



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 4

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS**

SISDREA:

**SISTEMA BASADO EN CASOS PARA LA IDENTIFICACIÓN DE PROBLEMAS EN LOS DISEÑOS DE
RECURSOS EDUCATIVOS ABIERTOS**

AUTORES

Yoelkys Hernández Aracil

Romilio Anibal Corona Prendes

TUTORES

Ing. Yasirys Terry González

Dra. C. Dunia María Colomé Cedeño

La Habana, junio del 2015

“Año 57 de la Revolución”

“Al mundo nuevo corresponde la Universidad nueva. A nuevas ciencias que todo lo invaden, reforman y minan nuevas cátedras. Es criminal el divorcio entre la educación que se recibe en una época, y la época (...) En tiempos científicos, universidad científica”.



AGRADECIMIENTOS

Yoelkys Hernández Aracil

En primer lugar agradecerle a mi país por permitirme cumplir el sueño de ser universitario y ahora ingeniero y sentir el orgulloso de ser parte de la salvaguarda de la Revolución.

A la UCI y la FEU por enseñarme tanto de la vida y de mí mismo.

A mi mamá por siempre creer en mí, por enseñarme a querer sin límites y porque todo lo que soy y seré habrá sido por ti mami y a ti lo dedicaré con amor.

A mi papá por ser esa estrella alta desde que tengo uso de razón, por ser luz y corazón de mi familia y enseñarme y darme todo; te amo papá, nunca me faltes, déjame siempre como principal motivo el de hacerte sentir orgulloso.

A mi abuelas: Ada, por ser mi segunda madre y porque cuando me sentía perdido su amor incondicional me levantaba; y Amparito, por sus consejos y charlas y por enseñarme que la juventud no tiene edad. Las amo mis viejas, gracias por hacerme sentir siempre tan querido.

A mis hermanos Yoyi, Nani, Yoan y la pequeña Yari; al resto de mis abuelos Titi, Terina y Pitín (que aunque no está físicamente lo recuerdo con mucho cariño). A mis primos y tíos, en especial a tía Caideiris por su cariño y por esos días de vacaciones donde podía trastear la computadora de William cuando descubría mi vocación por la informática, por simple que parezca esos momentos me hacían muy feliz; también a mi tía María Julia por soportarme y guiarme desde pequeño.

A mi esposa Yamila por su amor, paciencia y comprensión, por ser mi soporte y por darme el más grande tesoro que tengo; siempre te amaré mi amor, nunca dejes de tolerarme. También a mis suegros por acogerme y cuidar de Sahul.

A Lisbet por haber sido amiga y hermana por más de nueve años, te quiero mucho cebolli; a Fernando y Jake por el apoyo, las charlas y consejos; y a Renides por siempre estar y por enseñarme tanto sobre la Amistad y la Perseverancia; los quiero siempre.

Agradecimientos

A los amigos que quedan para siempre, el grupo 4X01: Yairon, Sandy, Eddy, Kenia, Claudia, Romilio (hermano y compañero de estresis), etc, etc. Del piquete de Holguín: Ale, Eiler y Ariel. A Vladis, Malidia, Claudio, Luis Eduardo y Luis Enrique por su amistad y preocupación; y a Rey, Tony, Alex, Jairo y Javier por su ayuda, ¡gracias!; los quiero mucho.

A mi tutora Yasirys por guiarnos y atendernos siempre a pesar de la gran "carga" y por luchar este sueño junto a nosotros. Al profe Adrián por su gran ayuda y disponibilidad y al profe Osvaldo por sus explicaciones.

De forma general a todas aquellas personas que de una forma u otra han contribuido a que hoy pueda cumplir mi gran anhelo de convertirme en un profesional de nivel superior.

Agradecimientos

Romilio Anibal Corona Prendes:

Antes que todo tengo que agradecer a mi hijo Ronaldo porque desde el momento en que llegó se convirtió en el motivo más grande para luchar y graduarme como ingeniero.

A mis padres por su ejemplo, amor, consejos, dedicación y entrega, para ellos, todo; a mi padre le debo el haber llegado hasta aquí, sus consejos me guiaron y me dieron luz cuando solo era capaz de ver oscuridad; a mi madre la vida, veinticuatro años dedicados a mi educación, salud, todo; su mano dura en los momentos oportunos me hizo el hombre en el que hoy me he convertido; a mi padrastro por estar tanto tiempo ahí, cuidando de mi mamá y de nosotros, por las historias, las experiencias y el amor incondicional.

A mis abuelos, en especial a Romilio que hoy no está entre nosotros, por su sabiduría, sus enseñanzas y experiencias, donde quiera que esté, este título también es para él.

A mis hermanos Leo y Frank, para ti brother, que te propusiste ser mi guía, hoy te digo que lo lograste, fue genial haber compartido contigo la universidad; a mi familia en general que son lo mejor de lo mejor.

A mi tutora Yasirys, a los profes Adrián y Osvaldo por su ayuda con la aplicación; a mis amigos, esos que hicieron de estos cinco años una gran experiencia, al Yuny, Víctor, al Juank, Sandy, Francisco y al Yoe, otro hermano y además compañero de tesis, a todos los demás que aunque no mencione saben quiénes son.

Agradecer a la Revolución cubana por haberme permitido estudiar en esta institución y brindarme los medios para convertirme en un profesional que sea capaz de servirle.

A quien reservé para el final, Rosi, mi novia, que tengo mucho que agradecerle, por amarme cuando no lo merecía porque era ciertamente cuando más lo necesitaba, su amor, confianza y comprensión me aportaron mucho, parte de las fuerzas que me permitieron estar aquí hoy, provienen de ella.

A todos ustedes por estar hoy aquí, ¡gracias!

DEDICATORIA

Yoelkys Hernández Aracil:

Dedico este trabajo y mis estudios a mi familia y amigos, pero en especial a dos personitas: mi hermana Anailys por haber soportado que me fuera a estudiar lejos cuando estábamos muy unidos y porque te amo mucho nani y quiero verte algún día llegar tan alto como tu sueño más ambicioso; y a mi hijo Sahul por llegar e iluminarme la vida y multiplicar mi inspiración y mis razones para existir, cada día te amo más nenito.

Romilio Aníbal Corona Prendes:

Dedico este trabajo a mi hijo, mis padres y mis hermanos, quienes son los pilares de mi vida y fueron capaces de lograr que me convirtiera en ingeniero.

RESUMEN

El creciente uso de las Tecnologías de la Información y las Comunicaciones en la sociedad se ha visto también en la esfera de la Educación, de esta forma surgen los Recursos Educativos Abiertos. Para facilitar el acceso y la gestión de estos recursos se utilizan los repositorios digitales, muchos de los cuales implementan sistemas de revisión que permiten determinar recursos que necesitan ser modificados y/o actualizados en cuanto a su diseño y contenido, a fin de que puedan cumplir de una mejor manera su función educativa. Para asistir este proceso de actualización se propone el sistema basado en casos SISDREA, capaz de, a partir de una base de casos histórica con recursos ya analizados, ofrecer una predicción de los posibles problemas de diseño que pueda contener el recurso que se examina. Para la creación de dicho sistema se interactuó con expertos para determinar los rasgos predictores a utilizar en la máquina de inferencia, se utilizó la técnica del Triángulo de Fuller para calcular el peso de los rasgos predictores y posteriormente se calculó la distancia entre los casos usando el algoritmo K-Nearest Neighbor; tomándose como base de casos la información sobre los Recursos Educativos Abiertos contenida en el repositorio RHODA. A la solución obtenida se le realizaron pruebas para comprobar la cantidad de clasificaciones correctas, así como el grado de satisfacción logrado en las respuestas.

Palabras claves: *diseño, problema, repositorio, recurso educativo abierto, sistema basado en casos*

Introducción	5
Capítulo 1: Fundamentación teórica de la investigación	10
1.1 Recursos Educativos Abiertos y Objetos de Aprendizaje	10
1.1.1 Objeto de Aprendizaje	10
1.1.2 Recursos Educativos Abiertos. Definición e importancia.....	11
1.2 Repositorios digitales de REA.....	12
1.2.1 Repositorio RHODA.....	13
1.2.2 Estándar LOM (Learning Object Metadata) para la catalogación de metadatos.....	14
1.2.3 Sistemas de revisión en los repositorios digitales	15
1.2.4 Instrumentos de revisión de recursos educativos	16
1.3 Interoperabilidad.....	16
1.3.1 REST (Representational state transfer)	16
1.3.2 XML (eXtensible Markup Language).....	17
1.4 Sistemas Expertos.....	18
1.4.1 Estructura de un Sistema Experto	19
1.4.2 Tipos de búsqueda en los Sistemas Expertos	19
1.4.3 Clasificación de los Sistemas Expertos.....	20
1.4.4 Sistema Basado en Casos y búsqueda por semejanza. Justificación de su selección.....	21
1.5 Sistemas Basados en Casos	22
1.5.1 Ciclo del razonamiento basado en casos.....	22
1.5.2 La Recuperación en el Razonamiento Basado en Casos	23
1.5.3 Algoritmos de recuperación de casos	23
1.5.4 Medidas de similitud	26

