

jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

### Bootstrap

Bootstrap es un *framework* para el desarrollo de aplicaciones de usuario, rápido y de fácil uso para la confección de sitios web. Se considera una librería de LESS<sup>33</sup> unida a otra de JavaScript. Soporta las versiones más recientes de los navegadores (Firefox, Internet Explorer, Opera, entre otros). Utiliza los estándares web de tal forma que con muy poco esfuerzo se pueden crear sitios web accesibles para personas que utilizan tecnologías de apoyo para navegar y HTML5 para hacer la web. Posee un gran número de elementos que HTML no incorpora, pero que se utilizan habitualmente como interfaz de usuario en aplicaciones web (31).

En su versión 3.0 Bootstrap incorpora nuevas mejoras mediante el empleo de varios recursos (estilos tipográficos, elementos de formulario, botones, tablas, barras de navegación, entre otros) facilitando la personalización en el módulo de búsqueda y recuperación a desarrollar.

### Symfony

Symfony es un poderoso *framework* orientado para la optimización del desarrollo de aplicaciones web basado en el patrón arquitectónico MVC<sup>34</sup>. El diseño del mismo separa las lógicas de negocio, de datos y la presentación; además proporciona herramientas para reducir el tiempo de desarrollo de aplicaciones complejas.

Para la propuesta de solución se utiliza la versión 2.3.7, la cual presenta las siguientes características y facilidades para el desarrollo.

---

<sup>31</sup> DOM (Modelo de Objetos del Documento). Permite ver el mismo documento de otra manera, describiendo el contenido del documento como un conjunto de objetos que un programa Javascript puede actuar sobre ellos.

<sup>32</sup> AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo Web para crear aplicaciones interactivas o RIA (Rich Internet Applications).

<sup>33</sup> LESS es una librería de estilos que incluye mixins (crean estilos semánticos para los diferentes elementos de la rejilla) para agregar sombras, transiciones, gradientes y bordes redondeados.

<sup>34</sup> MVC (Modelo Vista Controlador).

Permite el desarrollo de aplicaciones utilizando el lenguaje PHP. Su arquitectura interna está completamente desacoplada, lo que posibilita reemplazar o eliminar fácilmente aquellas partes que no se ajusten en el proyecto. Es compatible con los sistemas gestores de bases de datos MySQL, PostgreSQL, Oracle y SQL Server de Microsoft e independiente al sistema operativo del servidor, brindando así características esenciales para el desarrollo de aplicaciones web (32).

#### 1.4.6 Sistema gestor de base de datos

Un sistema gestor de base de datos (SGDB) es un software de propósito general que facilita los procesos de definir, diseñar, construir y manipular las bases de datos. Entre los diferentes SGDB cabe destacar PostgreSQL.

##### PostgreSQL

Es un sistema que permite la gestión de las bases de datos relacionales y orientados a objetos, desarrollado bajo una licencia de *Open Source* (código abierto). Es capaz de ajustar la manera más óptima de trabajar con las bases de datos según la cantidad de estaciones de trabajo y las prestaciones de hardware. Contiene un buen soporte para la implementación de triggers, cursores, índices, funciones y tipos de datos definidos por el usuario. Presenta una elevada integridad referencial para garantizar la validez de los datos, así como la implementación de un lenguaje procedural propio llamado PL/pgSQL (33). Para la presente investigación se utilizará PostgreSQL en su versión 9.1.

#### 1.4.7 ORM

Las bases de datos siguen una estructura relacional, PHP 5 y Symfony por el contrario son orientados a objetos. Por este motivo y de acuerdo al *framework* seleccionado, para acceder a la base de datos como si fuera orientada a objetos, es necesario una interfaz que traduzca la lógica de los objetos a la lógica relacional. Esta interfaz se denomina mapeo de objetos a bases de datos (ORM, de sus siglas en inglés Object-Relational Mapping) (34).

##### Doctrine

Doctrine 2.1 es un (ORM) para PHP 5.3.0 que proporciona persistencia transparente de objetos PHP. Una de las características clave de Doctrine es la opción de escribir las consultas de base de datos en un dialecto de Lenguaje de Consulta Estructurado (SQL) propio, orientado a

objetos conocido como Lenguaje de Consulta Doctrine (DQL), inspirado en Hibernate HQL<sup>35</sup>. Además, DQL difiere ligeramente de SQL en que abstrae considerablemente la asignación entre las filas de la base de datos y objetos, permitiendo a los desarrolladores escribir poderosas consultas de una manera sencilla y flexible. Doctrine 2.1 viene integrado a Symfony 2, este *framework* permite usarlo como una librería, para facilitar a los desarrolladores el acceso a los datos (35).

### 1.5 Librerías para realizar búsquedas

Actualmente existen diversas librerías de búsquedas, desarrolladas para enriquecer y facilitar el trabajo a los desarrolladores. Cada una de ellas utiliza criterios y elementos diferentes, que definen sus ventajas en su campo de acción.

A continuación se muestra una tabla comparativa donde se hace referencia de algunas librerías teniendo en cuenta que son las más empleadas en los procesos de búsquedas.

Tabla 1. Comparación de las librerías de búsqueda

| Características          | Lucene | Lemur | Xapian | Terrier |
|--------------------------|--------|-------|--------|---------|
| Multiplataforma          | SI     | SI    | NO     | NO      |
| PHP                      | SI     | NO    | SI     | NO      |
| SI                       | SI     | NO    | NO     | NO      |
| Formatos indexados       | SI     | SI    | SI     | SI      |
| Stemming <sup>36</sup>   | SI     | SI    | SI     | SI      |
| Búsqueda y actualización | SI     | NO    | NO     | NO      |

<sup>35</sup>Hibernate utiliza un lenguaje de consulta potente (HQL) que se parece a SQL. Sin embargo, comparado con SQL, HQL es completamente orientado a objetos y comprende nociones como herencia, polimorfismo y asociación.

<sup>36</sup> Stemming es la expansión automática de las palabras con las terminaciones habituales, como son los plurales y tiempos verbales, la coincidencia exacta, que indica que el término marcado con un signo de adición (+) delante debe aparecer obligatoriamente en el resultado de la búsqueda

|                               |                |                |                |                |
|-------------------------------|----------------|----------------|----------------|----------------|
| <b>Indexación incremental</b> | SI             | SI             | NO             | NO             |
| <b>Modelo</b>                 | Vectorial      | Probabilístico | Probabilístico | Probabilístico |
| <b>Licencia</b>               | Código abierto | Código abierto | Código abierto | Código abierto |

Las diferentes librerías presentadas poseen funcionalidades similares, sin embargo el autor considera que Lucene será la librería a incorporar para la ejecución de las funcionalidades del módulo de búsqueda, debido a las ventajas que provee. Lucene posee una estrategia de búsqueda ya definida y eficiente, puede detectar el modelo de datos utilizado en la aplicación, su principal ventaja es su flexibilidad, permite su utilización en cualquier sistema que lleve a cabo procesos de indización o búsqueda personalizados realizando un papel primordial en la recuperación de información.

## 1.6 Herramientas de software para el desarrollo de la solución

### Visual Paradigm

Visual Paradigm 8.0 es una herramienta de modelado, su característica más importante es su flexibilidad ya que es multiplataforma. Esta herramienta soporta todo el ciclo de vida del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Como lenguaje de modelado utiliza UML<sup>37</sup>, ofreciendo soluciones de software que permiten a las organizaciones desarrollar las aplicaciones con calidad más rápido, de forma satisfactoria. Incluye además, la capacidad para integrarse con distintos entornos de desarrollo y gestores de base datos. (36).

Visual Paradigm 8.0 es la herramienta empleada en la presente propuesta de solución para construir los artefactos ingenieriles asociado al desarrollo de la aplicación, debido a las facilidades que brinda para la modelación de sistemas y agiliza la creación de los diagramas definidos en la metodología de desarrollo RUP. La documentación que propone contribuye a lograr un mejor entendimiento del sistema por parte del equipo de desarrollo.

### Netbeans

---

<sup>37</sup>UML (por sus siglas en inglés **U**nified **M**odeling **L**anguage).

Netbeans es un Entorno de Desarrollo Integrado (IDE), diseñado para escribir, compilar, depurar y ejecutar aplicaciones. Está escrito para el lenguaje de programación java pero se puede extender para cualquier otro lenguaje de programación. También existe además un conjunto de herramientas y extensiones que permite usar el IDE con el objetivo de desarrollar en diversos lenguajes de programación. El IDE Netbeans es un proyecto libre y gratuito sin restricciones de uso alguno. (37).

Para la siguiente propuesta de solución se empleará la versión 7.4, la cual permite crear aplicaciones utilizando el lenguaje de programación PHP 5. Además integra una extensión con soporte para Symfony2 para facilitar el desarrollo de aplicaciones web.

### 1.6.1 Servidor web

Se define como un programa que procesa cualquier aplicación en el lado del servidor realizando conexiones bidireccionales y/o unidireccionales con el cliente y genera una respuesta en cualquier lenguaje o aplicación. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se utiliza el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI<sup>38</sup> (38).

#### Apache

Es un servidor web de código abierto para diferentes plataformas. Tiene como objetivo servir o suministrar páginas web (estáticas o dinámicas) a los clientes o navegadores que las solicitan (39).

Entre sus principales características se encuentran las siguientes (39):

- Mensajes de error altamente configurables.
- Es un servidor altamente configurable de diseño modular.
- Se ha concentrado en la escalabilidad, en la seguridad y en el rendimiento.
- Utiliza las tecnologías PHP y *servlets*<sup>39</sup> de Java.

#### Apache Solr

---

<sup>38</sup> OSI (por sus siglas en inglés, Open System Interconnection)

<sup>39</sup> Servelts es una clase en el lenguaje de programación java, utilizada para ampliar las capacidades de un servidor.

Es un servidor de búsquedas potente y sumamente optimizado, que se utiliza en grandes web que precisan realizar búsquedas de texto completo a gran velocidad y sobre numerosos registros. Se administra sobre un interfaz en HTML y responde a todos los estándares usados en la actualidad (XML y HTTP). Genera estadísticas de resultados para realizar monitoreo de tiempos y éxito en las búsquedas. Su configuración se basa en XML lo que permite un gran índice de escalabilidad (40).

Con anterioridad se definieron una serie de características de los servidores Apache 2.2.22 y Apache Solr 4.6.0 las cuales serán empleadas para el desarrollo de la propuesta de solución.

### **Conclusiones parciales**

Como resultado del estudio de las definiciones relacionadas, se ha logrado obtener una base conceptual que permite un mejor entendimiento de los temas a tratarse posteriormente asociados a la búsqueda y recuperación de información en los repositorios digitales. El análisis de las características principales de los repositorios, específicamente los que almacenan recursos educativos, proporcionaron una breve panorámica de los existentes en Cuba y en el mundo, identificando así las diferentes funcionalidades que estos poseen relacionadas con el módulo a desarrollar. Actualmente debido a la heterogeneidad de estas plataformas educativas es necesaria la existencia de estándares y especificaciones ampliamente aceptados que posibiliten la reutilización de los recursos y su interoperabilidad entre diferentes sistemas, destacándose la familia LOM como los más utilizados. Además se realizó el estudio detallado de las herramientas, tecnologías y lenguajes de programación existentes para el desarrollo de una herramienta web, siendo seleccionadas las opciones más factibles para dar cumplimiento al objetivo general.

## Capítulo 2: Propuesta de solución. Análisis y diseño

### Introducción

Actualmente existen disímiles herramientas, tecnologías y procesos para dar soluciones a problemas complejos que se presentan a diario y que favorece un mayor desempeño de grandes y medianas empresas. El proceso de construcción del software debe formar parte de un proceso definido, documentado y medido para poder ser gestionado. En el presente capítulo se elaboran diferentes artefactos de análisis y diseño propuestos por la metodología RUP, necesarios para dirigir y organizar todas las actividades involucradas en este flujo de trabajo.

### 2.1 Propuesta de solución

Este acápite describe las principales acciones implementadas para darle solución al problema planteado. Se muestran algunas de las interfaces correspondientes al módulo de búsqueda y recuperación de información. Se explica brevemente su funcionamiento y objetivos.

#### Búsqueda general

La búsqueda general representa la primera interfaz con que el usuario interactúa en el sistema. Contiene un campo de entrada y desde la misma se pueden acceder a las demás. Puede realizarla cualquier usuario sin necesidad de estar autenticado (invitado). Permite localizar los recursos educativos u objetos de aprendizajes que se encuentren publicados por una coincidencia parcial en el título, la descripción, palabras claves y en los contenidos de sus ficheros. En la misma se puede hacer uso de los operadores booleanos: *AND*, *NOT*, *OR* y cuando se desea obtener una frase exacta se debe encerrar entre comillas (""). En la siguiente figura es ilustrada la interfaz de la búsqueda general.



Ilustración 1. Búsqueda general

Este tipo de búsqueda se apoya en el objeto de tipo DOMDocument<sup>40</sup>, para acceder al fichero imsmanifest.xml, el cual contiene los datos del recurso educativo asociado a este. Además mediante el empleo del servidor de búsqueda Apache Solr posibilita el acceso a los contenidos indexados.