1.5.5	Métodos de adaptación	27
1.5.6	CBR y Lógica Difusa (Fuzzy Logic)	28
1.6	Soluciones existentes a nivel nacional e internacional	29
1.6.1	ProgEs.....	30
1.6.2	ENSEñanza Artificial Intilligence (ENS-AI)	30
1.6.3	SI-HOLMES.....	30
1.7	Herramientas y tecnologías a utilizar	31
1.7.1	Selection Engine.....	32
1.7.2	Marcos de desarrollo	32
1.7.3	Lenguaje Java	33
1.7.4	NetBeans 8.0.....	33
	Conclusiones parciales	34
Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA		35
2.1	Propuesta de solución	35
2.2	Obtención de la información de los REA almacenados en RHODA	36
2.3	Representación de los casos.....	37
2.3.1	Determinación de los rasgos predictores y objetivos	37
2.3.2	Cálculo del peso de los rasgos predictores usando la técnica del Triángulo de Fuller	39
2.4	Módulo de recuperación	41
2.4.1	Aplicación del algoritmo K-Nearest Nighbor (KNN)	42
2.4.2	Descripción del proceso de recuperación	43
2.5	Módulo de adaptación	44
2.6	Interfaz de E/S.....	46
	Conclusiones parciales	47

Capítulo 3: Implementación y validación del Sistema Basado en Casos SISDREA	49
3.1 Implementación	49
3.1.1 Estándares de codificación	49
3.1.2 Pre-procesamiento de la información. Librería LomStandarInterpreter	52
3.1.3 Estructuración de la base de casos	54
3.1.4 Clases definidas para el módulo de recuperación.....	55
3.1.5 Clases definidas para el módulo de adaptación.....	57
3.2 Validación	57
3.2.1 Aplicación de la técnica de ladov	57
3.2.2 Aplicación de la técnica Validación Cruzada (Cross-Validation).....	60
3.2.3 Aplicación de las pruebas unitarias.....	61
Conclusiones parciales	63
Conclusiones	64
Recomendaciones	65
Referencias bibliográficas.....	66
Anexos.....	75

Introducción

La enseñanza es el proceso mediante el cual se comunican o transmiten conocimientos especiales o generales sobre una materia. Este concepto es más restringido que el de educación, ya que la misma tiene por objeto de estudio la formación integral de la persona, mientras que la enseñanza se limita a transmitir, por diversos medios, determinados conocimientos (Edel Navarro, 2015).

Las tendencias actuales de la sociedad están cada vez más influenciadas por el uso de las Tecnologías de la Información y las Comunicaciones (TIC), de este modo por ejemplo, con la difusión del Internet ha hecho posible que los usuarios de la gran red puedan compartir información y conocimientos sin importar su localización geográfica, idioma o cultura. La enseñanza no ha quedado exenta de estas tendencias y en este sentido, ha evolucionado para potenciar el aprendizaje a distancia y la interacción más allá del hecho de compartir un mismo espacio físico, dando paso al surgimiento de los Recursos Educativos Abiertos (REA).

Los REA son recursos materiales gratuitos que se encuentran disponibles libremente en internet (UNESCO, 2002), expresión de una corriente mundial llamada *Open Access* (en español: acceso abierto) y de esta forma tienen como premisa fundamental la de extender su alcance educativo a toda la población de las naciones del mundo.

Dentro del Proceso de Enseñanza Aprendizaje (PEA) los REA tienen como propósito contribuir a la formación del estudiante a través del cumplimiento de objetivos instructivos y educativos, favoreciendo la adquisición de conocimientos y tributando a que el alumno desarrolle habilidades y estrategias para desenvolverse adecuadamente en las disímiles situaciones de aprendizaje. Con su uso también se puede contribuir a la participación activa de los propios estudiantes en la creación de contenidos digitales, ampliándose la transmisión de información y conocimientos y propiciando nuevas estrategias educativas, en las cuales el papel central radica en el educando y se torna algo menos activo el del profesor.

El *e-learning*, o aprendizaje a distancia, proporciona la oportunidad de crear ambientes de aprendizaje centrados en el estudiante. Estos escenarios se caracterizan además por ser interactivos, eficientes, fácilmente accesibles y distribuidos (M. Boneu, 2007). La adecuada integración de estos escenarios en los

centros educativos como una herramienta más al servicio de sus objetivos, puede significar grandes fortalezas de comunicación, creatividad y colaboración para el proceso de enseñanza aprendizaje, los alumnos, docentes y la institución educativa (Romero y otros, 2012).

Debido a la creciente demanda de tener a disposición cantidades considerables de REA, muchos profesionales e instituciones se han dado a la tarea de especializarse cada vez más en su proceso de revisión. Sin embargo aún se manifiestan muchos problemas en sus diseños, como la selección de contenido incorrecto, presencia de un nivel de detalle inadecuado o la selección de colores, imágenes o sonidos que interfieren con los objetivos de aprendizaje.

En los últimos años se ha creado un número importante de herramientas informáticas que apoyan la gestión de los REA. Entre estas se encuentran los repositorios digitales, los cuales permiten la preservación y reutilización del contenido, su acceso permanente, así como mayor visibilidad y facilidad de las búsquedas (Charnelli, 2013). Algunos ejemplos de estos repositorios son el Temoa, el CREA y el RHODA, este último desarrollado por el centro de desarrollo FORTES de la Universidad de las Ciencias Informáticas.

Por otro lado se conoce que los Sistemas Expertos (SE) son algunas de las herramientas que apoyan en la actualidad la gestión de los REA. Los avances de la Inteligencia Artificial (IA) en la educación están otorgando a profesores y estudiantes nuevas oportunidades donde el proceso educativo, respetando la diversidad, permita la adquisición de competencias que han sido difíciles de alcanzar en la educación tradicional, tales como: aprendizaje autodirigido, gestión del propio conocimiento, automotivación y autodirección. La inteligencia artificial como disciplina científica y el desarrollo tecnológico que se ha impulsado en el campo de los SE, ha abierto una nueva gama de posibilidades a docentes y alumnos en el marco de un modelo educativo centrado en el aprendizaje y no en la enseñanza (Vílchez Quesada, 2007).

En el ámbito educativo los sistemas expertos presentan ciertas ventajas, en particular los creados con fines pedagógicos e instruccionales. Un sistema experto instruccional diagnostica, depura y corrige la ejecución de los estudiantes en un área particular de conocimiento. El sistema determina el nivel cognoscitivo del alumno y lo ayuda a mejorar sus debilidades para que alcance un nivel superior de aprendizaje (Vílchez Quesada, 2007).

La utilización de los REA en el PEA es una práctica que cada vez ha tomado mayor fuerza en los entornos educativos, impulsando la expansión de un nuevo modelo pedagógico que sumado al uso de la inteligencia artificial, promueve que estudiantes y profesores usen, compartan y reutilicen sus recursos con el resto de la comunidad educativa.

Los repositorios digitales generalmente implementan sistemas de revisión que son capaces de identificar los REA que presentan dificultades, pudiendo tener problemas en su diseño y/o el contenido que muestran y que por ende necesitan ser rediseñados (Cañizares González, 2012). Aún sabiendo cuáles son estos REA defectuosos, a los diseñadores del aprendizaje se les hace en ocasiones difícil, en el proceso de revisión, identificar cuáles son los posibles problemas, dificultándose su corrección y la creación de patrones que eviten la reiteración de los mismos. Este proceso es importante en el ciclo de vida de un recurso educativo, ya que al ser rediseñados se aprovecha el carácter flexible y modificable de estos, para indagar en un aumento de su calidad a fin de que puedan cumplir de la mejor manera posible su función educativa.

Dada la anterior **situación problemática** surge el siguiente problema a resolver: ¿Cómo identificar problemas en los diseños de Recursos Educativos Abiertos defectuosos?

Teniendo en cuenta lo anteriormente planteado y para dar solución a la problemática enunciada, se define el **objeto de estudio**: proceso de revisión de Recursos Educativos Abiertos; centrándose en el **campo de acción**: sistema experto para la revisión de Recursos Educativos Abiertos.

Como **objetivo general** se establece: desarrollar un sistema experto para la identificación de problemas en los diseños de Recursos Educativos Abiertos defectuosos. Se plantea como **hipótesis**, que el desarrollo de un sistema experto para la identificación de problemas en los diseños de Recursos Educativos Abiertos defectuosos, facilitará el proceso de revisión de dichos recursos.

Para dar cumplimiento al objetivo general se establecen los siguientes objetivos específicos:

1. Analizar los principales elementos relacionados con la definición de Recursos Educativos Abiertos y los problemas más comunes que se cometen en su diseño.
2. Analizar sistemas expertos utilizados a nivel internacional y nacional en diversas esferas de la sociedad.

3. Implementar un sistema experto para la identificación de problemas en los diseños de Recursos Educativos Abiertos defectuosos.
4. Validar la propuesta de solución una vez finalizada su implementación.