### Búsqueda avanzada

La búsqueda avanzada posee gran semejanza con la búsqueda general, la misma puede ser realizada por cualquier usuario sin necesidad de estar autenticado. A diferencia de la búsqueda anterior se suman otros criterios de búsqueda, se puede buscar por el autor, la categoría, la subcategoría, el contenido de los archivos de texto, las extensiones de dichos archivos, por fecha de creación y los resultados pueden ordenarse cronológicamente o alfabéticamente. En la siguiente figura es ilustrada la interfaz de la búsqueda avanzada:

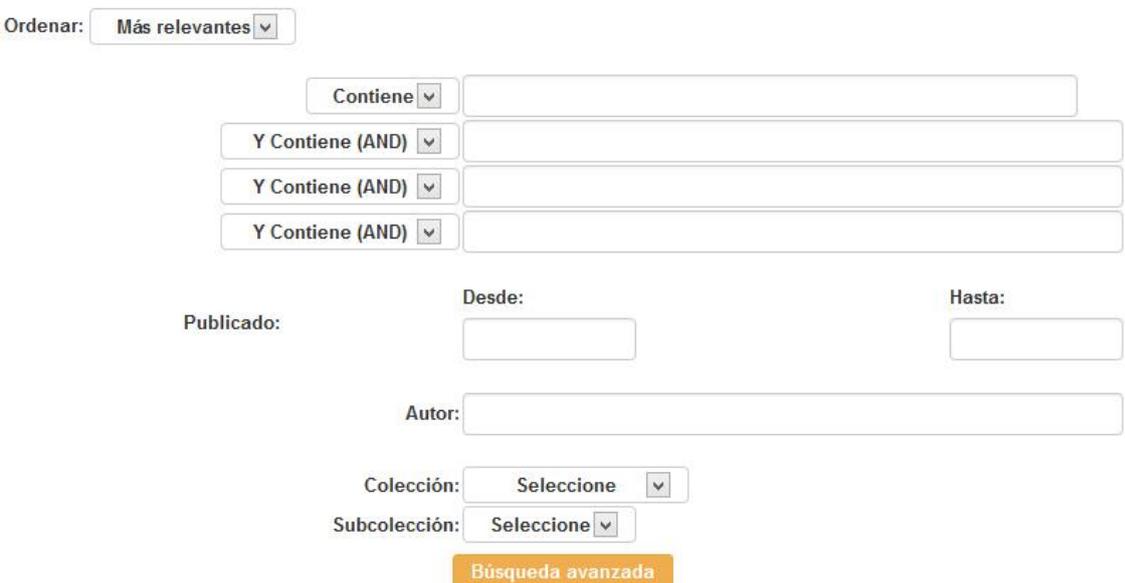


Ilustración 2. Búsqueda avanzada

En este tipo de búsqueda se hace un mayor uso de las facilidades que brinda el servidor Apache Solr, el cual utiliza la librería Lucene para la indexación y búsqueda de contenidos.

### Configuración de la búsqueda

<sup>40</sup> DOM (por sus siglas en inglés Document Object Model) es un conjunto de utilidades específicamente diseñadas para manipular documentos XML.

## Capítulo 2: Propuesta de solución. Análisis y diseño

### Módulo para la búsqueda y recuperación de información en RHODA v3.0

Mediante esta funcionalidad el usuario desarrolla un papel fundamental en el proceso de búsqueda. Es posible configurar varios elementos como son la paginación por defecto, el autocompletamiento de las búsquedas, el formato de los resultados por defecto y los criterios de búsqueda. En este proceso la configuración se almacena permanentemente, aunque el usuario cierre el navegador. Para modificar la configuración definida por defecto el usuario debe estar autenticado en el sistema. En la siguiente figura es ilustrada la interfaz de configuración de búsqueda:

#### Configuración de la búsqueda

**Autocompletar la búsqueda:**

Ofrecer predicciones de búsqueda en el cuadro de búsqueda.  
 No ofrecer predicciones de búsqueda en el cuadro de búsqueda.

**Número de resultados :** La opción predeterminada (10 resultados), ofrece los resultados con mayor rapidez.  
Mostrar  resultados por página.

**Campos de búsqueda :** Mostrar los siguientes campos en la búsqueda.

Buscar por intervalo de tiempo.  
 Buscar por autor.  
 Buscar por colecciones/subcolecciones.

**Tipos de los resultados :** Seleccione el área en la cual desea buscar.

Buscar en Item(Recursos) y Objetos de Aprendizaje(Título, Descripción, Palabra clave).  
 Buscar solo en Objetos de Aprendizaje(Título, Descripción, Palabra clave).  
 Buscar solo en Item(Recursos).

[Guardar preferencias](#)

Ilustración 3. Configuración de la búsqueda

### Historial de búsqueda

Esta funcionalidad solo pueden realizarla los usuarios que estén autenticados. Tiene como objetivo principal almacenar todas las búsquedas realizadas por los mismos, y testearlas de nuevo en el sistema. Esta funcionalidad posee gran importancia, ya que permite a los usuarios regresar a criterios anteriores de búsqueda que pudieron arrojar resultados de su interés y fueron olvidados. En la siguiente figura es ilustrada la interfaz del historial de búsqueda:

| Historial de búsqueda                         |            |        |
|---|------------|--------|
| Consulta                                      | Fecha      | Buscar |
| Principios                                    | 11/06/2014 | Buscar |
| "Repositorio de objetos de aprendizaje RHODA" | 11/06/2014 | Buscar |
| Tecnologías de la Información                 | 11/06/2014 | Buscar |

[Borrar Historial](#)

Ilustración 4. Historial de búsqueda

En la misma se hace uso de la sesión de usuario de Symfony2, para el almacenamiento del historial, muy semejante a la configuración de la búsqueda.

Una vez definidas algunas de las funcionalidades implementadas en el módulo de búsqueda y recuperación de los recursos educativos en el repositorio RHODA, se crean las bases para el diseño del modelado del sistema, generando así los artefactos determinados de acuerdo a la metodología seleccionada.

## 2.2 Modelo conceptual

Un Modelo conceptual constituye una especificación de los conceptos más importantes de la vida real que se utilizarán en el sistema, “una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés” (38). RUP define la creación de un Modelo conceptual como uno de los artefactos a generar durante el flujo de trabajo: Modelamiento del negocio.

El Modelo conceptual que a continuación se describe constituye una representación gráfica de los principales conceptos que forman parte del módulo para la búsqueda y recuperación de recursos educativos en RHODA.

### Definición de conceptos fundamentales:

**Invitado:** Persona que puede interactuar con el repositorio sin necesidad de estar autenticada.

**Usuario:** Persona registrada en el sistema.

**Recursos educativos:** Recurso compuesto por contenidos educativos que pueden ser cursos completos, materiales para cursos, materiales multimedia, objetos de aprendizaje y cualquier otro recurso con fin educativo con una alta pertinencia y certificación de calidad.

**Búsqueda:** Servicio que brinda el repositorio, este es utilizado por los usuarios que acceden al sistema, con el fin de localizar los recursos educativos.

**Historial de búsqueda:** Conjunto de consultas o búsquedas realizadas por un usuario.

**Configurar búsquedas:** Permite configurar un conjunto de opciones que brindan valor agregado al sistema.

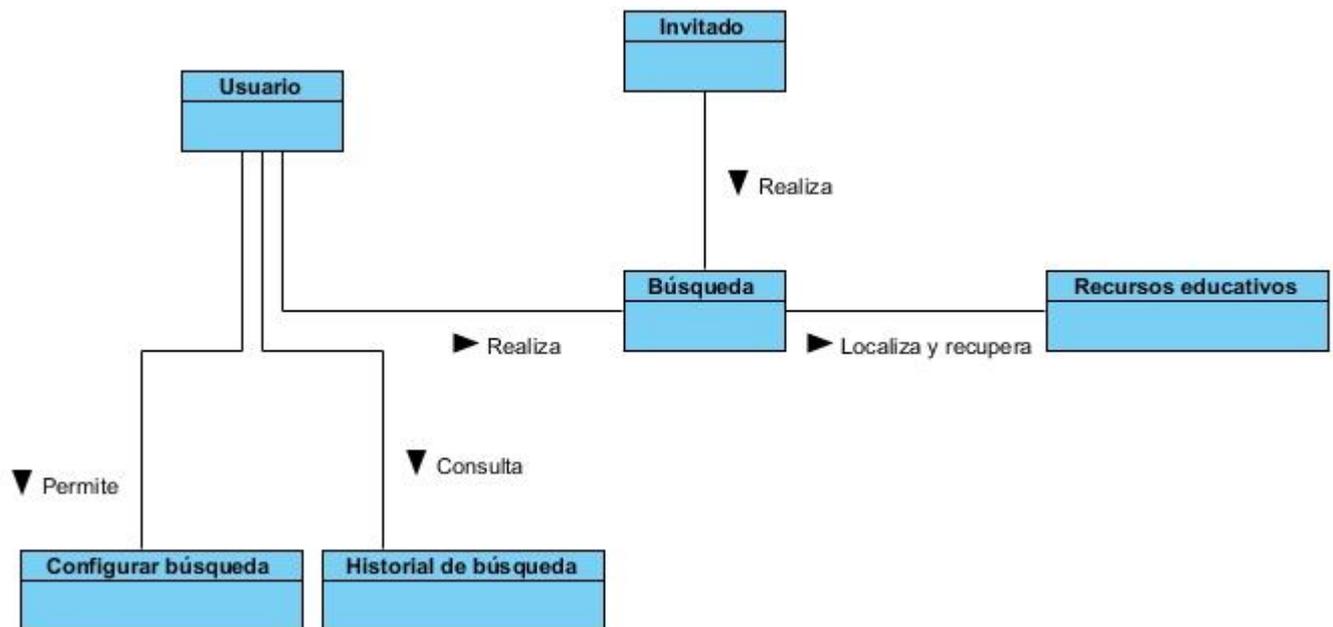


Ilustración 5. Modelo conceptual

## 2.3 Especificación de requisitos

Luego de analizar el dominio del problema es necesario definir qué debe hacer el sistema, para ello deben ser analizadas todas las ideas que los clientes, usuarios y miembros del equipo de proyecto tengan como candidatas a requisitos. Los requisitos se clasifican en funcionales (RF) y no funcionales (RNF).

### 2.3.1 Requisitos funcionales

Los requerimientos o requisitos funcionales no son más que las condiciones que el sistema debe cumplir, el punto de partida para identificar qué debe hacer el sistema. Los requerimientos funcionales deben ser comprendidos tanto por los desarrolladores como los usuarios. A continuación se relacionan los que debe cumplir la aplicación a desarrollar:

**RF-1. Realizar búsqueda general de recursos educativos:** Permite realizar búsquedas de recursos educativos introduciendo un parámetro.

**RF-2. Realizar búsqueda avanzada de recursos educativos:** Permite realizar búsquedas de recursos educativos introduciendo varios parámetros, de igual modo permite especificar un rango de fecha de publicación, el autor, así como la colección y subcolección a la que pertenece el recurso que se desea localizar.

**RF-2.1. Ordenar resultados de la búsqueda avanzada:** Permite ordenar los resultados de la búsqueda descendientemente, ya sea por los más recientes o los más relevantes.

**RF-3. Realizar búsqueda por los metadatos de los recursos educativos:** Permite realizar búsquedas por los metadatos de los recursos, tales como el título, la descripción y el autor.

**RF-4. Realizar búsqueda sobre los metadatos recolectados con el estándar OAI-PMH:** Permite realizar búsquedas a partir de metadatos recolectados en otros sistemas.

**RF-5. Realizar búsqueda federada mediante el estándar SSI:** Permite realizar búsquedas de recursos en otros sistemas.

**RF-6. Configurar búsquedas:** Permite al usuario la configuración de la búsqueda teniendo en cuenta los campos:

- Autocompletar la búsqueda, en este caso permite definir si se desea contar o no con predicciones.
- Definir la cantidad de resultados que se van a mostrar.
- Seleccionar los campos que se desean que aparezcan en la búsqueda avanzada.
- Seleccionar el área donde se quiere realizar la búsqueda.

**RF-7. Ver historial de búsqueda:** Permite consultar las búsquedas realizadas hasta el momento, mostrando el recurso consultado y la fecha en que se realizó dicha búsqueda.

**RF-7.1. Borrar historial de búsqueda:** Permite borrar las búsquedas realizadas hasta el momento.

### **2.3.2 Requisitos no funcionales:**

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable (22).

Existen múltiples categorías para clasificar a los requisitos no funcionales, siendo las siguientes, representativas de un conjunto de aspectos que se deben tener en cuenta, aunque no limitan a la definición de otros.

**Usabilidad:**

**RNF 1. Ambiente:** Se debe contar con el servidor de bases de datos relacional PostgreSQL y un servidor de aplicaciones, cada uno con más de 2 GB de memoria RAM y 160 GB de disco duro en el caso del servidor de bases de datos. El acceso a la aplicación se realizará desde varias estaciones clientes, las que deberán tener como mínimo 512 MB de memoria RAM y un navegador web (Mozilla Firefox v15 o superior). El tiempo de respuesta de la aplicación no debe exceder los 5 segundos para cada petición realizada.

**Confiabilidad:**

**RNF 2. Respuesta ante ocurrencia de una excepción:** Cuando ocurre una excepción el sistema debe mostrar un mensaje de error que indique el posible error en un formato entendible para el usuario, sin llegar a especificaciones técnicas que puedan constituir una brecha en la seguridad del sistema.

**Eficiencia:**

**RNF 3. Rendimiento:** La cantidad de datos que son transmitidos por segundos es de 500 KB.

**RNF 4. Capacidad:** El sistema consume por cada sesión de usuario alrededor de 30 MB.

**RNF 5. Estándar de codificación:** El estándar de codificación utilizado es el definido en el marco de trabajo Xalix.

**Restricciones de diseño:**

**RNF 6. Lenguaje de programación:** El lenguaje de programación utilizado es PHP 5.4.

**RNF 7. Entorno de Desarrollo Integrado:** El Entorno de Desarrollo Integrado utilizado es Netbeans 7.4

**RNF 8. Framework de desarrollo:** El *framework* de desarrollo utilizado es Symfony 2.3.7.

**RNF 9. Estilo arquitectónico:** El estilo arquitectónico utilizado es Modelo Vista Controlador.

**RNF 10. Biblioteca JavaScript:** La biblioteca JavaScript utilizada es jQuery v1.10.2.

**RNF 11. Framework CSS:** Para el diseño de las interfaces gráficas se empleó Bootstrap v 3.0.

**RNF 12. Servidor de búsqueda:** Para la realización de búsquedas se empleó Apache Solr 4.6.0.

#### Requisitos de licencia:

**RnF.13. Licencia:** El repositorio debe estar registrado bajo la Licencia Pública General (GPL) versión 3.0.

#### Estándares aplicables:

**RNF.14 Estándar para la realización de búsquedas SQI:** El repositorio permite la realización de búsquedas federadas en otros repositorios mediante el estándar *Simple Query Interface* (SQI).

**RNF.15 Estándar para la realización de búsquedas por metadatos recolectados OAI-PMH:** El repositorio es capaz de recolectar metadatos en otros repositorios y aplicaciones *e-learning* para posteriormente realizar búsquedas a partir de los mismos.

**RNF.16 *Learning Object Metadata* (LOM):** El repositorio utiliza la especificación LOM para la descripción de los recursos educativos incluyendo sus nueve categorías.

Una vez recopilados los requisitos, el ingeniero del software (analista) puede crear un conjunto de escenarios que identifiquen una línea de utilización para el sistema que va a ser construido. Los escenarios, algunas veces llamados Casos de uso, facilitan una descripción de como el sistema se usara (41).

## 2.4 Modelo de casos de uso del sistema

El Modelo de casos de uso posibilita a los usuarios, y a los desarrolladores llegar a un acuerdo sobre cómo utilizar el sistema. La mayoría de los sistemas poseen diferentes usuarios los cuales se representan mediante un actor. Estos utilizan el sistema al interactuar con los casos de uso (CU). Todos los actores y CU del sistema forman un Modelo de casos de uso. Un diagrama de casos de uso describe parte del Modelo de casos de uso y muestra un conjunto de CU y actores con una asociación entre cada par actor/caso de uso que interactúan (21).

A continuación se identifican los Casos de usos (CU) del sistema para la propuesta de solución:

**CU\_1:** Realizar búsqueda general de RE.

**CU\_2:** Realizar búsqueda avanzada de RE.

**CU\_3:** Realizar búsqueda por los metadatos de los RE.

**CU\_4:** Realizar búsqueda de RE por metadatos recolectados.

**CU\_5:** Realizar búsqueda federada de RE.

**CU\_6:** Configurar búsquedas.

**CU\_7:** Gestionar historial de búsquedas.

### Actores del sistema

Los actores del sistema representan personas u otros sistemas que interactúan con el sistema. Cada actor asume un conjunto coherente de papeles durante la interacción (21).

Tabla 2. Actores del sistema

| Actor    | Objetivos   |
|----------|---|
| Invitado | Persona que no posee un usuario en el sistema y que por lo tanto solo tiene acceso a las funcionalidades básicas del mismo.   |
| Usuario  | Persona registrada en el sistema que además de poder acceder a las funcionalidades del invitado, se le permite el acceso a su perfil, la mensajería del sistema y un área de trabajo para agrupar los recursos. |

### Diagrama de casos de uso

Un diagrama de CU del sistema representa gráficamente los procesos y su interacción con los actores.

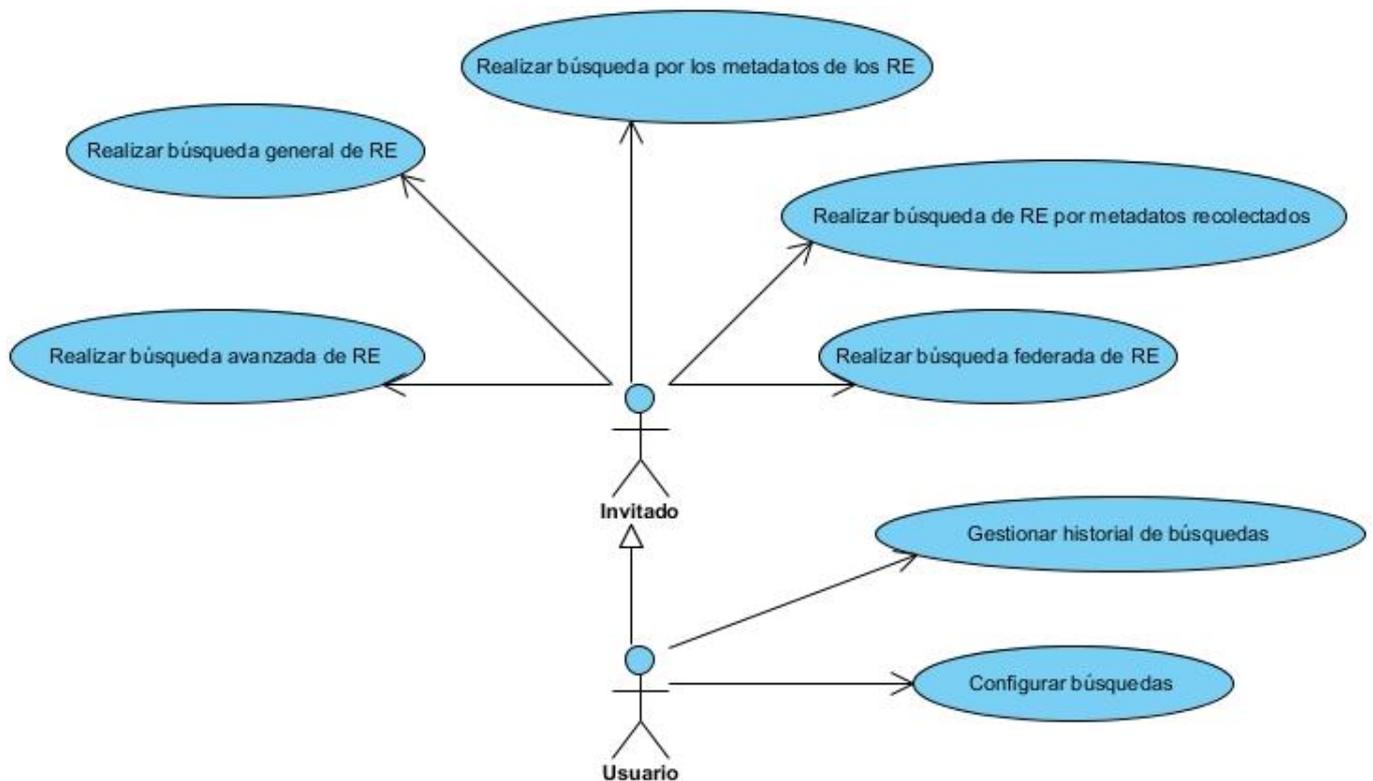


Ilustración 6. Diagrama de CU

### 2.4.1 Descripción de casos de uso

Las descripciones de los CU permiten comprender mejor el flujo de actividades que realiza un actor al hacer uso del sistema y las respuestas que emite el mismo. Estos constituyen una descripción absoluta de los requisitos que engloban (41).

En esta sección serán expuestas las descripciones de los CU Realizar búsqueda general de RE y Gestionar historial de búsqueda, los restantes se podrán localizar en el **Anexo I**.

Tabla 3 Descripción del CU búsqueda general de RE

|                        |   |
|------------------------|---|
| <b>Objetivo</b>        | Listar los Recursos Educativos que coinciden con un término en específico.  |
| <b>Actores</b>         | Invitado (inicia).  |
| <b>Resumen</b>         | El CU se inicia cuando el invitado selecciona el criterio de búsqueda y termina cuando se muestra el resultado de dicha búsqueda. |
| <b>Complejidad</b>     | Alta  |
| <b>Prioridad</b>       | Crítico   |
| <b>Precondiciones</b>  | Debe existir al menos un RE en el sistema.  |
| <b>Postcondiciones</b> | Se muestra el resultado de la búsqueda realizada.   |

| <b>Flujo de eventos</b>                              |   |  |
|--|---|--|
| <b>Flujo básico Realizar búsqueda general de RE.</b> |   |  |
|  | <b>Actor</b>  | <b>Sistema</b>   |
| 1  | Selecciona en el menú búsqueda la opción de la lista desplegable “ <i>Búsqueda general</i> ”.                     | 1.1 Muestra un campo de texto para que el actor introduzca el criterio de búsqueda.  |
| 2  | Introduce un término que identifique la información que desea buscar.<br>Hace clic en el botón “ <i>Buscar</i> ”. | 2.1 Valida el dato introducido.<br>2.2 Muestra la cantidad de coincidencias encontradas y el lugar donde realizó la búsqueda, así como el listado de los recursos educativos.<br>De cada recurso se muestran los siguientes datos:<br>-Texto<br>- Nombre del recurso: (Formato: alfabético y/o numérico).<br>- Autor: (Formato: alfabético y/o numérico).<br>-Palabras claves:(Formato: alfabético y/o numérico).<br>- Descripción (Formato: alfabético y/o numérico). |
| 3  |   | 3.1 Termina el caso de uso.  |
| <b>Flujos alternos</b>                               |   |  |
| <b>2.a “Campo vacío”</b>                             |   |  |
|  | <b>Actor</b>  | <b>Sistema</b>   |
| 2  |   | 2.1 Muestra el mensaje “ <i>Rellene este campo</i> ”.  |
|  |   | 2.2 Regresa al paso 2 del flujo básico.  |
| <b>2.b “No se encontraron coincidencias”</b>         |   |  |
|  | <b>Actor</b>  | <b>Sistema</b>   |
| 2  |   | 2.1 Muestra el mensaje “ <i>No se encontraron coincidencias</i> ”.   |
|  |   | 2.2 Regresa al paso 2 del flujo básico.  |

Tabla 4. Descripción del CU Gestionar historial de búsqueda

|   |  |  |
|---|--|--|
| <b>Objetivo</b>                                     | Listar los Recursos Educativos (RE) que están almacenados en el historial de búsqueda.                                       |  |
| <b>Actores</b>                                      | Usuario (inicia).  |  |
| <b>Resumen</b>                                      | El CU se inicia cuando el usuario selecciona el criterio de historial de búsqueda y termina cuando consulta dicho historial. |  |
| <b>Complejidad</b>                                  | Alta   |  |
| <b>Prioridad</b>                                    | Crítico  |  |
| <b>Precondiciones</b>                               | -----  |  |
| <b>Postcondiciones</b>                              | -----  |  |
| <b>Flujo de eventos</b>                             |  |  |
| <b>Flujo básico Gestionar historial de búsqueda</b> |  |  |
|   | <b>Actor</b>   | <b>Sistema</b>   |
| 1   | Selecciona en el menú búsqueda la opción de la lista desplegable “ <i>Historial de búsqueda</i> ”.                           | 1.1. Muestra el listado de las búsquedas realizadas, de cada una muestra el criterio de búsqueda, la fecha en que se realizó y la opción buscar (Ver Sección: “ <i>Buscar</i> ”).<br><br>Además permite la opción:<br>-Borrar historial: (Ver Sección: “ <i>Borrar historial</i> ”). |
| 2   |  | 2.1 Termina el caso de uso.  |
| <b>Flujos alternos</b>                              |  |  |
| <b>1.a “No existen búsquedas realizadas”</b>        |  |  |
|   | <b>Actor</b>   | <b>Sistema</b>   |
| 1   |  | 1.1 Muestra el mensaje “ <i>No existen búsquedas realizadas</i> ”.   |
|   |  | 1.2 Termina el caso de uso   |
| <b>Sección : “Buscar”</b>                           |  |  |
| <b>Flujo básico Buscar</b>                          |  |  |
|   | <b>Actor</b>   | <b>Sistema</b>   |
| 1.  | Selecciona la opción “ <i>Buscar</i> ”.  | El sistema muestra los resultados que se obtuvieron en la búsqueda realizada y permite realizar la misma consulta nuevamente.  |
| 2   |  | 2.1 Termina el caso de uso.  |

|  |  |   |
|--|--|---|
| <b>Sección : “Borrar historial”</b>    |  |   |
| <b>Flujo básico Borrar historial</b>   |  |   |
|  | <b>Actor</b>                                       | <b>Sistema</b>  |
| 1                                      | Hace clic en el botón: “ <i>Borrar historial</i> ” | 1.1 Muestra un mensaje de interrogación: “¿Está seguro que desea eliminar el <i>historial</i> ?”. Y permite: <ul style="list-style-type: none"> <li>• Eliminar</li> <li>• Cancelar</li> </ul> |
| 2                                      | Selecciona la opción: “ <i>Eliminar</i> ”.         | 2.1 Elimina el historial de búsqueda.<br>2.2 Retorna a la interfaz principal.<br>2.3 Termina el caso de uso.  |
| <b>Flujos alternos</b>                 |  |   |
| <b>2.a: “Cancelar acción eliminar”</b> |  |   |
|  | <b>Actor</b>                                       | <b>Sistema</b>  |
| 2                                      | Selecciona la opción: “ <i>Cancelar</i> ”.         | 2.1 Se mantiene la interfaz asociada al historial de búsqueda.  |
|  |  | 2.2 Termina el caso de uso.   |

## 2.5 Modelo del análisis

Una vez identificados los requisitos del sistema y agrupadas las funcionalidades en los CU, es posible obtener una visión más detallada del sistema si se especifica con anterioridad qué debe hacer el mismo, lo cual es posible representar a través del Modelo de análisis.

Los desarrolladores lo utilizan para comprender como debería ser diseñado e implementado el software. Constituye la primera aproximación al Modelo de diseño y está conformado por las clases del análisis (interfaz, control y entidad), las cuales encapsulan las diferentes funcionalidades que representan los CU (21).

### 2.5.1 Diagramas de clases del análisis

Al igual que otros artefactos mencionados con anterioridad, las clases del análisis se utilizan para denotar el intercambio de información y responsabilidades que se establecen entre ellas y las relaciones que tienen con los actores, es por eso que se hace necesario construir

## Capítulo 2: Propuesta de solución. Análisis y diseño

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

un diagrama que permita representar las relaciones que existen, para dar respuesta a esta necesidad se hace uso del Diagrama de clases del análisis.

A continuación se representan los Diagramas de clase del análisis (DCA) del CU Realizar búsqueda general de RE y Realizar búsqueda avanzada de RE, los restantes DCA se pueden encontrar en el **Anexo II**.