Entre los resultados que se esperan con el desarrollo de la presente investigación se encuentra: la obtención de un sistema experto que sirva de herramienta a los diseñadores del aprendizaje en el proceso de revisión de los Recursos Educativos Abiertos defectuosos, permitiendo a través del uso de técnicas de la inteligencia artificial, la identificación de problemas en los diseños de los mismos.

Para el desarrollo de la investigación se utilizaron los métodos listados a continuación.

Métodos Teóricos

Análisis histórico-lógico: se utiliza para realizar la valoración acerca de la evolución de las tecnologías en el campo de la educación vinculada a los sistemas expertos existentes en la actualidad, potenciando el conocimiento sobre su funcionamiento y estructura.

Analítico-sintético: se emplea en el estudio de las tecnologías viendo la relación existente entre cada una de ellas y el funcionamiento de las mismas por sí solas para el desarrollo de la investigación.

Métodos Empíricos

Entrevistas: dentro de la entrevista es empleada la no estructurada, en el momento de hacer investigaciones con expertos en el trabajo con Recursos Educativos Abiertos.

El documento está estructurado por tres capítulos, siendo estos:

Capítulo 1: Fundamentación teórica de la investigación, incluye un análisis del estado del arte en Cuba y el mundo sobre la utilización de los Recursos Educativos Abiertos y los sistemas expertos en el campo de la educación. Se hace una descripción de las tendencias, técnicas, tecnologías, herramientas y algoritmos usados para el desarrollo del sistema experto y se justifica la utilización de las tecnologías para su implementación.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA, describe las tecnologías seleccionadas para la realización del sistema y una explicación detallada de los componentes que lo integran.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA, en este capítulo se describen los estándares de programación y codificación utilizados, se detallan los diagramas UML utilizados para el desarrollo de la solución y se muestran los resultados de las pruebas de validación realizadas.

Capítulo 1: Fundamentación teórica de la investigación

El objetivo fundamental de este capítulo es abordar los aspectos más relevantes que conforman el marco teórico de la investigación y que contribuirán al diseño e implementación de SISDREA. En este se describen los conceptos principales para comprender el basamento teórico de la investigación y se analizan los tipos de sistemas expertos y tecnologías utilizadas en su implementación, que pueden servir como referencia para la elaboración de la solución.

1.1 Recursos Educativos Abiertos y Objetos de Aprendizaje

La posibilidad de interconexión existente en la actualidad gracias a Internet, al mismo tiempo que posibilita consultar información casi de cualquier tipo en la web, también ha traído consigo la necesidad de centralizar y estandarizar los recursos y herramientas disponibles con el objetivo de optimizar su utilidad y usabilidad y facilitar las búsquedas y consultas de los usuarios.

La educación no ha quedado exenta de estos cambios, vista como el conjunto de conocimientos, órdenes y métodos por medio de los cuales se ayuda al individuo en el desarrollo y mejora de las facultades intelectuales, morales y físicas (Edel Navarro, 2015); se ha permitido evolucionar para dar surgimiento a términos como Objetos de Aprendizaje y Recursos Educativos Abiertos.

1.1.1 Objeto de Aprendizaje

El término de OA ha ido evolucionando a medida que se ha difundido su aplicación y uso (estrechamente relacionado con la evolución tecnológica), por estas cuestiones se ha hecho complicado converger en una definición única.

Algunos autores lo precisan como materiales educativos en pequeñas unidades que permiten el aprendizaje de una forma sencilla y que pudieran conectarse entre sí, desarrollando piezas de aprendizaje fácilmente interoperables. Autores como (Barberá Gregori y otros, 2005) lo definen como: “entidad, digital o no digital, que puede ser utilizada, reutilizada o referenciada durante el aprendizaje con soporte en la tecnología”.

Los OA se introducen en el ámbito educativo no necesariamente como un recurso digital, sin embargo a partir de su combinación con las TIC es que precisamente cobra fuerza la idea de lograr unidades de

aprendizaje autocontenidas, interoperables, reutilizables, durables y actualizables (Serrano Islas y otros, 2013).

1.1.2 Recursos Educativos Abiertos. Definición e importancia

Por otro lado REA es un término muy similar al de OA pero que más específicamente es expresión de una tendencia mundial llamada movimiento Open Access (acceso abierto). De esta manera está más enfocado a la libre y abierta disponibilidad de los mismos, teniendo como premisa fundamental extender su alcance educativo a toda la población de la naciones del mundo, enriqueciendo el desarrollo cultural de los pueblos (Celaya Ramírez y otros, 2010).

La siguiente imagen describe los REA más detalladamente:



Figura 1: Recurso Educativo Abierto (Cañizares González, 2012)

Una gran cantidad de recursos educativos se encuentran hoy a disposición de escolares y educadores en los centros educacionales. Influenciado por su impacto positivo, se ha originado una disposición hacia su apertura y democratización.

Capítulo 1: Fundamentación teórica de la investigación

Mediante el uso de REA los estudiantes pueden obtener información abundante en varios puntos de vista, los trabajadores pueden aprender algo que les ayudará en el trabajo, los investigadores pueden compartir datos y desarrollar nuevas líneas de investigación, los profesores pueden encontrar nuevas y atractivas maneras de ayudar a sus estudiantes a aprender, sirviéndoles como medio para el desarrollo de competencias o manifestaciones de apropiación que le permitan al educando trascender más allá del rol de un usuario común. Las personas pueden conectarse con otras que se encuentran a grandes distancias para compartir información e ideas, los materiales pueden ser traducidos, combinados, reutilizados y recompartidos.

El intercambio de conocimiento ha sido y es la base de la educación. La educación abierta combina esta característica básica de la enseñanza con las posibilidades inmensas que ofrece Internet para crear una extraordinaria oferta de recursos educativos compartidos que respondan a las necesidades de los alumnos (Cooperman, 2014).

Un progreso adicional de los REA, además de encontrarse públicos para su acceso y utilización, es que muchas instituciones a nivel internacional se estén dando a la tarea de optimizar y facilitar la consulta de los mismos a través de repositorios que intentan garantizar búsquedas efectivas permitiendo el acceso a materiales de alta calidad, confiabilidad y legalidad (Celaya Ramírez y otros, 2010).

1.2 Repositorios digitales de REA

Para facilitar el acceso a los REA se utilizan los repositorios digitales que permiten la preservación y reutilización de contenido, acceso permanente, mayor visibilidad, facilidad de la búsqueda y recuperación mediante el uso de etiquetas y clasificadores. A su vez los repositorios digitales son abiertos e interactivos pues cumplen con protocolos internacionales que permiten la interoperabilidad entre ellos (Charnelli, 2013).

Dentro de los repositorios digitales educativos se encuentran contenidos, digital y virtualmente, los recursos educativos públicos al Internet. Estos residen en servidores específicos y utilizan determinados estándares y protocolos para su acceso. Son espacios especializados donde exclusivamente se pueden encontrar recursos educativos digitalizados, como pueden ser: Recursos Educativos Abiertos, Objetos de Aprendizaje, programas/software y otro tipo de aplicaciones. Pueden ser de distintos tipos dependiendo del área del conocimiento que traten (Mortera Jiménez, 2013).

Capítulo 1: Fundamentación teórica de la investigación

Son varias las universidades que se han sumado al movimiento de acceso abierto creando repositorios institucionales para albergar y difundir su producción académica. Estos repositorios suelen ser híbridos, o sea no exclusivamente de REA, sino con otros materiales de investigación y docencia. Dentro de estos los REA no tienen el objetivo de difundir los resultados de estudios investigativos, sino de ser útiles dentro del proceso de enseñanza-aprendizaje. Elementos como la granularidad, reutilización, actualización e intercambio de REA; son claves para el éxito de estos repositorios.

En su afán por expandir el alcance y la utilidad de los recursos educativos que almacenan y su repercusión en el proceso de enseñanza aprendizaje y en la cultura de los pueblos de forma general, la mayoría de los repositorios publican, mediante estándares de interoperabilidad, servicios que permiten la obtención de información sobre los recursos que almacenan.

Para el desarrollo de la presente investigación se utiliza el repositorio RHODA (ROA, 2010) como repositorio de referencia para la adquisición de la información sobre los REA, consumiendo los servicios web que este publica. A continuación se ofrece una descripción sobre dicho repositorio.

1.2.1 Repositorio RHODA

En la Universidad de las Ciencias Informáticas se encuentra el repositorio de recursos educativos RHODA, herramienta informática desarrollada con tecnologías libres. El mismo posee un sistema integral de revisiones basado en procesos personalizables para las Instituciones de la Educación Superior (IES), que combina estándares de interoperabilidad para lograr la comunicación con sistemas externos (LMS, herramientas de autor y otros repositorios).

El repositorio RHODA contiene un transformador de estándares de catalogación en forma de librería desarrollada en php e integrado al framework Symfony, que puede ser reutilizado por otros sistemas en la catalogación de recursos.

Este repositorio utiliza el estándar LOM para la catalogación de metadatos y publica servicios de recolección de los mismos usando el estándar OAI-PMH sobre el protocolo REST. Ofrece además servicios web para obtener las evaluaciones de los REA que almacena. De esta forma se puede concluir que RHODA brinda la información necesaria que requerirá el sistema resultado de la presente investigación.