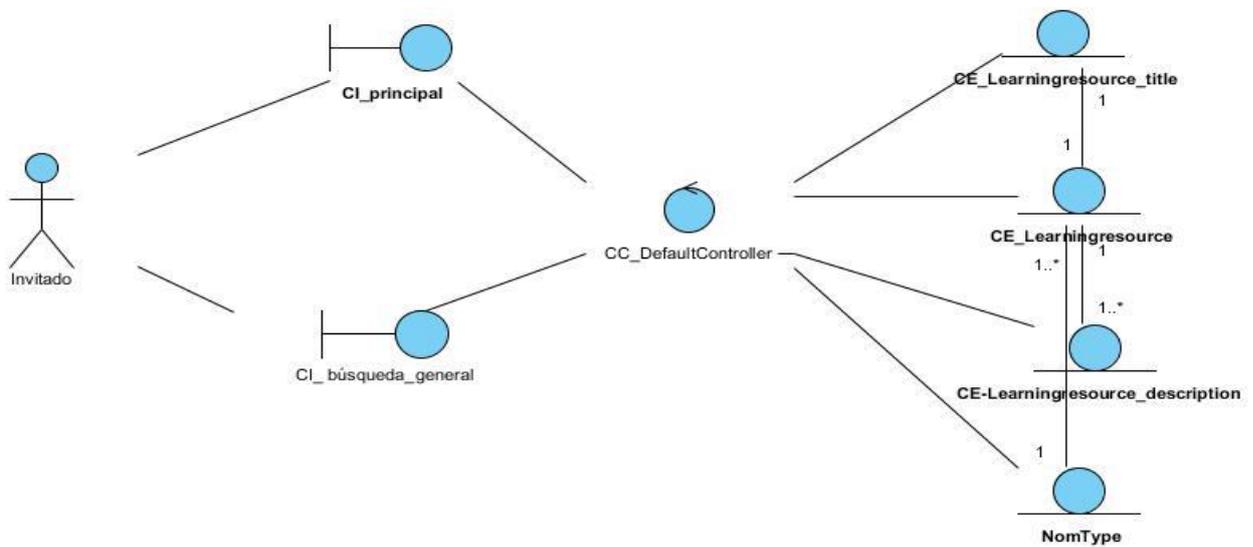


Ilustración 7. DCA Realizar búsqueda general de RE

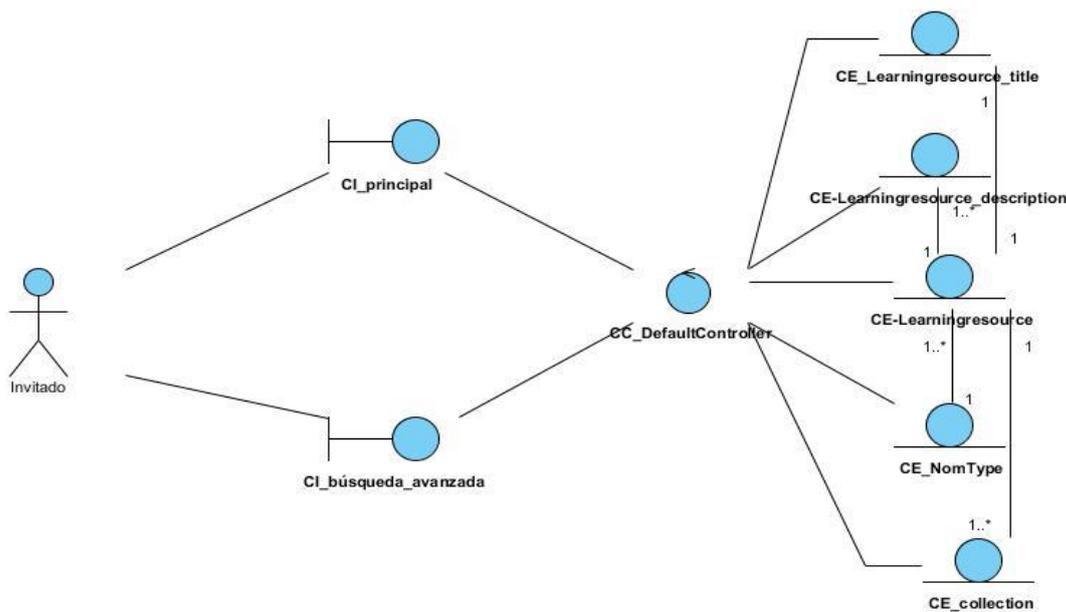


Ilustración 8. DCA Realizar búsqueda avanzada de RE

### 2.5.2 Diagramas de colaboración

Un diagrama de colaboración es un diagrama de interacción que modela los objetos y enlaces implicados en la implementación de una interacción, por lo general solo se representan los objetos de CU implicados en la colaboración. Este enfatiza la organización estructural de los objetos que envían y reciben mensajes, muestra las interacciones organizadas alrededor de instancias y de los enlaces entre ellas y son muy útiles cuando se desea mostrar el diseño detallado de procedimientos (42).

Las características antes enunciadas, fueron las que hicieron posible que en la presente investigación se le diera un peso fundamental a este tipo de diagrama de interacción.

A continuación se representan los Diagramas de colaboración (DC) del CU Realizar búsqueda general de RE y Configurar búsquedas, los restantes DC se pueden encontrar en el **Anexo III**.

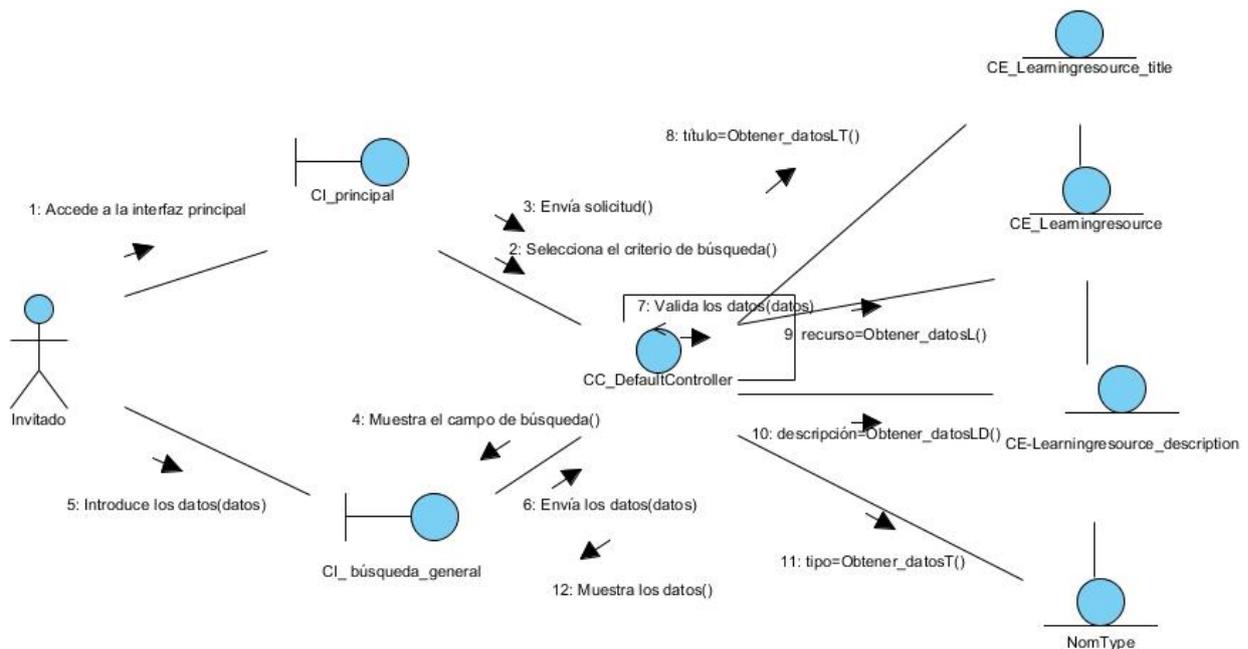


Ilustración 9. DC Realizar búsqueda general de RE

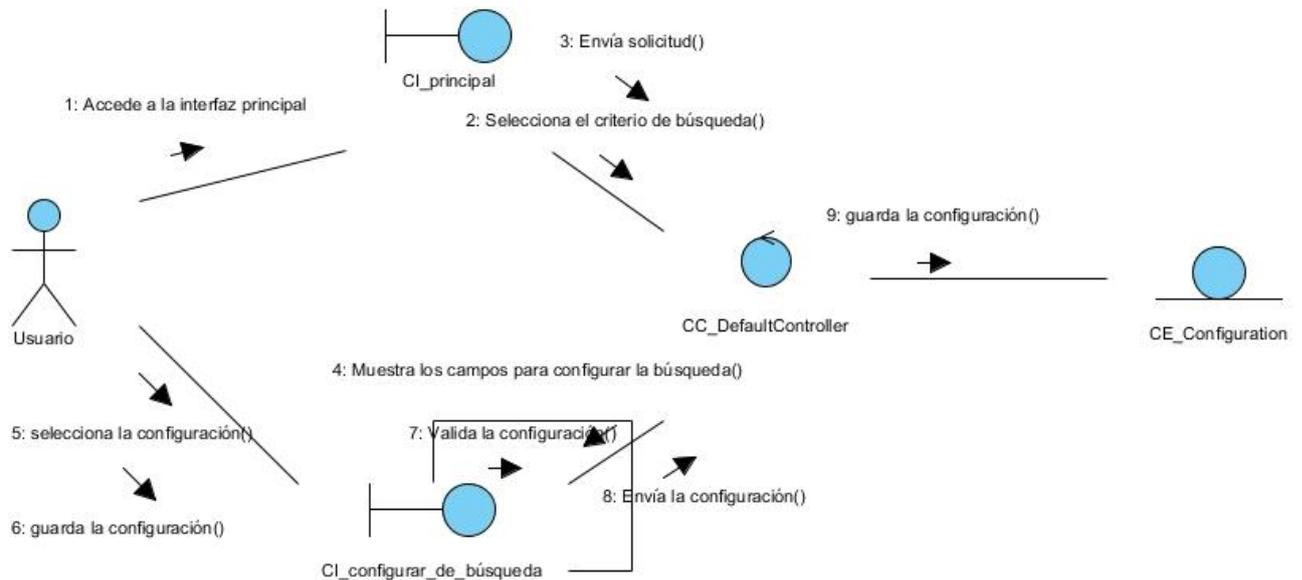


Ilustración 10. DC Configurar búsquedas

## 2.6 Patrones de diseño utilizados

La ventaja fundamental de la utilización de *frameworks* para el desarrollo de aplicaciones es que los mismos incluyen patrones de diseño en su arquitectura de forma nativa, entre ellos los patrones: Patrones Generales de Software para Asignar Responsabilidades (del inglés *General Responsibility Assignment Software Patterns* (GRASP)) y Grupo de 4 (del inglés *Gang of Four* (GoF)).

### 2.6.1 Patrones de diseño GRASP

Es un conjunto de patrones que representa los principios básicos de la asignación de responsabilidades. A continuación se muestran los patrones GRASP empleados para el diseño del sistema (43).

**Experto:** Dentro de la arquitectura del *framework* Symfony2, este patrón se muestra muy evidenciado en la capa perteneciente al modelo de datos. En la misma se encuentran las clases encargadas de interactuar directamente con la base de datos a través del ORM Doctrine, las cuales generalmente se encuentran bajo el nombre de *name\_entityRepository*, presentando la responsabilidad de realizar directamente las acciones de consultas debido a que conocen los atributos necesarios para realizar dichas operaciones.

**Creador:** Este patrón guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento (44). El mismo se evidencia en la

## Capítulo 2: Propuesta de solución. Análisis y diseño

### Módulo para la búsqueda y recuperación de información en RHODA v3.0

acción `search_federateAction()` de la clase controladora `DefaultController.php` cuando se crea un objeto de Tipo `DOMDocument` para trabajar con el archivo de extensión XML llamado manifiesto, como se muestra en la siguiente figura.

```
public function search_federateAction(Request $request) {
    $em = $this->getDoctrine()->getManager();
    $dateBefore = microtime();
    $searching = true;
    $repositories = $em->getRepository('RhodaInteroperabilityBundle:SQI')->findAll();
    if ($request->getMethod() == 'POST') {
        $query = trim($request->get('q'));
        $repositoriesPostTemp = $request->get('repositories');
        $results = array();
        if (strlen($query) > 0) {
            if (count($repositoriesPostTemp) > 0) {
                $repositoriesPost = $em->getRepository('RhodaInteroperabilityBundle:SQI')->findById($repositoriesPostTemp);
                $count = 0;
                $dom = new \DOMDocument();
                foreach ($repositoriesPost as $repository) {
                    $url_session_services = $repository->getUrlAuthentication();
                    $url_query_services = $repository->getUrlSearch();

                    $sqiConsumer = new \Rhoda\SearchBundle\Utils\SqiConsumer($url_session_services, $url_query_services);
                    if ($sqiConsumer->__get("_error")) {
                        $searching = false;
                        $this->get('session')->getFlashBag()->add('error', 'Error de conexión');
                    }
                }
            }
        }
    }
}
```

Ilustración 11 Ejemplo de utilización del patrón Creador

**Bajo Acoplamiento:** Este patrón plantea que la dependencia entre las clases que conforman la arquitectura del sistema debe ser mínima, debido a que a la hora de realizar modificaciones en la clase, no afecte la estructura de las restantes. En la aplicación se concibió mantener separadas las clases pertenecientes al modelo de datos, las vistas de usuario y las clases controladoras.

**Alta Cohesión:** Este patrón plantea que las clases se deben apoyar del funcionamiento de otras, para lograr su objetivo; y no incluir en ella misma todo el proceso del negocio, ya que sería difícil de comprender, reutilizar y le afectarían constantemente los cambios. Ejemplo de ello dentro de la aplicación, cuando son de realizar búsquedas por metadatos recolectados, la acción `search_metadata_catalogAction()` utiliza la colaboración de otras clases, tanto del

## Capítulo 2: Propuesta de solución. Análisis y diseño

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

modelo como de la vista, y además hace llamada a la función `isValidQuery($query)`, con el objetivo de verificar que el parámetro introducido no contenga algunos caracteres inválidos como se representa en la siguiente figura.

```
public function search_metadata_catalogAction(Request $request) {
    $em = $this->getDoctrine()->getManager();
    $dateBefore = microtime();
    $repositories = $em->getRepository('RhodaInteroperabilityBundle:OAI')->findAll();
    if ($request->getMethod() == 'POST') {
        $searching = true;
        $query = trim($request->get('q'));

        $repositoriesPost = $request->get('repositories');
        $results = array();
        if (strlen($query) > 0 & $this->get('rhoda_search.search')->isValidQuery($query)) {
            if (count($repositoriesPost) > 0) {
                $repositoriesPost = $em->getRepository('RhodaInteroperabilityBundle:OAI')->findById($repositoriesPost);
                foreach ($repositoriesPost as $repository) {
                    $repositoriesMetadata = $em->getRepository('RhodaInteroperabilityBundle:OAIResource')->findBy(array('repository_id'
                    => $repository));
                    foreach ($repositoriesMetadata as $repositoryMetadata) {
                        if ($this->get('rhoda_search.search')->hasMatch($query, $repositoryMetadata->getTitle() . ' '
                        | $repositoryMetadata->getDescription() . ' ' . $repositoryMetadata->getAuthor() . ' ' . $repositoryMetadata->
                        $results[] = $repositoryMetadata;
                    }
                }
            }
        }
    }

    public function isValidQuery($query) {
        return strlen(str_replace(array('"', ':', '(', ')', ' ', ' '), '', $query)) != 0;
    }
}
```

Ilustración 12. Ejemplo de utilización del patrón Alta Cohesión

**Controlador:** En Symfony2 todas las peticiones web son manipuladas por el controlador frontal, que es el punto de entrada de toda la aplicación en un entorno determinado. Los eventos se separan en varias acciones para que se aumente la cohesión y disminuya el acoplamiento. En el caso del módulo a desarrollar se evidencia en la clase controladora `DefaultController.php`.

### 2.6.2 Patrones de diseño GoF

Los patrones GoF se clasifican en tres categorías basadas en su propósito: creacionales, estructurales y de comportamiento (43). Seguidamente se presenta el patrón GoF utilizado para el diseño del sistema.

**Decorador:** Este patrón plantea la utilización de una o varias plantillas globales, que guarden el código que es usual en varias plantillas del sistema, para no tener que repetirlo en cada interfaz. En la presente aplicación se utilizó una herencia de tres niveles para reutilizar el máximo código posible como se representa en la siguiente figura.

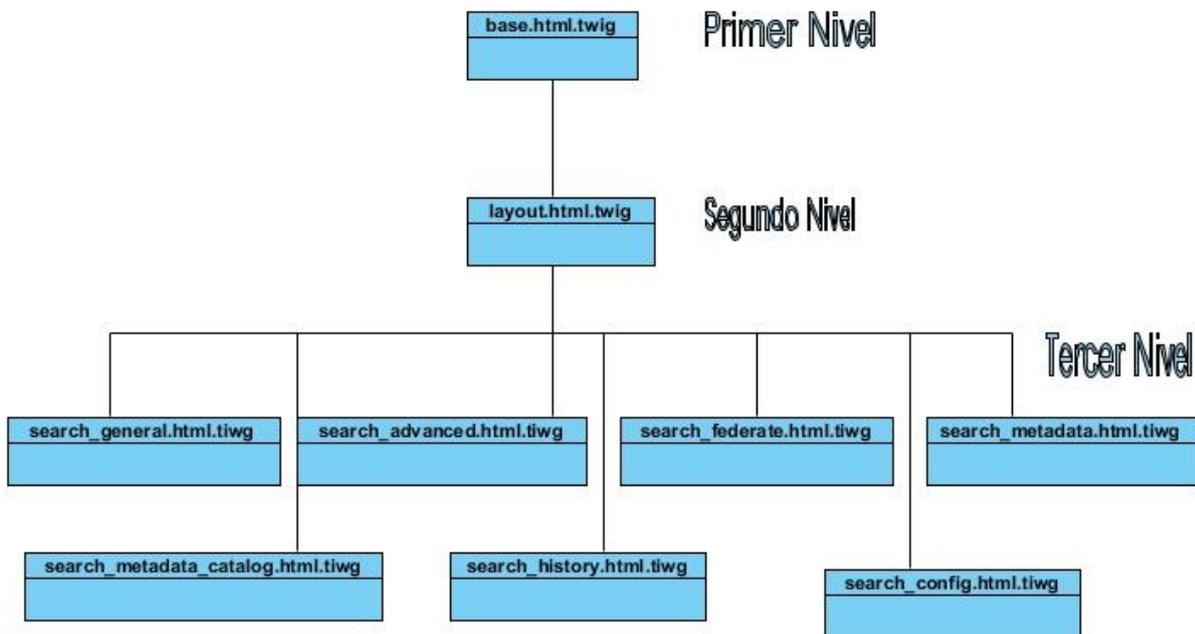


Ilustración 13. Ejemplo de utilización del patrón Decorador

**Strategy:** Este patrón plantea la utilización de varias estrategias para representar comportamientos distintos, utilizando una misma interfaz. Estos comportamientos pueden ser modificados e intercambiados en la aplicación, sin que afecte la lógica del sistema. En este trabajo es utilizado cuando se necesita crear un label<sup>41</sup> de tipo objeto de aprendizaje o recurso educativo como se representa en la siguiente figura.

```

    public function searchCreateLabel() {
        $this->setLabelSearch(self::SEARCH_OA_R);

        if ($this->config->getTypeResult() == 1) {
            $this->setLabelSearch(self::SEARCH_OA);
        }
        if ($this->config->getTypeResult() == 0) {
            $this->setLabelSearch(self::SEARCH_R);
        }
    }
  
```

Ilustración 14. Ejemplo de utilización del patrón Strategy

### 2.6.3 Patrón arquitectónico Modelo Vista Controlador

Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos. En el Modelo se

<sup>41</sup> Label: representa una etiqueta para un elemento en una interfaz de usuario.

encapsulan los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y/o comportamiento de entrada. Define la lógica de negocio en la cual la base de datos pertenece a esta capa. La vista es la encargada de originar las páginas que son mostradas como resultado de las acciones. En el controlador se encuentran las acciones, las cuales son el núcleo de la aplicación, pues contienen toda la lógica de la misma. Estas acciones utilizan el modelo y precisan las variables para la vista. A través del *framework* Symfony son gestionadas todas las solicitudes por un controlador frontal (44).

En el módulo propuesto, un ejemplo de este patrón se observa en el bundle<sup>42</sup> SearchBundle donde se encuentra la clase controladora DefaultController.php; en el directorio Entity se sitúa las clases de entidades que representa el modelo History.php y Configuration.php y en el directorio *Resources* están contenidas las plantillas .twig<sup>43</sup>, los archivos javascript y css.

## 2.7 Modelo del diseño

El Modelo de diseño es un modelo de objetos que describe la realización física de los CU centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, el Modelo de diseño sirve de abstracción a la implementación del sistema, por lo que es utilizado como una entrada fundamental de las actividades de implementación.

### 2.7.1 Diagrama de clases del diseño

Son clases con un mayor nivel de detalle, que se conciben para satisfacer los requisitos funcionales y no funcionales, teniendo en consideración la tecnología en la cual se implementará el diseño. Muestran el diseño del sistema desde un punto de vista estático, a través de una colección de elementos declarativos, colaboraciones y sus relaciones (21).

Los Diagramas de clases del diseño (DCD) constituyen diagramas de clases a los que se le aplican extensiones UML llamados estereotipos (pequeñas etiquetas que aplicadas a los elementos o relaciones de un diagrama indican significado adicional).

En el presente subepígrafe se representaran los DCD correspondiente al CU Realizar búsqueda general de RE y Realizar búsqueda avanzada de RE, los restantes DCD se encontraran en el **Anexo IV**.

---

<sup>42</sup> Bundle es un directorio que contiene todo tipo de archivos dentro una estructura jerarquizada de directorios.

<sup>43</sup> Twig es un motor de plantillas

# Capítulo 2: Propuesta de solución. Análisis y diseño

## Módulo para la búsqueda y recuperación de información en RHODA V3.0

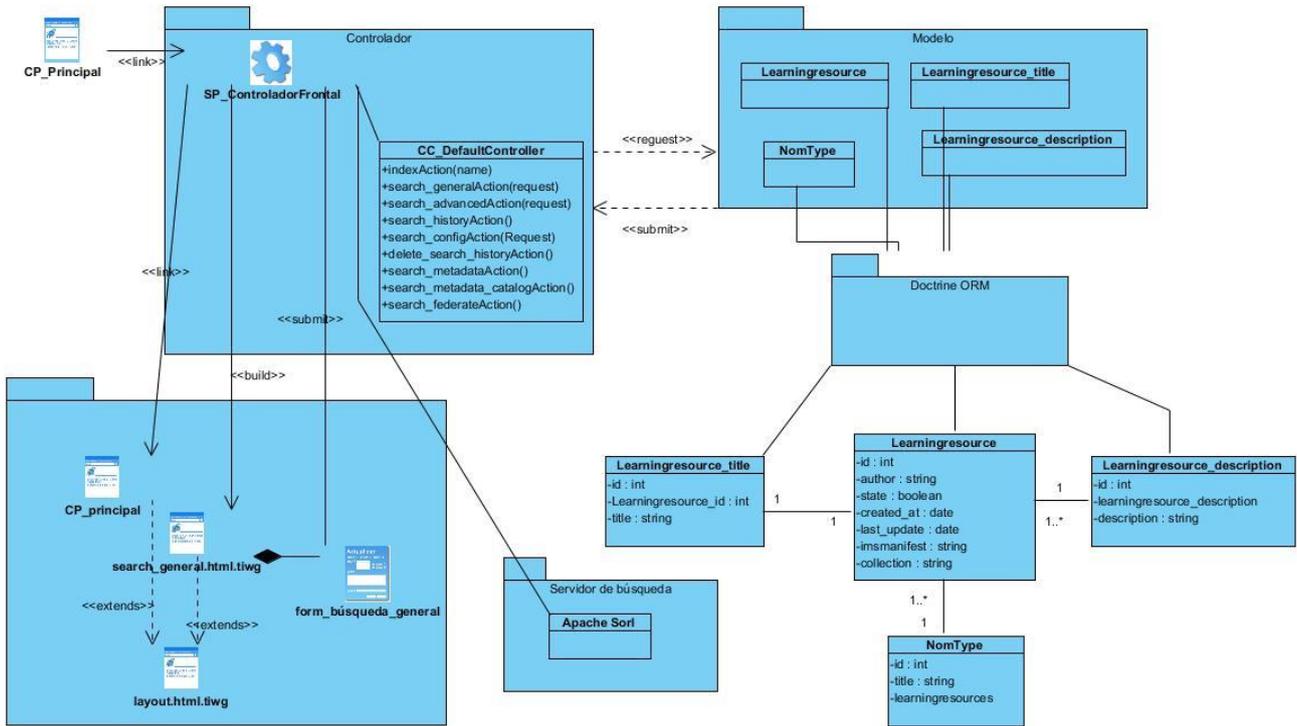


Ilustración 15. DCD Realizar búsqueda general de RE

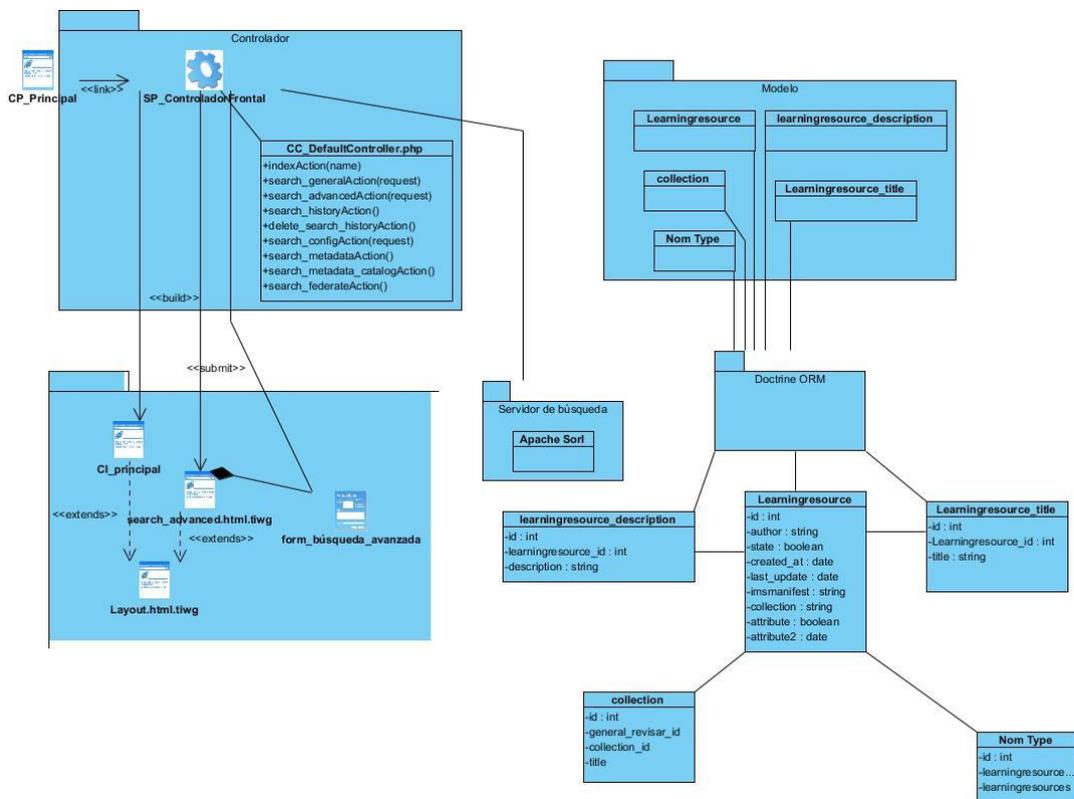


Ilustración 16 DCD Realizar búsqueda avanzada de RE

### **Conclusiones parciales**

Con la culminación del presente capítulo se delimitaron los artefactos necesarios según la metodología seleccionada, con el objetivo de atenuar la complejidad y facilitar una mejor comprensión de la solución propuesta. Se elaboraron los Diagramas de casos de uso y las relaciones existentes entre actores y CU, dando a conocer los usuarios que interactúan con el sistema y una representación detallada de los requisitos que engloban los mismos. Se realizó una descripción minuciosa de los CU, se establecieron los requisitos funcionales y no funcionales, lo que favorece la aceptación y calidad del producto Este segmento del flujo de trabajo contribuye a refinar el entendimiento de cada proceso o tarea que debe programarse durante la implementación y prueba del sistema.