1.2.2 Estándar LOM (Learning Object Metadata) para la catalogación de metadatos

LOM es un modelo de datos, usualmente codificado en XML, usado para describir un objeto de aprendizaje y otros recursos digitales similares empleados para el apoyo al aprendizaje. El propósito de este estándar es ayudar a la reutilización de los recursos educativos y facilitar la interacción entre ellos, usualmente en el contexto de sistemas de aprendizaje on-line (Salvat, 2011).

Los metadatos son información sobre un objeto, sea éste físico o digital. Al tiempo que el número de objetos continúa creciendo exponencialmente y que las necesidades de aprendizaje aumentan igualmente, la falta de información o de metadatos sobre los objetos limita de manera fundamental y crítica la capacidad para la búsqueda, la gestión y el uso de recursos. Este estándar afronta el problema definiendo una estructura para la descripción interoperable de recursos educativos (Carlos C. y otros, 2002).

Estructura

El estándar LOM está dividido en nueve categorías donde cada una está compuesta por varios elementos de metadatos que describen un objeto educativo, las mismas se describen a continuación (Carlos C. y otros, 2002):

- a) *General*: agrupa la información general que describe un objeto educativo de manera global.
- b) *Ciclo de vida*: agrupa las características relacionadas con la historia y el estado actual del objeto educativo y aquellas que le han afectado durante su evolución.
- c) *Meta-Metadatos*: agrupa información sobre la propia instancia de metadatos (en lugar del objeto educativo descrito por la instancia de metadatos).
- d) *Técnica*: agrupa los requerimientos y características técnicas del objeto educativo.
- e) *Uso educativo*: agrupa las características educativas y pedagógicas del objeto.
- f) *Derechos*: agrupa los derechos de propiedad intelectual y las condiciones para el uso del objeto educativo.
- g) *Relación*: agrupa las características que definen la relación entre este objeto educativo y otros objetos educativos relacionados.
- h) *Anotación*: permite incluir comentarios sobre el uso educativo del objeto e información sobre cuándo o por quién fueron creados dichos comentarios.

- i) *Clasificación*: describe este objeto educativo en relación a un determinado sistema de clasificación.

El uso del estándar LOM por el repositorio RHODA posibilita la obtención de los metadatos de los REA aprovechando su codificación en XML y uno de sus principales propósitos: la reutilización de los recursos educativos.

1.2.3 Sistemas de revisión en los repositorios digitales

Los repositorios digitales suelen contener una gran cantidad de recursos educativos de disímiles temas, enfoques y formatos; dirigidos a usuarios de diferentes edades y perfiles educativos. Es por esto que a la hora de realizar una selección de un determinado recurso, por ejemplo por un profesor, se puede encontrar algunos materiales de contenido pobre o incorrecto, o que dada su estructura o diseño no estén acordes con las edades y los perfiles de los alumnos ni con el modelo educativo de su institución.

Para solventar esta problemática una posible solución es aplicar un sistema de revisión a los recursos educativos antes de ser publicados en el repositorio. Entiéndase por revisión a los procesos que se llevan a cabo en cualquier ámbito donde se necesite garantizar la calidad de “algo”, en otras palabras, es la evaluación y selección de qué información aceptar para publicar y cuál no, basándose en el nivel de calidad que esta posea (Cañizares González, 2012).

Existen varios métodos o formas para realizar revisiones, que son aplicadas en dependencia de los intereses y/o características de las instituciones y los recursos educativos que estas manejan. Entre estos métodos se encuentran: *revisión por pares o revisión entre pares* (Kuramoto, 2010), *evaluación recíproca* (Pereira y otros, 2010) y *revisión colaborativa pos-publicado* (Cañizares González, 2012).

Actualmente hay otras revisiones que se especializan en determinados instrumentos de evaluación, como el “Modelo de participación convergente para la evaluación de OA” (Morales, 2010) que se apoya exclusivamente del instrumento LORI, sobre el cual se estará indagando en el próximo epígrafe.

Los sistemas de revisión, luego de procesar los resultados de las revisiones, permiten que los repositorios puedan generar un informe de los REA que presentan dificultades. Estos REA necesitarán ser rediseñados por sus autores para que puedan contar con una mejor calidad y utilidad para ser presentado a los usuarios.

1.2.4 Instrumentos de revisión de recursos educativos

Para facilitar la integración en los repositorios de los procesos de revisión y evaluación de los recursos, se hace necesaria la determinación de indicadores, parámetros o elementos que puedan ser aplicados de forma común a cualquier recurso a fin de que su calidad pueda ser evaluada de la forma más eficiente posible (Codina, 2000).

Estas metodologías, indicadores o criterios constituyen los instrumentos de evaluación. En el Anexo 1 se podrán encontrar algunos de estos instrumentos con los respectivos indicadores que definen. Dado que se basan en criterios genéricos, la selección de un instrumento u otro (o alguna combinación de varios) depende de las características específicas de la institución que desea aplicarlos.

En la presente investigación se determina usar LORI como instrumento de evaluación de referencia ya que los indicadores que propone se consideran los más completos para evaluar aspectos del REA desde el punto de vista formativo y de diseño (Toll Palma y otros, 2011). El repositorio RHODA permite utilizar el instrumento LORI para realizar las revisiones a los recursos educativos.

Los indicadores de evaluación que propone este instrumento conllevan a su vez una serie de posibles problemas que pudieran atentar contra la integridad de cada uno de los propios indicadores, dichos problemas se encuentran definidos en el Manual de Usuario del Instrumento LORI (Nesbit y otros, 2003) y son expuestos en el Anexo 2. SISDREA utilizará estos problemas, junto a la categoría más afectada, como salidas de recomendación.

1.3 Interoperabilidad

Autores como (Izquierdo, 2008), (de Vergara, 2001) y (Risso Patrón, 2003) abarcan el concepto de interoperabilidad, por lo que se puede resumir como la capacidad de dos o más sistemas o componentes para intercambiar la información y utilizarla.

1.3.1 REST (Representational state transfer)

Un servicio web es un componente de software independiente de plataforma e implementación que lleva a cabo una acción en concreto y que puede integrarse con otros servicios. Se trata de un recurso residente

en la web, con una dirección URL accesible y que devuelve información al cliente que quiera utilizarlo, pero los detalles de implementación y despliegue del servicio web no son relevantes para el programa que lo invoca (Peris Soler y otros, 2004).

REST es un estilo de arquitectura de software para sistemas hipermedias distribuidos tales como la web y se refiere estrictamente a una colección de principios para el diseño de arquitecturas en red. Estos principios resumen cómo los recursos son definidos y diseccionados. El término es frecuentemente utilizado en el sentido de describir a cualquier interfaz que transmite datos específicos de un dominio sobre HTTP sin una capa adicional. REST no es un estándar sino un estilo de arquitectura aunque está basado en estándares tales como el protocolo *Hypertext Transfer Protocol* (HTTP) y *Uniform Resource Locator* (URL) (Navarro Maset, 2007).

REST presenta algunas características que hacen de él una buena opción para realizar la conexión con los repositorios de REA, algunas de dichas características son:

- ✓ Es muy ligero, sus respuestas contienen exactamente la información que se necesita.
- ✓ Los servicios RESTful proporcionan una buena infraestructura de almacenamiento en caché a través de HTTP y el método GET.
- ✓ Son servicios fáciles de integrar con los sitios web existentes y están expuestos para que las páginas HTML los puedan consumir con la misma facilidad.

A través del servicio REST serán extraídos los REA del repositorio RHODA.

1.3.2 XML (eXtensible Markup Language)

XML es un lenguaje de marcado extensible que se está convirtiendo en un estándar de amplio uso para el intercambio de datos. A diferencia del HTML, separa el contenido de la presentación. Es un lenguaje abstracto que está definido por lo que se denomina etiquetas, que se especifican entre los símbolos < y >, de la siguiente forma: <etiqueta> (Castro, 2015).

XML sirve para definir, validar y compartir formatos de documentos en la web. La diferencia fundamental entre HTML y XML es que el primero estaba orientado a la presentación de datos, mientras que XML está orientado a los datos en sí mismos, por lo que cualquier software informático trabajará mejor con XML. Sin

duda, esta diferencia es fundamental para los nuevos desarrollos de la Web donde se da suma importancia al contenido de los datos y su tratamiento y no sólo a su presentación.

El lenguaje XML sirve como base para el tratamiento de los metadatos de los REA que se obtendrán una vez consumido el servicio web REST publicado por el repositorio RHODA.

1.4 Sistemas Expertos

Los SE han encontrado aplicaciones en múltiples esferas de la sociedad, convirtiéndose en una herramienta importante capaz de resolver problemas de manera rápida, eficiente y segura. A continuación se brindan algunas definiciones de SE.

“Un Sistema Experto es un sistema computacional que adquiere conocimiento especializado en un campo específico para explotarlo mediante métodos de razonamiento que emulan el desempeño del experto humano en la solución de problemas” (Peña Ayala, 2006). Según (Begaña Vitoriano y otros, 2003) es un “programa que resuelve problemas que requieren experiencia humana, mediante representación del conocimiento y procedimientos de decisión”.