## Capítulo 3: Implementación y pruebas

### Introducción

Durante las fases anteriores de definición y descripción de las características del sistema, se ejecutaron un conjunto de tareas necesarias antes de dar inicio a la implementación de la propuesta de solución y posteriormente a las pruebas. Durante este flujo de trabajo hay elementos y artefactos que influyen directamente como lo es el Modelo de datos, Modelo de Implementación, Diagrama de componentes y los Casos de prueba que finalmente permiten comprobar que se construyó el software acorde a lo que se especificó en los requerimientos.

### 3.2 Modelo de datos

El Modelo de datos se compone de tres piezas de información interrelacionadas: los objetos de datos, los atributos que describen a estos y la relación que conecta a los objetos de datos (45). En este acápite se presenta el Modelo de datos a utilizar en la implementación del sistema, donde serán almacenados los datos necesarios, para darle cumplimiento a los requerimientos del negocio.

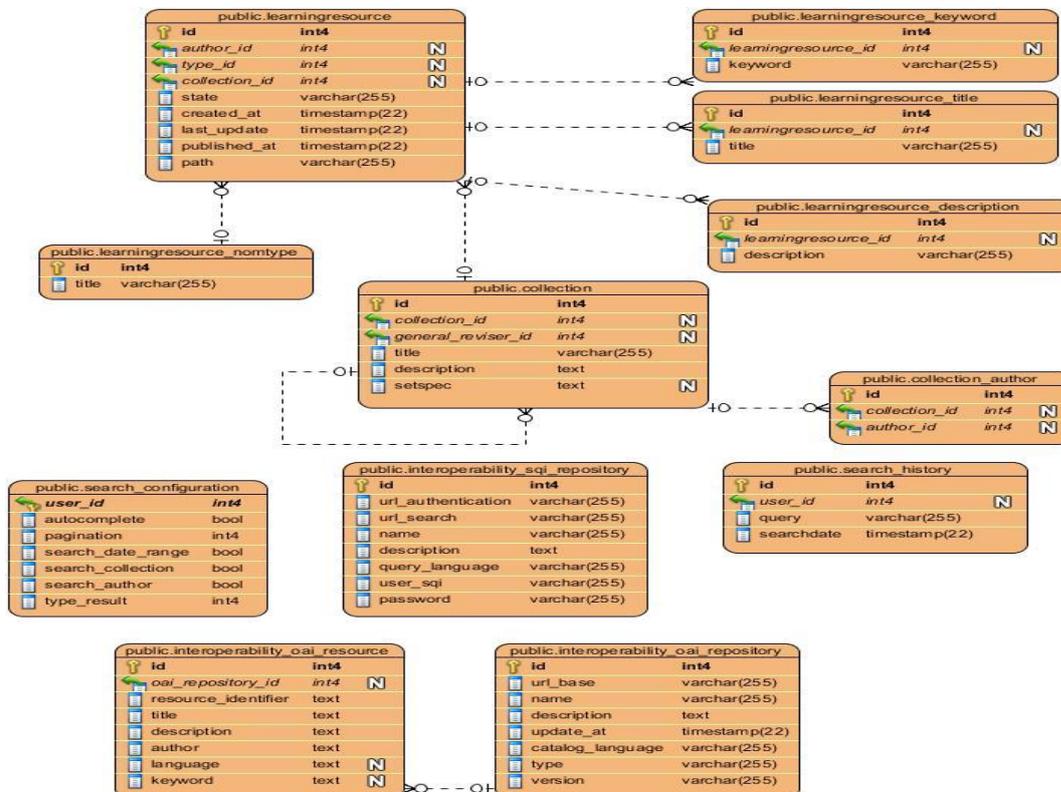


Ilustración 17. Modelo de datos

*Capítulo 3: Implementación y pruebas*

*Módulo para la búsqueda y recuperación de información en RHODA  v3.0*

A continuación se muestra una breve descripción de algunas de las tablas asociadas con la implementación del módulo de búsqueda y recuperación de recursos educativos en RHODA 3.0, las restantes tablas se pueden encontrar en el **Anexo V**.

Tabla 5. Descripción de la tabla Learningresource

| <b>Learningresource</b> |                           |   |
|-------------------------|---------------------------|---|
| <b>Atributo</b>         | <b>Tipo</b>               | <b>Descripción</b>  |
| Id                      | Integer                   | Número que identifica y es único para cada recurso educativo. |
| author_id               | Integer                   | Identificador de los actores.                                 |
| collection_id           | Integer                   | Identificador de las colecciones.                             |
| State                   | character<br>varying(255) | Estado del recurso educativo.                                 |
| created_at              | timestamp(0)              | Fecha inicial cuando fue creado el recurso educativo.         |
| last_update             | timestamp(0)              | Última fecha de actualización referente al recurso educativo. |
| published_at            | timestamp(0)              | Determina si el recurso es publicado.                         |
| Path                    | character<br>varying(255) | Dirección del fichero imsmanifest.xml.                        |

Tabla 6. Descripción de la tabla Learningresource\_keyword

| <b>Learningresource_keyword</b> |                           |   |
|---------------------------------|---------------------------|---|
| <b>Atributo</b>                 | <b>Tipo</b>               | <b>Descripción</b>  |
| Id                              | Integer                   | Número que identifica y es único para cada palabra clave. |
| learningresource_id             | Integer                   | Identificador del recurso educativo.                      |
| Keyword                         | character<br>varying(255) | Nombre de las palabras claves.                            |

Tabla 7. Descripción de la tabla Learningresource\_title

| <b>Learningresource_title</b> |                           |   |
|-------------------------------|---------------------------|---|
| <b>Atributo</b>               | <b>Tipo</b>               | <b>Descripción</b>  |
| Id                            | Integer                   | Número que identifica y es único para cada título de los recursos educativos. |
| learningresource_id           | Integer                   | Identificador del recurso educativo.  |
| Title                         | character<br>varying(255) | Título del recurso educativo.   |

Tabla 8. Descripción de la tabla Learningresource\_description

| <b>Learningresource_description</b> |                           |  |
|-------------------------------------|---------------------------|--|
| <b>Atributo</b>                     | <b>Tipo</b>               | <b>Descripción</b>   |
| Id                                  | Integer                   | Número que identifica y es único para cada descripción de los recursos educativos. |
| learningresource_id                 | Integer                   | Identificador del recurso educativo.   |
| Description                         | character<br>varying(255) | Descripción del recurso educativo.   |

### 3.3 Modelo de implementación

El Modelo de implementación describe cómo los elementos del Modelo de diseño se implementan en términos de componentes. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados (46).

#### 3.3.1 Diagrama de componentes

Este modelo representa en términos de componentes los ficheros de código fuente, scripts, ficheros de código binario, ejecutables y todos los elementos necesarios para desarrollar el subsistema, además de mostrar la relación existente entre los mismos (42).

A continuación se representa el diagrama de componentes asociado con la propuesta de solución y seguidamente se describen los elementos que forman parte del mismo.

# Capítulo 3: Implementación y pruebas

## Módulo para la búsqueda y recuperación de información en RHODA v3.0

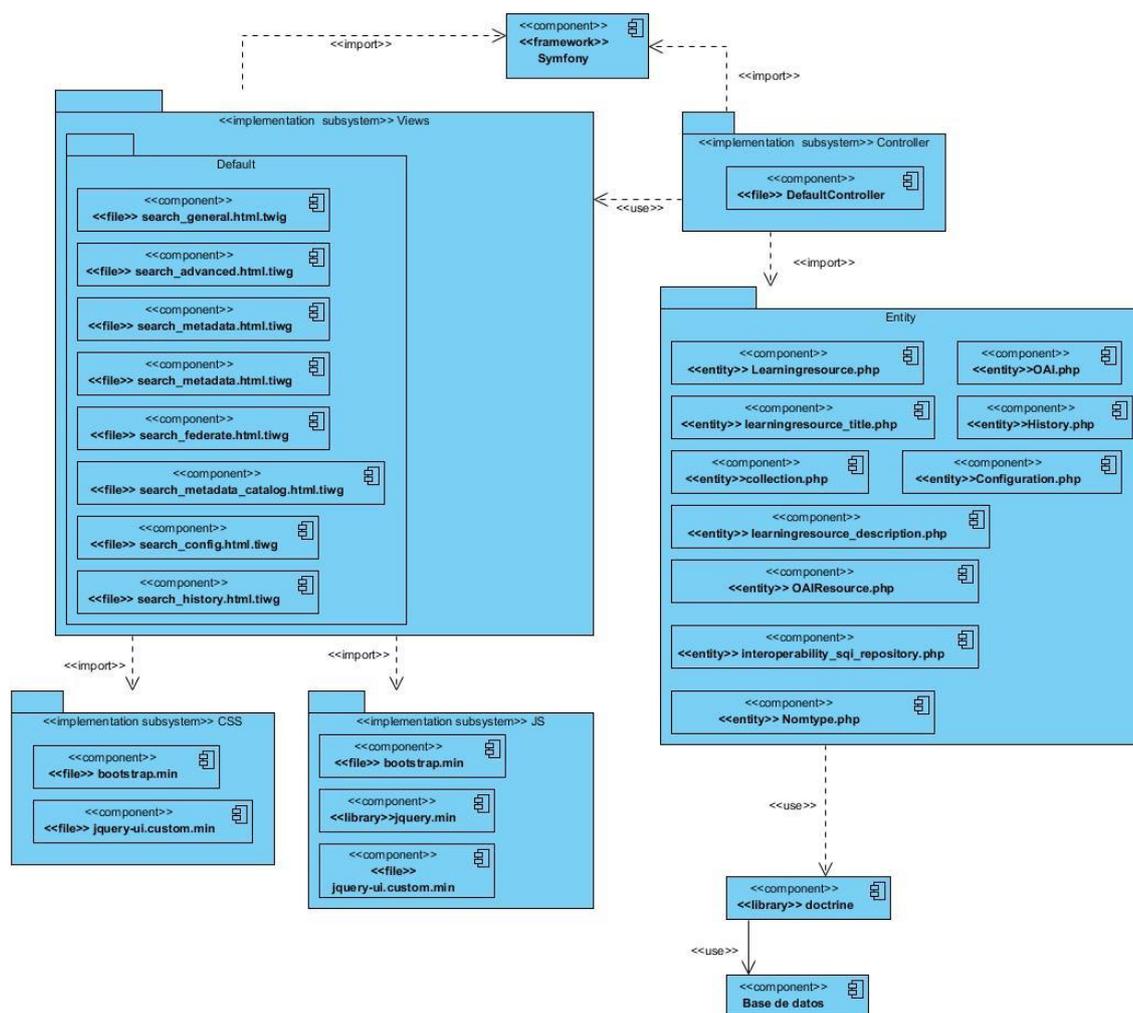


Ilustración 18. Diagrama de componentes

**Paquete Entity:** Agrupa las clases entidad, establece las relaciones y dependencias entre el paquete Controller y el componente Doctrine.

**Paquete Views:** Agrupa los componentes relacionados con las vistas, tales como: las plantillas, las direcciones y los formularios.

**Paquete Controller:** Agrupa la clase controladora encargada de gestionar todas las acciones para las plantillas, establece las relaciones y dependencias con los componentes de Symfony y el paquete Entity.

**Paquete CSS:** Contiene los archivos que le brindan estilos a las plantillas.

**Paquete JS:** Almacena los archivos con código javascript.

**Componente Symfony:** Contiene todos los elementos del *framework* que intervienen en la realización de los CU, establece las relaciones y dependencias con los paquetes Views y Controller.

**Componente doctrine:** Permite generar clases a partir de una base de datos creada.

### 3.4 Pruebas

La Prueba es la disciplina del proceso de ingeniería de software cuyo propósito es integrar y probar el sistema (47). Elemento crítico para la garantía de la calidad del software y representan una revisión final de las especificaciones, del diseño y de la codificación. Su objetivo fundamental es descubrir diferentes clases de errores con la menor cantidad de tiempo y de esfuerzo (42). Dentro de los procedimientos para la realización de las pruebas al software se encuentran las pruebas de Caja negra y las pruebas de Caja blanca.

#### 3.4.1 Pruebas de Caja blanca

La prueba de Caja blanca, en ocasiones llamada prueba de Caja de cristal, es un método de prueba que usa la estructura de control descrita como parte del diseño al nivel de componentes para derivar casos de prueba (1). Estas pruebas son importantes para determinar errores internos en la estructura y lógica del código generado a partir de la programación. Para la verificación del correcto funcionamiento de las clases implementadas se realizaron las pruebas Unitarias.

#### Pruebas unitarias realizadas

Symfony 2 ha optado por utilizar la librería PHPUnit, de esta forma, los *tests* (pruebas) unitarios de Symfony 2 combinan la potencia de PHPUnit con las utilidades y facilidades proporcionadas por el *framework* de desarrollo. Por convención, cada test unitario se define en una clase cuyo nombre acaba en *Test* encontrándose dentro del directorio Tests/ del bundle. PHPUnit utiliza *assertions* (aserciones) para verificar que el comportamiento de una unidad de código es el esperado. Cuando se produce un error, PHPUnit muestra el texto “*failures*” (fallo) como resumen de la ejecución. Antes muestra el listado de todos los *tests* que han fallado, indicando para cada error la clase y método erróneos. El mensaje “*ok*” (Bien) indica que todos los *tests* se han ejecutado correctamente (32).

Estas pruebas fueron realizadas a las clases History.php y Configuration.php del Modelo de datos como se representa en las siguientes ilustraciones.