Por su parte (Velázquez López, 2010) define como SE "un programa inteligente de computador que utiliza conocimiento y procedimientos de inferencia para resolver problemas que son lo suficientemente difíciles y requieren experticia humana significativa para su solución. El conocimiento de un Sistema Experto consiste en hechos y heurísticas". Se asume este último concepto como referencia para la presente investigación pues se considera el más completo y actual para describir los SE.

Entre los campos donde se aplican los SE destacan la medicina, las finanzas, la industria y la electrónica (Peña Ayala, 2006); otros autores se concentran en la aplicación de los SE en la educación como (Cortinas Bernal y otros, 2013).

La inclusión de los sistemas expertos en la educación es una forma de transmitir el conocimiento, intercambiando las experiencias obtenidas con la práctica a lo largo de los años, por ello no solo se habla de educación con tecnología, sino de un sistema experto que esté enriquecido con las experiencias de colegas que aporten conocimientos y experiencias de enseñanza.

1.4.1 Estructura de un Sistema Experto

Los SE suelen tener una estructura común (Pérez Bello y otros, 2002):

- ✓ *Base de conocimiento*: contiene los conocimientos relativos a la tarea. Usa formalismos de representación para codificar los conocimientos en la base.
- ✓ *Motor o máquina de inferencia*: medio por el cual controla y aplican los conocimientos. Permiten que el sistema razone a partir de los datos, noticias o conocimientos de entrada para producir los resultados de salida. Gobiernan el orden en el que el sistema realiza los pasos de razonamiento, acepta entradas y produce salidas.
- ✓ *Interfaz de Entrada/Salida (E/S)*: para que el usuario proporcione hechos y datos y el sistema responda.

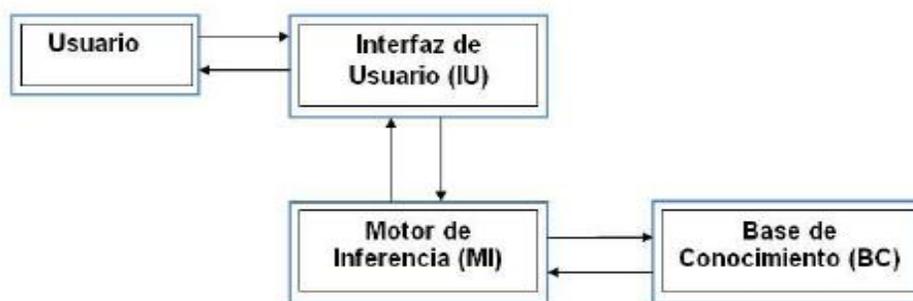


Figura 2. Esquema Básico de un Sistema Experto (Pérez Bello, 2002)

1.4.2 Tipos de búsqueda en los Sistemas Expertos

Durante la recuperación de los casos se necesita realizar un proceso de búsqueda el cual brindará como resultado el o los casos que podrán ser tomados como solución al problema. Según la forma de realización de las búsquedas estas se clasifican en búsquedas por analogía o por semejanza.

- **Búsqueda por analogía**

La búsqueda por analogía pretende determinar si la descripción del problema es equivalente a la descripción de alguno de los casos almacenados, es decir, probar la equivalencia entre los valores de un rasgo, aunque sean diferentes si semánticamente tienen igual significado. Para este análisis se utiliza una red semántica

en la cual los nodos representan valores y rasgos y los enlaces indican el grado de relación entre ellos. Dos valores son equivalentes si están enlazados con el mismo concepto y tipo de relación, a partir de un igual punto de vista.

- **Búsqueda por semejanzas**

En la búsqueda por semejanzas, el objetivo es obtener el caso almacenado que más se parece al nuevo problema. Existen diversas formas de medir el grado de semejanza, que va desde expresiones analíticas de distancia hasta la descripción de algoritmos para obtener el grado de parecido entre dos objetos. Esto depende del tipo de información que se presente (Rodríguez, 2003).

En este tipo de búsquedas las funciones de semejanza indican cómo se deben comparar los casos, pero es necesario saber qué se debe comparar. De todos los atributos que conforman un caso, es necesario definir cuáles son los más relevantes para efectuar la comparación (Martínez Sánchez y otros, 2008). Existen varias funciones de semejanzas que han sido propuestas para recuperar casos semejantes, muchas de ellas trabajan con atributos numéricos y no son apropiadas para atributos simbólicos; otras permiten buscar semejanza en una estructura determinada. Por ejemplo, la distancia Euclidiana es utilizada para los atributos numéricos o aquellos atributos nominales a los que se les pueda asignar un orden.

1.4.3 Clasificación de los Sistemas Expertos

En la clasificación de los SE más utilizados se encuentran los Sistemas Basados en Reglas y los Sistemas Basados en Casos. A continuación se brinda una breve descripción.

1.4.3.1 Sistemas Basados en Reglas

Los Sistemas Basados en Reglas (SBR) son un tipo de SE que utiliza como forma de representación del conocimiento las reglas de producción. Las reglas utilizan un formato SI – ENTONCES (IF – THEN) para representar el conocimiento, la parte SI (IF) de una regla es una condición o premisa y la parte ENTONCES (THEN) de la regla es la acción o conclusión, la cual permite inferir un conjunto de hechos nuevos si se verifican las condiciones de la parte SI (IF).

El procedimiento de solución del problema utilizado por los SBR sigue el método conocido como Modus Ponens, creando una cadena de inferencias que constituye un camino entre la definición del problema y la

solución. Este camino normalmente es creado por una *búsqueda primero en profundidad* dirigido por objetivo o datos en dependencia de la cantidad de datos de entrada y de conclusiones (Pérez Bello, 2002).

1.4.3.2 Sistemas Basados en Casos

Los Sistemas Basados en Casos (SBC) son una de las técnicas de Inteligencia Artificial más empleadas en la actualidad en los procesos de toma de decisiones, son un tipo de SE que utiliza los casos como forma de representación del conocimiento y emplean el Razonamiento Basados en Casos (CBR) como mecanismo de inferencia.

El CBR es un proceso de considerar casos anteriores y entonces llegar a una decisión en dependencia de la comparación entre la situación actual y los casos viejos. Dado un nuevo caso para clasificar, un razonador basado en casos verifica primero si existe un caso de entrenamiento idéntico. Si encuentra uno, retorna la solución acompañando a ese caso. Si no encuentra ningún caso idéntico, el razonador buscará los casos de entrenamiento que tienen componentes similares a aquellas del nuevo caso. El sistema es capaz de aprender a través de la agregación de nuevos casos (Pérez Bello, 2002).

1.4.4 Sistema Basado en Casos y búsqueda por semejanza. Justificación de su selección

Los SE son utilizados generalmente para la predicción, por lo que se definió usar un SBC debido a que no existe un dominio exacto de la información relacionada con los REA y la implicación que tiene la información contenida en sus metadatos en posibles problemas en su diseño. Además existe una base de información en el proyecto RHODA con casos (recursos educativos) ya almacenados, la cual se podría utilizar como base de conocimiento para inferir la solución. El SBC estará restringido a las variaciones sobre situaciones conocidas y a partir de estas producirá recomendaciones que estarán en correspondencia con la similitud entre el caso de entrada y los recursos recuperados.

Para la recuperación de los casos se utiliza el tipo de búsqueda por semejanza ya que se emplearán fórmulas analíticas para calcular la semejanza entre el caso recibido como entrada y el resto de los casos de la base de casos.

1.5 Sistemas Basados en Casos

Los SBC están compuestos por tres componentes fundamentales: la Base de Conocimientos (BC), la Máquina de Inferencia (MI) y la Interfaz de Usuario (IU). En la BC se almacena el conocimiento necesario para resolver los problemas del dominio de aplicación atendiendo a una forma de representación del conocimiento. La MI es un procedimiento basado en un esquema de razonamiento o método de solución de problemas que utiliza el conocimiento para resolver los problemas de ese dominio y la IU permite la comunicación entre el usuario y el ordenador (Pérez Bello, 2002).