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

```
Cosme@D5-L102-PC18:/var/www/cosme_all/rhoda3.0$ phpunit -c app src/Rhoda/SearchBundle/Tests/Entity/HistoryTest.php
PHPUnit 3.6.10 by Sebastian Bergmann.

Configuration read from /var/www/cosme_all/rhoda3.0/app/phpunit.xml.dist

F.

Time: 1 second, Memory: 3.50Mb

There was 1 failure:

1) HistoryTest::testValidateQuery
La consulta debe tener al menos 10 caracteres
Failed asserting that 0 is greater than 0.

/var/www/cosme_all/rhoda3.0/src/Rhoda/SearchBundle/Tests/Entity/HistoryTest.php:
35

FAILURES!
Tests: 2, Assertions: 5, Failures: 1.
Cosme@D5-L102-PC18:/var/www/cosme_all/rhoda3.0$
```

Ilustración 19. Prueba Unitaria realizada a la entidad HistoryTest

```
Cosme@D5-L102-PC18:/var/www/cosme_all/rhoda3.0$ phpunit -c app src/Rhoda/SearchBundle/Tests/Entity/ConfigurationTest.php
PHPUnit 3.6.10 by Sebastian Bergmann.

Configuration read from /var/www/cosme_all/rhoda3.0/app/phpunit.xml.dist

F..

Time: 0 seconds, Memory: 3.50Mb

There was 1 failure:

1) ConfigurationTest::testValidatePagination
El paginado no puede ser un número negativo
Failed asserting that 0 is greater than 0.

/var/www/cosme_all/rhoda3.0/src/Rhoda/SearchBundle/Tests/Entity/ConfigurationTest.php:27

FAILURES!
Tests: 3, Assertions: 5, Failures: 1.
Cosme@D5-L102-PC18:/var/www/cosme_all/rhoda3.0$
```

Ilustración 20. Prueba unitaria realizada a la entidad ConfigurationTest

Fueron realizadas dos iteraciones de las pruebas Unitarias, en las cuales se detectaron los siguientes errores:

1era iteración:

- Algunas funcionalidades no validaban los datos de entrada.

- No se validaba que todos los campos deben ser llenados, es decir que no se dejen campos vacíos.

Estos errores fueron corregidos de manera inmediata y en la segunda iteración no se detectaron errores, por tanto las funcionalidades probadas funcionan correctamente.

### 3.4.2 Pruebas de Caja negra

Las pruebas de Caja negra, también conocidas como pruebas de Comportamiento, se concentran en los requisitos funcionales del software. Una prueba de este tipo examina algún aspecto funcional del sistema que tiene poca relación con la estructura lógica interna del software. Se encargan de encontrar errores de: funciones incorrectas o faltantes, de interfaz, en estructuras de datos o el acceso a base de datos externas, errores de comportamiento y errores de inicialización.

#### Métodos de pruebas de Caja negra

Para ejecutar las pruebas de Caja negra es necesario utilizar métodos para obtener mejores resultados, entre los que se encuentran según Pressman (1):

- **Método de partición equivalente:** divide el dominio de entradas de un programa en clases de datos, a partir de las cuales pueden derivarse Casos de prueba.
- **Método del análisis de valores límites:** se encarga del análisis de los valores límites en los campo de entrada.
- **Método de prueba de la tabla ortogonal:** se aplica en problemas donde el dominio de entrada es relativamente pequeño, el método resulta útil para encontrar fallas de región (categoría de error asociado con los defectos de la lógica en un componente de software).

Para la verificación del correcto funcionamiento de las funcionalidades definidas en el sistema se llevaron a cabo las pruebas de Caja negra, utilizando el método de Partición equivalente que propone dividir el dominio de entrada del programa en clases de datos a partir de las cuales pueden revisarse los Casos de prueba (CP). La partición equivalente se esfuerza por definir un CP que descubra ciertas clases de errores, reduciendo el número total de CP que deben desarrollarse. La descripción de los CU va a permitir generar los Diseños de casos de prueba (DCP).

### Casos de prueba

Un Caso de prueba especifica una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse (21).

En esta sección serán expuestos los DCP de los CU Realizar búsqueda general de RE y Realizar búsqueda avanzada de RE, los restantes se podrán localizar en el **Anexo VI**.

Tabla 9. DCP del CU Realizar búsqueda general de RE

| Escenario                       | Descripción   | Buscar | Respuesta del sistema  | Flujo central                           |
|---------------------------------|---|--------|--|---|
| EC 1 Realizar búsqueda general. | Selecciona en el menú búsqueda la opción de la lista desplegable “ <i>Búsqueda general</i> ”.           |        | Muestra un campo de texto.   | Búsqueda/Búsqueda general               |
| EC 2 Introduce el dato.         | Introduce un término que identifique la información que desea buscar. Y hace clic en el botón “Buscar”. | V      | Muestra la cantidad de coincidencias encontradas y el lugar donde realizó la búsqueda, así como el listado de los recursos educativos. De cada recurso se muestran los siguientes datos:<br>- Texto<br>- Nombre del recurso<br>- Autor<br>- Palabras claves<br>- Descripción | Inicio/Búsqueda/Búsqueda general/Buscar |
| EC 3 No introduce el dato       | No introduce el dato.   | /      | Muestra un mensaje de información.   | Búsqueda/Búsqueda general/Buscar        |

|                                      |                                  |   |   |                                  |
|--------------------------------------|----------------------------------|---|---|----------------------------------|
|                                      |                                  |   | Regresa al escenario 2.                                       |                                  |
| EC 4 No se encontraron coincidencias | No se encontraron coincidencias. | / | Muestra un mensaje de información.<br>Regresa al escenario 2. | Búsqueda/Búsqueda general/Buscar |

Tabla 10. DCP del CU Realizar búsqueda avanzada de RE

| Escenario                           | Descripción   | Ordenar | Contiene | Contiene (AND, OR, NOT) | Contiene (AND, OR, NOT) | Contiene (AND, OR, NOT) | Publicado(Desde: Hasta) | Autor | Colección | Subcolección | Respuesta del sistema  | Flujo central                     |
|-------------------------------------|---|---------|----------|-------------------------|-------------------------|-------------------------|-------------------------|-------|-----------|--------------|--|-----------------------------------|
|                                     |   |         |          |                         |                         |                         |                         |       |           |              |  |                                   |
| EC 1<br>Realizar búsqueda avanzada. | Selecciona en el menú búsqueda la opción de la lista desplegable "Búsqueda avanzada". |         |          |                         |                         |                         |                         |       |           |              | Muestra las siguientes opciones<br>-Ordenar:<br>Más relevantes<br>Más recientes<br>-Contiene<br>-Contiene (AND, OR, NOT)<br>-Publicado<br>.Desde<br>.Hasta<br>-Autor<br>-Colección<br>-Subcolección<br>-Buscar | Inicio/Búsqueda/Búsqueda avanzada |
| EC 2<br>Introduce los datos.        | Introduce los términos que identifiquen la información que los que la que             | N / A   | V / A    | N/ A                    | N/ A                    | N/ A                    | N/ A                    | N / A | N/ A      | N/ A         | Muestra la cantidad de coincidencias y el lugar donde realizó la búsqueda, así como el   | Búsqueda/Búsqueda avanzada/Buscar |

## Capítulo 3: Implementación y pruebas

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

|   |  |       |   |      |      |      |       |       |      |      |  |                                   |
|---|--|-------|---|------|------|------|-------|-------|------|------|--|-----------------------------------|
|   | desea buscar.<br>Y hace clic en el botón "Buscar".                       |       |   |      |      |      |       |       |      |      | listado de los recursos educativos.<br>De cada recurso se muestran los siguientes datos:<br>-Texto<br>- Nombre del recurso:<br>- Autor<br>- Palabras claves<br>- Descripción |                                   |
| EC 3 Fecha de inicio (Desde) mayor que la fecha de fin (Hasta). | Selecciona la Fecha de inicio (Desde) mayor que la fecha de fin (Hasta). | N / A | V | N/ A | I    | N/ A | I     | N / A | N/ A | N/ A | Muestra un mensaje: <i>"La fecha de inicio no puede ser mayor que la fecha final"</i> .<br>Regresa al escenario 2.   | Búsqueda/Búsqueda avanzada/Buscar |
| EC 4 Rango incompleto.  | Selecciona la opción Por fecha(Desde) o (Hasta)                          | N / A | V | N/ A | I    | N/ A | I     | N / A | N/ A | N/ A | Muestra un mensaje: <i>"Rango de fechas" incompleto</i> .<br>Regresa al escenario 2.   | Búsqueda/Búsqueda avanzada/Buscar |
| EC 5 Rellene este campo.  | No selecciona ningún campo y selecciona la opción Buscar.                | I     | I | N/ A | N/ A | N/ A | N/ A  | N / A | N/ A | N/ A | Muestra un mensaje: <i>"Rellene este campo"</i> .<br>Regresa al escenario 2.   | Búsqueda/Búsqueda avanzada/Buscar |
|   |  | I     | I | I    | N/ A | N/ A | N / A | N/ A  | N/ A |      |  |                                   |
|   |  | I     | I | I    | I    | N/ A | N/ A  | N / A | N/ A |      |  |                                   |
|   |  | I     | I | I    | I    | I    | N/ A  | N / A | N/ A |      |  |                                   |

*Capítulo 3: Implementación y pruebas*

**Módulo para la búsqueda y recuperación de información en RHODA  v3.0**

Tabla 11.D CP del CU Realizar búsqueda avanzada, SC1 colección, subcolección y autor

| Escenario                                   | Descripción   |       |           |              | Respuesta del sistema   | Flujo central                            |
|---|---|-------|-----------|--------------|---|--|
|   |   | Autor | Colección | Subcolección |   |  |
| EC 2.1<br>Selecciona e introduce los datos. | Selecciona la colección y/o subcolección y/o autor deseados<br>Y hace clic en el botón “Búsqueda avanzada”. | N/A   | N/A       | N/A          | Muestra el resultado de la búsqueda que coincide con el texto introducido, dando de cada uno de los RE encontrados los datos generales. | Inicio/Búsqueda/Búsqueda avanzada/Buscar |
| EC 2.2 No se encontraron coincidencia.      | Selecciona la colección y/o subcolección y/o autor deseados y selecciona la opción Búsqueda avanzada.       | I     | N/A       | N/A          | Muestra un mensaje de información.<br>Regresa al escenario 2.   | Inicio/Búsqueda/Búsqueda avanzada/Buscar |
|   |   | N/A   | I         | N/A          |   |  |
|   |   | N/A   | N/A       | I            |   |  |

Tabla 12. Descripción de las variables

| No | Nombre de campo | Clasificación      | Valor Nulo | Descripción   |
|----|-----------------|--------------------|------------|---|
| 1  | Ordenar         | Campo de selección | no         | Ordena los resultados según el criterio seleccionado: <ul style="list-style-type: none"> <li>• Más relevantes</li> <li>• Más recientes</li> </ul> |
| 2  | Contiene        | Campo de selección | no         | Campo de texto para determinar el criterio de búsqueda seleccionado mediante una palabra clave.   |

|   |                         |                    |    |   |
|---|-------------------------|--------------------|----|---|
| 3 | Contiene(AND,OR,NOT)    | Campo de selección | No | Campo de texto para determinar mediante una palabra clave el criterio de búsqueda seleccionado, puede ser por uno (1) o hasta tres (3) a la vez, mediante las condiciones (AND, OR, NOT). |
| 4 | Publicado(Desde: hasta) | Campo de texto     | No | Permite la selección de un intervalo establecido por fechas, según el criterio del usuario: <ul style="list-style-type: none"> <li>• Desde</li> <li>• Hasta</li> </ul>                    |
| 5 | Autor                   | Campo de texto     | No | Autor de los recursos educativos.   |
| 6 | Colección               | Campo de selección | No | Campo para seleccionar una colección determinada de recurso educativo.  |
| 7 | Subcolección            | Campo de selección | No | Campo para verificar una subcolección.  |

### 3.5 Resultados de las pruebas

Durante la realización de las pruebas a los requisitos funcionales se encontraron un conjunto de no conformidades las cuales fueron clasificadas en: significativas y no significativas. Para ello se efectuaron tres iteraciones, donde se detectaron un total de 15 no conformidades; 9 de significativas y 6 no significativas, los resultados se exponen en la siguiente ilustración.

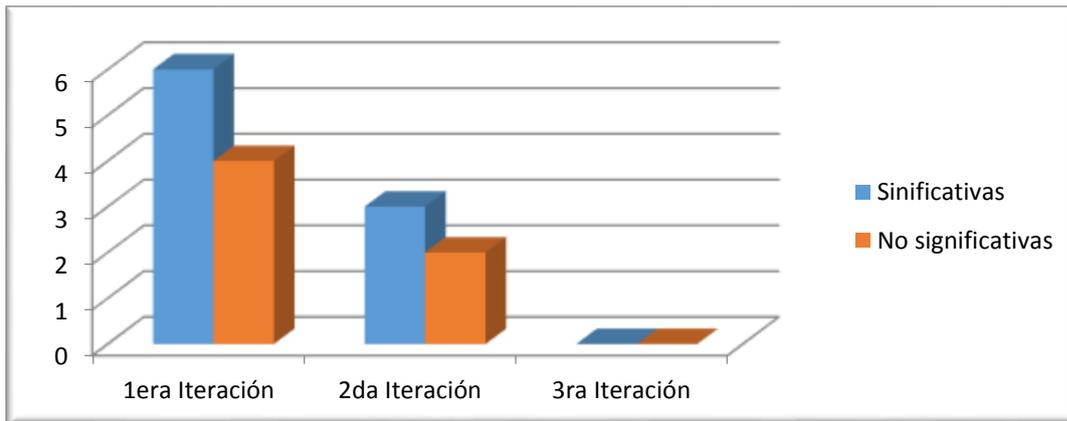


Ilustración 21. Resultado de las iteraciones

Las no conformidades encontradas, una vez concluida cada iteración de prueba, se analizaron por parte del equipo de desarrollo, determinando cuáles constituían fallas del sistema o presentaban variaciones en la descripción de los CU. Las pruebas se realizaron de forma iterativa e incremental, verificando la corrección de los errores de la iteración anterior, lo que contribuyó a mejorar la calidad y funcionalidad del software.

### Conclusiones parciales

Los artefactos desarrollados como parte del modelo de implementación facilitaron una vista física e interna de las clases utilizadas para la implementación de la solución propuesta. Mediante la realización de las pruebas de Unidad se verificó el correcto funcionamiento de las clases definidas. La ejecución de las pruebas funcionales teniendo en cuenta el método de Caja negra permitió detectar y erradicar las no conformidades identificadas en las diferentes iteraciones.

## **Conclusiones generales**

Con el desarrollo de la presente investigación se dio respuesta a los objetivos propuestos, arribando a las siguientes conclusiones:

- El estudio realizado en base a la bibliografía y los sistemas similares existentes, posibilitaron: identificar las especificaciones IMS-SQL y OAI-PMH para lograr la interoperabilidad entre los datos almacenados en los repositorios y el estándar LOM para la descripción de los recursos educativos y definir los tipos de búsqueda a implementar en la propuesta de solución.
- El estudio realizado en base a la selección de las herramientas y tecnologías existentes para el desarrollo del módulo de búsqueda y recuperación de recursos educativos, permitió realizar la implementación de la propuesta de solución planteada.
- Se logró la implementación de un módulo búsqueda y recuperación de información, que permite la búsqueda de recursos educativos en el repositorio RHODA 3.0.
- La aplicación de pruebas al sistema, tales como pruebas Unitarias y pruebas de Caja negra, permitieron validar que la herramienta desarrollada cumple con los requisitos funcionales determinados por el cliente.

## Glosario de términos

A continuación se muestran en orden alfabético los principales términos abordados durante todo el transcurso del presente trabajo.

**Autocompletar búsqueda:** Permite la selección de dos opciones para determinar el autocompletamiento del criterio de búsqueda seleccionado.

**Autor:** Es la persona u organización responsable de la creación del contenido intelectual del recurso.

**Campos de búsqueda:** Permite configurar los siguientes campos (buscar por intervalo de tiempo, buscar por autor y buscar por colecciones /subcolecciones) en la búsqueda avanzada, donde el autor realiza la selección del deseado.

**Colección:** Campo para seleccionar una colección determinada de recurso educativo.

**Contiene:** Campo de texto para determinar el criterio de búsqueda seleccionado mediante palabras claves.

**Contiene (AND, OR, NOT):** Campo de texto para determinar mediante palabras claves el criterio de búsqueda seleccionado, puede ser por uno (1) o hasta tres (3) a la vez, mediante las condiciones (AND, OR, NOT).

**Descripción:** Es una descripción textual del recurso, puede ser un resumen en el caso de un documento.

**Guardar preferencias:** Botón que permite guardar los cambios realizados por el actor.

**Learning Object Metadata (LOM):** Es una especificación en la que se definen un conjunto de metadatos que permitan describir recursos educativos digitales con la finalidad de facilitar la recuperación de los mismos.

**Número de resultados:** Ofrece resultados del 10 al 50(incrementando en 10,20,...50) de acuerdo a la selección del actor.

**Open Archives Initiative Protocol for Metadata Harvesting (AOI-PMH):** Define un mecanismo para la recolección de registros que contienen los metadatos de los repositorios.

**Ordenar:** Ordena los resultados según el criterio seleccionado.

**Palabras claves:** Son los tópicos del recurso. Típicamente expresará las claves o frases que describen el título o el contenido del recurso.

**Publicado (Desde: hasta):** Permite la selección de un intervalo establecido por fechas, según el criterio del usuario.

*Módulo para la búsqueda y recuperación de información en RHODA  v3.0*

**Seleccione:** Campo para seleccionar el o los metadatos por los que se quiere realizar la búsqueda, puede ser por uno (1) o hasta tres (3) a la vez.

**Simple Query Interface (SQI):** Es un protocolo, que proporciona interoperabilidad entre repositorios de recursos educativos y aplicaciones de búsqueda y recuperación de los mismos recursos.

**Subcolección:** Campo para verificar una subcolección.

**Texto:** Elemento que representa la cantidad de coincidencias encontradas.

## Bibliografía referenciada

1. Ribeiro-Neto, R. Baeza-Yates y B. Modern information retrieval. [En línea] 1999. [Citado el: 12 de 1 de 2014.] [ftp://mail.im.tku.edu.tw/seke/slide/baeza-yates/chap10\\_user\\_interfaces\\_and\\_visualization-modern\\_ir.pdf](ftp://mail.im.tku.edu.tw/seke/slide/baeza-yates/chap10_user_interfaces_and_visualization-modern_ir.pdf). 0-201-39829-X.
2. González, Leonardo Rodríguez. Análisis y Diseño de la versión 3.0 de RHODA. 2011.
3. GONZÁLEZ, ROXANA CAÑIZARES. REPOSITORIO DE RECURSOS EDUCATIVOS PARA LAS INSTITUCIONES DE EDUCACIÓN SUPERIOR. La Habana, Cuba : s.n., 2011.
4. Peñalvo, C. López Guzmán & F. J. García. Los Repositorios de Objetos de Aprendizaje como soporte a un entorno elearning. [En línea] 2005. [Citado el: 11 de 02 de 2014.] [http://gredos.usal.es/jspui/bitstream/10366/56649/1/DIA\\_Repositoriosobjetos.pdf](http://gredos.usal.es/jspui/bitstream/10366/56649/1/DIA_Repositoriosobjetos.pdf).
5. Leiva Pérez, Yideilis y Bello Gonzáles. Implementación de una nueva versión del módulo Búsqueda para el Repositorio de Objetos de Aprendizaje RHODA. Ciudad de la Habana, Cuba: Universidad de Ciencias Informáticas : s.n., 2010.
6. G. Salton, Mc Gill, M.J. Introduction to Modern Information Retrieval. [En línea] 1983. [Citado el: 23 de 2 de 2014.]
7. Vidaurázaga, Ignacio Nieto. MEMETRACKER – TDT : Detección y seguimiento de temas en noticias y blogs.
8. Román, J. Villena. Sistemas de Recuperación de Información. [En línea] 1997. [Citado el: 20 de 2 de 2014.] <http://www.mat.upm.es/~jmg/doct00/RecupInfo.pdf>.
9. [En línea] <http://trevinca.ei.uvigo.es/~evali/doctorado0507/sri/RI.pdf>.
10. Naveiras, Diego. Técnicas de indexación y recuperación de documentos utilizando referencias geográficas y textuales. Santiago de Compostela, España, Universidade de Santiago de Compostela : s.n., 2009.
11. [En línea] <http://jolt.merlot.org/>.
12. [En línea] <http://www.careo.org/index.html>.
13. National Science Foundation, National STEM Education Digital Library program. Welcome to the SMETE Digital Library. [En línea] 2010. [Citado el: 5 de 3 de 2014.] <http://www.smete.org/smete/>.
14. Repositorio de recursos educativos de la Universidad Virtual de Salud de Cuba. [En línea] <http://www.acimed.sld.cu/index.php/acimed/index>.
15. Fernandez-Majón, Baltasar. Especificaciones y estándares en e-learning. [En línea] 2006. [Citado el: 6 de 4 de 2014.] [http://reddigital.cnice.mec.es/6/Articulos/Articulo\\_resumen.php](http://reddigital.cnice.mec.es/6/Articulos/Articulo_resumen.php).
16. RAE. Real Academia Española. [En línea] 2010. [Citado el: 6 de 3 de 2014.] [www.rae.es](http://www.rae.es).
17. Salvador, Broche Orlando Felipe and Soler, Martín Javier. Implementación de un Repositorio de. [En línea] 2010. [Citado el: 8 de 4 de 2014.]
18. Iglesias, Borja Manero. Estudio de la propuesta IMS de estandarización de enseñanza asistida. [En línea] 2003.
19. Nguen, Van Nhu. Binding the Simple Query Interface. [En línea]
20. Fernando de la Prieta, Ana Gil González, Juan M. Corchado y Eladio Sanz. Sistema multiagente orientado a la búsqueda, recuperación y filtrado de objetos digitales educativos.

# Módulo para la búsqueda y recuperación de información en RHODA v3.0

[En línea] [Citado el: 11 de 3 de 2014.]

<https://dtm.usal.es/archivos/cedi%20de%20la%20prieta.pdf>.

21. Jacobson, Ivar, Booch, Grady and Rumbaugh, James. El proceso unificado de desarrollo de software. [En línea] 2000. 84-7829-036-2.

22. Rumbaugh, James, Booch, Grady y Ivar Jacobson. El Proceso Unificado de Desarrollo de Software. [En línea] 2000. [Citado el: 4 de 2 de 2014.]

23. Lobos, María Elena de. emagister. [En línea] <http://www.emagister.com/>.

24. González, Enrique. Aprender a Programar. [En línea] [Citado el: 10 de 2 de 2014.] <http://www.aprenderaprogramar.com>.

25. Pérez, Javier Eguíliz. Introducción a XHTML. [En línea] 2013. [Citado el: 4 de 2 de 2014.] <http://www.w3.org/standards/webdesign/htmlcss>.

26. Domínguez-Dorado, M. Bases de datos en el cliente con JavaScript DB. [En línea] 2005.

27. TODOJS. [En línea] 2013. <http://www.todojs.com/ref/javascript/base.html>.

28. CSS3. [En línea] [Citado el: 3 de 1 de 2014.] <http://www.ecured.cu/index.php/CSS>.

29. [En línea] [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/Framework.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf).

30. [En línea] <http://www.ecured.cu/index.php/JQuery>.

31. Mark Otto, Jacob Thornton, Traductor: Javier Eguiluz. Bootstrap 3, el manual oficial.

32. Symfony2. [En línea] [Citado el: 3 de 3 de 2014.] <http://symfony.cubava.cu/introduccion/symfony-2/>.

33. Pecos, Daniel. PostgreSQL vs. MySQL. PostgreSQL vs. MySQL. [En línea] [Citado el: 11 de 3 de 2014.] <http://danielpecos.com/documents/postgresql-vs-mysql/>.

34. [En línea] [http://librosweb.es/symfony\\_1\\_2/capitulo\\_1/conceptos\\_basicos.html#mapeo\\_de\\_objetos\\_a\\_bases\\_de\\_datos\\_orm](http://librosweb.es/symfony_1_2/capitulo_1/conceptos_basicos.html#mapeo_de_objetos_a_bases_de_datos_orm).

35. [En línea] <http://www.ecured.cu/index.php/Doctrine>.

36. [En línea] [http://www.ecured.cu/index.php/Visual\\_Paradigm](http://www.ecured.cu/index.php/Visual_Paradigm).

37. [En línea] [Citado el: 10 de 2 de 2014.] <http://www.ecured.cu/index.php/NetBeans>.

38. [En línea] <http://tecnoweb2.com/que-es-servidor-web>.

39. [En línea] [http://httpd.apache.org/docs/2.0/new\\_features\\_2\\_0.html](http://httpd.apache.org/docs/2.0/new_features_2_0.html).

40. [En línea] <http://gratis.portalprogramas.com/Apache-Solr.html>.

41. Sommerville, Ian. Ingeniería del Software. s.l. [En línea] 2005. 84-7829-074-5.

42. Presman, Roger. Ingeniería de Software: Un Enfoque Práctico. 2005. 9701054733.

43. Larman, Craig. UML y Patrones 2da edición. Una introducción al análisis y diseño orientado a objetos y al proceso unificado.

44. Eguiluz, Javier. Maestros dela web. El framework Symfony, una introducción práctica (I parte). [En línea] 2007. [Citado el: 28 de 02 de 2014.] <http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/>.

45. S.Pressman, Roger. Ingeniería de software.Un enfoque práctico. Mexico : s.n., 2002. 8448132149..

46. Acuña, Karenny Brito. RUP Diseño e implementación del sistema. [En línea] [Citado el: 5 de 2 de 2014.]  
<http://www.eumed.net/libros/2009c/584/RUP%20Diseno%20e%20implementacion%20del%20sistema.htm>.
47. Corp, IBM. Rational Unifiel Process. Rational Unifiel Process. [En línea] 2007.
48. Grabiél H. Tolosa, Fernando R.A Bordignon. Introducción a la Recuperación de Información. [En línea] [Citado el: 10 de 02 de 2014.]
49. GRAEME, D. Learning Objects Repositories. [En línea] 2004.
50. EDUSOURCE. EDUSOURCE . [En línea] 2011. [Citado el: 2014 de 3 de 5.]  
<http://www.edusource.org/>.
51. Alvarez, Miguel Angel. Manual de jQuery.
52. Foundation, The jQuery. The jQuery Foundation. [En línea] [Citado el: 2 de 2 de 2014.]  
<http://jquery.com/>.
53. Scott, Martin Fowler and Kendal. UML Gota a Gota. s.l. [En línea] 1996. 9684443641.
54. López, Guzmán Clara. Los Repositorios de Objetos de Aprendizaje como soporta un entorno elearning. Salamanca, España : s.n., 2005.
55. Márquez, José Manuel Vázquez. Estado del arte del eLearning. Ideas para la definición de una plataforma universal. [En línea] 2007.
56. EGUILUZ, JAVIER. Introducción a JavaScript . [En línea] [Citado el: 8 de 3 de 2014.]  
<http://librosweb.es/javascript/index.html>.
57. G, Susana. Visual Paradigm. [En línea] 18 de 9 de 2012. [Citado el: 11 de 4 de 2014.]  
[http://www.ecured.cu/index.php/Visual\\_Paradigm](http://www.ecured.cu/index.php/Visual_Paradigm).
58. Gretter, Plá Reinoso. Sistema Automatizado para el diseño y evaluación de puestos de trabajo ERGODIS 3.0. Ciudad de la Habana : s.n., 2007.
59. PROJECT, J.T. Report on Open Source Learning Object Repository Systems. [En línea] 2012