1.5.1 Ciclo del razonamiento basado en casos

A continuación se representa el funcionamiento del ciclo de un CBR, dándose a conocer las definiciones para cada una de las etapas del proceso:

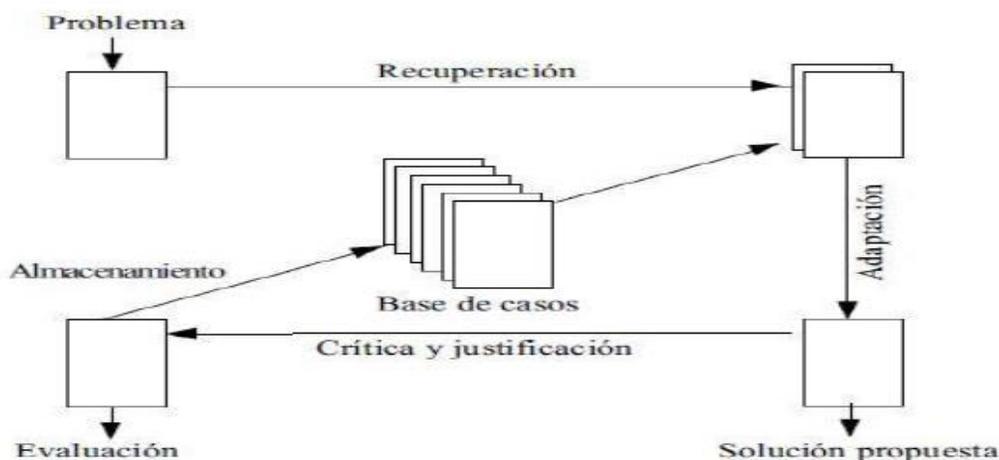


Figura 3: Ciclo de vida de un CBR (Vega y otros, 2000)

- ✓ La *recuperación* de casos es la selección en la base de conocimiento, de aquellos cuya descripción se ajusta más a la información presentada en el nuevo caso.
- ✓ La *adaptación* consiste en adecuar la solución del caso más parecido a las condiciones del nuevo caso. Esto es necesario dado que normalmente los fenómenos o síntomas que se presentan en un diagnóstico no son idénticos a los ocurridos en los casos anteriores.

- ✓ La etapa de *crítica y justificación* consiste en la validación de la solución propuesta. Esta validación se realiza contrastando diferentes soluciones o simulando la solución para estimar qué tan acertada es. Esta etapa está altamente influenciada por el grado de conocimiento que tiene el experto sobre el fenómeno ocurrido y quién juzga la efectividad de la solución propuesta en base a su experiencia.
- ✓ En la etapa de *evaluación* se aplica la solución propuesta y se analiza el resultado de su aplicación. Si los resultados son los esperados se confirma la solución, pero si existen diferencias, se debe averiguar por qué ocurrieron las mismas y cómo pueden evitarse. Esta información debe servir para mejorar la definición del caso.
- ✓ La etapa de *almacenamiento* consiste en registrar, en la base de conocimiento, la información derivada del nuevo caso, ya sea como un caso nuevo o un caso mejorado.

1.5.2 La Recuperación en el Razonamiento Basado en Casos

Dependiendo de la situación, uno de los componentes más importantes del RBC suele ser la recuperación. Este proceso puede dividirse en dos etapas: Selección y Recuperación. La primera se encarga de seleccionar, entre todos los casos almacenados, cuáles son los candidatos que pueden servir para resolver el nuevo problema. La segunda etapa se encarga de recuperar dentro de esos casos seleccionados, cuáles son los más parecidos al nuevo problema de forma tal que pueda utilizarse la solución dada a ellos (Agudo, 2001).

Una vez que se cuenta con el conjunto de casos candidatos, la recuperación del caso que mayor semejanza tiene con el problema puede producir dos resultados: existe un caso que coincide completamente con el patrón de búsqueda, o varios casos que difieren al menos un rasgo con respecto al patrón de búsqueda. El primer caso significa que el problema estaba almacenado en la base de casos y se tiene la solución. En el segundo, existen dos alternativas, una es realizar la búsqueda por analogía y la otra determinar un valor numérico que indique el grado de semejanza entre el problema y cada caso candidato (Agudo, 2001).

1.5.3 Algoritmos de recuperación de casos

Dentro de los algoritmos que se usan para la recuperación de casos predominan los de minería de datos. Estos proporcionan un conjunto de cálculos y reglas, con el fin de analizar los datos y extraer patrones específicos o tendencias.

Los algoritmos de recuperación más investigados son: K-vecinos más cercanos y árboles de decisión y sus derivados. Estas técnicas necesitan de una medida de similitud para determinar la proximidad entre casos, por ejemplo la distancia euclídea, de hamming o de levenshtein.

1.5.3.1 K-Nearest Neighbor (KNN)

El KNN (o K-Vecinos más Cercanos) es una de las técnicas que permite realizar clasificación a partir de un conjunto de datos sin necesidad de construir modelos previamente. Es uno de los algoritmos basados en distancia más conocido y utilizado, su función consiste en clasificar una instancia de la clase más fuertemente representada en sus K vecinos más próximos. De esta forma se recuperan los K primeros casos que se consideren más cercanos al caso base teniendo en cuenta las funciones de similitud entre rasgos y entre objetos que se definan. Las características o rasgos de cada objeto o caso, poseen un peso en correspondencia con su nivel de importancia en la situación (Molina López y otros, 2006).

Una típica función de evaluación empleada para encontrar el vecino que mejor emparejan es la mostrada a continuación:

$$\text{similarity}(Case_I, Case_R) = \frac{\sum_{i=1}^n w_i \times \text{sim}(f_i^I, f_i^R)}{\sum_{i=1}^n w_i}$$

Fórmula 1: Fórmula usada por el método KNN para encontrar al vecino más cercano

Donde w_i es el peso de importancia de una característica, sim es la función de similitud de la misma y f_i^I y f_i^R son los valores para la característica i de la entrada y del caso recuperado respectivamente.

1.5.3.2 Árboles de decisión

Los árboles de decisión se construyen a partir de la utilización de la técnica de aprendizaje inductivo para extraer reglas a partir de datos pasados. De esta forma, en los sistemas con razonamiento basado en casos (CBR), la base de casos es analizada para determinar qué características son las que mejor diferencian los casos entre sí para producir un árbol de decisión que clasifica o indexa los casos.

Un árbol de decisión es un conjunto de condiciones organizadas en una estructura jerárquica, de tal manera que la decisión final a tomar se puede determinar siguiendo las condiciones que se cumplen desde la raíz del árbol hasta alguna de las hojas. Dependiendo de los criterios de partición y otras extensiones, se han desarrollado varios algoritmos tales como: ID3, C 4.5, Cart y CN2 (Hernández Orallo y otros, 2004).

1.5.3.3 Comparación y selección del algoritmo

El método de los vecinos más próximos (KNN) tiene entre sus ventajas que permite aprovechar de manera más eficiente la información del conjunto de entrenamiento. Se caracteriza además por su robustez y efectividad en grandes conjuntos de datos. Su principal desventaja es la velocidad en la que se recuperan los casos ya que debe analizar toda la base de casos en busca del que mejor se ajusta. Esto quiere decir que una comparación de similitud (distancia) debe calcularse para cada característica indexada haciendo que el algoritmo sea ineficiente a medida que aumenta la base de casos.

Los árboles de decisión dependen de la preindexación ya que es construido antes de que la recuperación comience. Este proceso consume muchos recursos para una base de casos grande y tiene que repetirse cada vez que un nuevo caso es añadido a la base, sin embargo, los tiempos de recuperación son más rápidos. La principal desventaja de esta técnica es que si los datos de un caso se pierden o son desconocidos, no se puede recuperar el caso. El vecino más próximo es menos sensible a pérdidas.

SISDREA, está concebido para trabajar básicamente con los metadatos de los REA, constituyendo su principal fuente de información. Se ha analizado que la información contenida en los metadatos de los recursos generalmente es pobre e incompleta. Esto ocurre porque al ser creados los recursos educativos a sus autores muchas veces se les hace engorrosa la tarea de describir cada una de las especificaciones de metadatos descritas por LOM, restándosele además importancia al proceso de completamiento de metadatos, del que depende un futuro procesamiento automático e inteligente de los recursos.

Teniendo en cuenta lo anteriormente planteado y considerando además que la base de recursos (casos) a procesar es relativamente pequeña (actualmente solo 150 recursos), se decide utilizar el algoritmo KNN en el Módulo de Recuperación por las ventajas anteriormente mencionadas respecto a otros algoritmos.

1.5.4 Medidas de similitud

Los métodos basados en distancias parten del razonamiento por analogía, donde si se tienen problemas similares se pueden aplicar soluciones similares, siendo un caso particular del razonamiento basado en casos. Algoritmos como KNN, análisis de clúster o métodos de escalado multidimensional, aplican estos métodos en busca de obtener medidas similares entre objetos usando distancias.

El método KNN utiliza principalmente dos fórmulas para el cálculo de la distancia entre dos escenarios, estas son la Euclidiana o Euclídea y la de Manhattan.

1.5.4.1 Distancia Euclidiana

La distancia Euclidiana es la más común y usada en la implementación del método KNN. Se define distancia Euclidiana entre dos puntos X y Y como se muestra en la Fórmula 2, donde X_i es la coordenada de X en la dimensión i y Y_i es la coordenada de Y en la dimensión i .

$$d_E(x, y) = \sum_{i=1}^n \sqrt{x_i^2 + y_i^2}$$

Fórmula 2: Fórmula para determinar la distancia Euclidiana (Giménez Arjona, 2006)

1.5.4.2 Distancia de Manhattan

La distancia de Manhattan entre dos puntos es la suma de los valores absolutos de las diferencias de sus componentes. Se define la distancia de Manhattan entre un punto X y otro punto Y como se muestra en la Fórmula 3, donde X_i es la coordenada de X en la dimensión i y Y_i es la coordenada de Y en la dimensión i .

$$d_A(x, y) = \sum_{i=1}^n |x_i - y_i|$$

Fórmula 3: Fórmula para determinar la distancia de Manhattan (Giménez Arjona, 2006)

Para la implementación de SISDREA se considera factible la utilización de la distancia Euclidiana para determinar la similitud entre recursos educativos, ya que la misma suele ser efectiva al utilizarse para tipos de atributos nominales y numéricos.

1.5.5 Métodos de adaptación

Recuperados los casos más similares se efectúa el procedimiento de adaptación, que consiste en la modificación y combinación de las soluciones de los casos similares para formar una nueva solución. Es el proceso de reutilización de casos transforma una solución recuperada en una solución apropiada para el problema actual (Lozano y otros, 2005).

1.5.5.1 Reinstanciación

Este método de sustitución se usa cuando la similitud entre el caso actual y el recuperado (ó recuperados) es obvia y no hay otros tipos de restricciones o requerimientos impuestos para la solución. En otras palabras, el caso actual y el recuperado son netamente similares aunque ciertos roles tengan valores diferentes. El proceso a seguir sería tomar la solución recuperada y actualizar los roles que difieran por los nuevos.

1.5.5.2 Ajuste de parámetros

Para el ajuste de parámetros se modifican los de la solución de casos anteriores de acuerdo con las diferencias entre las descripciones de los casos en cuestión. Este método se emplea para interpolar los valores de una solución anterior a una nueva solución. Es decir, dada una solución previa y un caso nuevo que difiere en algún grado de dicha solución, esta se modifica para la extensión en que difieren los dos casos. Por ejemplo, en un posible sistema de justicia, un nuevo crimen semejante pero más cruel que uno anterior ya almacenado, requiere una sentencia más dura.

Es de mencionar que el ajuste de parámetros es una interpolación, es decir, no se debe confundir con otros métodos que sí crean verdaderas nuevas soluciones. Este método sólo construye una nueva solución a partir de una existente ajustando sus parámetros (Lozano, y otros, 2005).

1.5.5.3 Búsqueda local

Así como la reinstanciación sustituía un conjunto completo de roles, hay situaciones en las que sólo es necesaria la sustitución de una pequeña parte de una solución previa para que se adapte perfectamente al nuevo caso.

Se denomina búsqueda local al proceso de buscar en una jerarquía abstracta algo relativamente similar que pueda ser sustituido. Se suele usar cuando una solución previa es casi correcta para el caso nuevo y puede ajustarse con alguna sustitución menor. Así, para recorrer esa jerarquía mencionada, este método busca primero en los hermanos del elemento previo para ver si alguno sirve como sustitución. Si no tiene éxito busca en los primos y así sucesivamente. La búsqueda local debe restringirse puesto que de otro modo sería ineficiente. Además se suelen incorporar ciertas guías de movimiento hacia arriba y hacia abajo en la jerarquía.

En el presente trabajo se propone un nuevo enfoque donde se aplican los principios de la Lógica Difusa para realizar la adaptación.

1.5.6 CBR y Lógica Difusa (Fuzzy Logic)

La lógica difusa es una metodología que proporciona una manera simple y elegante de obtener una conclusión a partir de información de entrada vaga, ambigua, imprecisa, con ruido o incompleta. En general imita cómo una persona toma decisiones basada en información con las características mencionadas. Una de sus principales ventajas es la fácil integración con otras técnicas de la inteligencia artificial donde se maneje información con estas peculiaridades (D' Negri y otros, 2006).

Al ser esencialmente lógicas multivaluadas que extienden las lógicas clásicas, procuran crear aproximaciones matemáticas en la resolución de ciertos tipos de problemas, pretendiendo producir resultados exactos a partir de datos imprecisos. Mientras que en la teoría de conjuntos convencional un elemento pertenece o no a un conjunto, en un conjunto difuso su frontera no está precisamente definida, sino que se asigna un grado de pertenencia con un valor entre 0 y 1 a cada conjunto.

Capítulo 1: Fundamentación teórica de la investigación

El adjetivo “difuso” aplicado a ellas se debe a que los valores de verdad utilizados en ellas tienen, por lo general, una connotación de incertidumbre. Lo difuso puede entenderse como la posibilidad de asignar más valores a los enunciados que los clásicos “falso” o “verdadero”.

Algunas investigaciones como (Hansen y otros, 2000), (Fdez-Riverola y otros, 2003) y (Takagi y otros, 1985) combinan el razonamiento basado en casos y la metodología de lógica difusa para realizar la inferencia y producir salidas con un mayor nivel de exactitud. Por ejemplo (Hansen, y otros, 2000) propone una aplicación llamada WIND-1 que utiliza un método llamado Fuzzy-KNN para realizar predicciones meteorológicas con gran precisión, combinando el algoritmo KNN con los preceptos de la también llamada, Lógica Fuzzy.

1.6 Soluciones existentes a nivel nacional e internacional

En la Universidad de las Ciencias Informáticas (UCI) se han desarrollado SE que se especializan en esferas como la medicina, la educación, el soporte y la gestión de procesos, entre otros. Para la presente investigación se consultaron algunos de ellos:

- ✓ Sistema Basado en Casos para predecir la ocurrencia de reacciones adversas a medicamentos en la consulta médica (Corona Prendes y otros, 2013).
- ✓ Sistema Basado en Reglas para detectar interacción medicamentosa desde la consulta médica (Reyes García y otros, 2014).
- ✓ Sistema basado en casos para la asistencia al proceso de soporte y ayuda en la gestión de incidencias tecnológicas en la Universidad de Ciencias Informáticas (Cruz Rojas y otros, 2009).

Específicamente en el campo de la educación se han desarrollado sistemas a nivel internacional y nacional, cuyos objetivos están enfocados a la gestión de REA insertados en el PEA. Estos sistemas favorecen la recopilación, el acceso y el intercambio de recursos y ofrecen mecanismos flexibles que permite la búsqueda y recuperación de dichos recursos. Además facilitan el acceso a los metadatos, lo que posibilita el desarrollo de soluciones que integren estos elementos a arquitecturas y elementos existentes (Menéndez Domínguez y otros). A continuación se mencionan algunos de ellos.

1.6.1 ProgEs

ProgEs es un sistema cuyo objetivo es utilizar las ventajas que proporcionan los Sistemas de Aprendizaje Inteligente (SAI), las TIC, el *e-learning* y los REA. Este sistema incluye técnicas de inteligencia artificial que apoyan la instrumentación del proceso de enseñanza-aprendizaje. El proceso está basado en un grafo genético cuyos nodos en el nivel más alto de abstracción representan los objetivos instruccionales (Sánchez Guerrero, 2009).

1.6.2 ENSeñanza Artificial Intilligence (ENS-AI)

La denominación ENS-AI corresponde al acrónimo de ENSeñanza Artificial Inteligence, este sistema funciona como guía y ayuda para la práctica de la educación. Es un sistema experto capaz de simular razonamientos de expertos en educación. El objetivo final es orientar al profesor en las decisiones metodológicas más eficaces para su práctica educativa teniendo en cuenta los diagnósticos elaborados sobre las características de los alumnos del entorno de enseñanza-aprendizaje y las habilidades y destrezas del profesor (Barroso, 1994).

El modelo conceptual de ENS-AI consta de tres grandes conjuntos de datos: alumno, profesor y entorno de enseñanza-aprendizaje. Estos a su vez, están divididos en subconjuntos más específicos. Representan espacios de datos que, desde el punto de vista del conocimiento de los expertos en educación, permiten clasificar los datos en espacios deductivos significativos. Este modelo conceptual ha servido para decidir qué datos son más generales y constituyen la base de todo razonamiento; qué datos son específicos y sólo se han de considerar en unos determinados casos; cuándo se debe profundizar en aspectos concretos y qué hechos se pueden inferir de los datos considerados (Barroso, 1994).

1.6.3 SI-HOLMES

SI-Holmes es una aplicación desarrollada por investigadores de la Universidad de las Ciencias Informáticas. Está implementada bajo el paradigma de aplicaciones compuestas y usando el lenguaje Java cuya fuerza radica en su independencia de la plataforma, su seguridad, su portabilidad, su escalabilidad y en que promueve la reutilización de código.

El sistema tiene la ventaja de interactuar al mismo tiempo con múltiples bases de casos y con la información incluida (muestra sus nombres, cantidades de rasgos y casos de cada una de ellas) que pueden aparecer almacenadas en Sistemas Gestores de Bases de Datos, ficheros y documentos Excel y que son consumidas mediante el uso de servicios de datos, que le brindan a la aplicación una mayor flexibilidad a su capa de acceso a datos, pues estos servicios pueden ser compuestos en el desarrollo de otras aplicaciones que necesiten reutilizar esta estructura de base de casos.

SI-Holmes está diseñado bajo los principios de interoperabilidad que defiende SOA, para permitir que la información viaje desde el punto donde se proporciona hasta el punto donde se consume, sin importar las diferencias de la tecnología en que están desarrolladas las aplicaciones consumidoras (Febles Díaz, 2012).

El estudio de estas soluciones evidenció que solo responden a las necesidades particulares de las instituciones por las cuales fueron creadas, además de que la máquina de inferencia sobre la cual fueron creadas no es posible ajustarlas a otro tipo de información, por lo que se hace necesario la implementación de un nuevo sistema.

1.7 Herramientas y tecnologías a utilizar

Una metodología es un conjunto de filosofías, etapas, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de sistemas de información (Chipia Lobo, 2015).

Conocidos estos aspectos se puede plantear que para el desarrollo del SBC no se hace necesario el uso de metodologías ágiles o robustas de desarrollo de software, fundamentalmente porque como se explicó en epígrafes anteriores, los SBC plantean su propio ciclo de vida para el desarrollo de aplicaciones, donde cada etapa del mismo identifica los procedimientos a realizar para la implementación del sistema. Por otro lado, la documentación generada por metodologías convencionales de desarrollo de software reflejarían información con poco o ningún valor de utilidad debido a que el SBC se centrará esencialmente en elementos algorítmicos, matemáticos y de programación. Como solución se especificará en el Capítulo 2 la explicación detallada y minuciosa de los pasos seguidos para la implementación del SBC.

1.7.1 Selection Engine

Selection Engine es un proyecto de código abierto desarrollado por Baylor Wetzel en el 2001 utilizado para la implementación de sistemas basados en casos. Está implementado en el lenguaje de programación Java y utiliza el algoritmo K-Nearest Neighbor (KNN) en la implementación del módulo de recuperación para calcular la similitud con los casos buscados. Selection Engine lee datos y metadatos de ficheros de texto y envía la respuesta a la salida estándar (Giménez Arjona, 2006).

1.7.2 Marcos de desarrollo

Los marcos de desarrollo (*frameworks*) se pueden considerar como soluciones completas que contemplan herramientas de apoyo a la construcción (ambiente de trabajo o desarrollo) y motores de ejecución (ambiente de ejecución). Aceleran el proceso de desarrollo, permiten reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones (Zaninotto y otros, 2008).

Existen varias herramientas para la implementación de aplicaciones usando el CBR con el objetivo de recuperar y reutilizar las soluciones de experiencias pasadas para resolver problemas parecidos y así obtener mejores resultados, a continuación se ilustran algunos ejemplos.

Tabla 1: Resumen de herramientas CBR (Giménez Arjona, 2006)

Herramienta	Entorno	Recuperación	Adaptación	Interfaz
CASPIAN	MS-DOS, MAC	Nearest Neighbor	Mediante reglas	Línea de comandos
SELECTION ENGINE	Cualquier computadora que contenga la máquina virtual de Java, PCs Windows 95/98/NT, Sun Solaris y Macintosh, plataformas Unix	Nearest Neighbor	No dispone	Entorno gráfico

Capítulo 1: Fundamentación teórica de la investigación

CBR-Shell	Mozilla 1.2, Linux, Explorer 6.0, Windows XP y Mac O	Nearest Neighbor y K-Nearest Neighbor	Medidas de confianza adaptativas	de	Applet java
-----------	------------------------------------------------------	---------------------------------------	----------------------------------	----	-------------

Al comparar estas herramientas para la implementación de SBC se determina la utilización de SELECTION ENGINE ya que usa el algoritmo de los KNN y en sentido general sus características peculiares son las que mejor se corresponden con las necesidades de implementación del CBR deseado y las fases del ciclo de vida a desarrollar.

1.7.3 Lenguaje Java

Se utilizó Java en su versión 8u45 como lenguaje de programación orientado a objetos por las facilidades que ofrece, siendo actualmente uno de los lenguajes más usados para la programación en todo el mundo. Ofrece una gran seguridad, escalabilidad y portabilidad a las aplicaciones, además de ser multiplataforma y promover la reutilización de código (Belmonte Fernández, 2005). Está concebido además por la herramienta Selection Engine como lenguaje de desarrollo.

1.7.4 NetBeans 8.0

Como IDE de desarrollo se utilizó NetBeans en su versión 8.0, el mismo constituye un entorno de desarrollo integrado que ofrece características para ayudar en la construcción y mantenimiento de aplicaciones web, móviles y de escritorio. Es libre y gratuito sin restricciones de uso, resultado de un proyecto exitoso de código abierto con una gran comunidad de usuarios. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación.

Entre los cambios implementados en NetBeans IDE 8.0 destacan los analizadores de código y editores para trabajar con las últimas tecnologías Java 8 - Java SE 8, Java SE Embedded 8 y Java ME Embedded 8. El IDE también tiene una gama de nuevas mejoras que ratifican aún más su apoyo a Maven y Java EE con PrimeFaces, nuevas herramientas para HTML5, en particular para AngularJS y mejoras en la compatibilidad con PHP y C / C ++ (Oracle Corporation and/or its affiliates, 2015).

Conclusiones parciales

Con el estudio de los REA se conocieron las principales definiciones relacionadas con los mismos, su estructura y otras características necesarias para la realización del sistema experto. A partir del análisis de los sistemas expertos y el estudio de soluciones existentes a nivel internacional y nacional, se determinó que las mismas no brindan solución al problema planteado esencialmente porque no es posible modificar sus máquinas de inferencia para dar solución al objetivo principal de la presente investigación; por lo que se definió la implementación de un nuevo SBC tomando como base de casos la información existente en el repositorio RHODA. Se definieron además las categorías establecidas por el instrumento de evaluación LORI como fuente de origen de los rasgos objetivos de la base de casos del sistema.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

Conocidos los elementos referentes al diseño teórico de la investigación en curso y las tecnologías necesarias para la implementación de una solución que responda a la problemática planteada, se describen a continuación las principales características de la solución propuesta.

El objetivo fundamental del presente capítulo es detallar los elementos tenidos en cuenta para el diseño de la propuesta de solución, en correspondencia con los componentes de los sistemas basados en casos. Para una mejor comprensión de su funcionamiento se describe la estructura de la base de casos utilizada, el flujo realizado en el proceso de inferencia y la interfaz de entrada y salida de los datos al sistema. En cada una de estas fases se especifican los algoritmos, fórmulas y técnicas utilizadas.

2.1 Propuesta de solución

Para dar solución a la problemática planteada en la presente investigación se propone la implementación de SISDREA, un sistema experto que servirá de ayuda a los diseñadores del aprendizaje facilitando su labor en la realización del proceso de revisión de los REA defectuosos. El sistema es capaz de, a partir de un recurso defectuoso, realizar una inferencia aplicando el razonamiento basado en casos y brindar una sugerencia de posibles problemas a corregir en su diseño, de esta forma SISDREA servirá de guía a los diseñadores en la corrección y/o actualización de REA.

Para interpretar la información de los metadatos referentes a cada recurso, SISDREA reconoce el estándar de recolección de metadatos LOM en su versión 1.0. En el caso de la evaluación del recurso, el sistema es capaz de realizar sus predicciones en base a los 9 indicadores de evaluación propuestos por el instrumento LORI en su versión 1.5, determinando cuál de las categorías es la más afectada y obteniendo así una relación de los posibles problemas que pueda presentar el recurso.

La eficiencia y eficacia de un SBC depende principalmente de la medida en que se logren seleccionar y adaptar correctamente las técnicas y algoritmos más adecuados para estructurar cada una de sus etapas

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

y/o componentes, teniendo en cuenta las características peculiares de la información que se maneja y su entorno.

En epígrafes posteriores se especificará más detalladamente los algoritmos y técnicas usados para desarrollar cada una de las etapas del sistema que se obtiene como principal resultado de la presente investigación, el cual desarrolla solamente los componentes fundamentales de un SBC: la base de casos, la fase de recuperación de casos y la fase de adaptación de soluciones.

2.2 Obtención de la información de los REA almacenados en RHODA

Antes de explicar los procedimientos seguidos para la estructuración de la base de casos es necesario definir el origen de la información y su método de obtención. La mayor parte de la información a procesar se encuentra en los metadatos de los REA, los que pueden definirse mediante varios estándares entre los que se encuentra LOM, que constituye uno de los estándares de recolección de metadatos más completos y utilizados para la catalogación de recursos educativos, siendo el manejado por defecto en el repositorio RHODA para este fin, y por tanto, el reconocido en la primera versión de SISDREA.

Además de la información contenida en los metadatos se hace necesario obtener la evaluación correspondiente de cada recurso. Si determinado recurso aún no ha sido revisado y evaluado, no formará parte de la Base de Casos de SISDREA.

La información necesaria se encuentra en el repositorio RHODA y para obtenerla se hace uso de una serie de servicios web que este publica, precisamente, para ser consumidos por sistemas externos. Estos servicios corresponden a los métodos de los estándares SQI (*Simple Query Interface*), OAI-PMH (*Open Archives Initiative Protocol for Metadata Harvesting*) y SPI (*System Programming Interface Portal*) y se pueden encontrar dos conectores, uno para los servicios publicados por REST y otro para los publicados por SOAP (Cañizares González, 2012).

Los servicios de tipo SOAP están diseñados para brindar opciones de búsquedas de REA mientras que los publicados por REST permiten una recolección más exhaustiva de los metadatos de los recursos, además de que son más ligeros y sus respuestas contienen exactamente la información que se necesita, en este caso, el archivo XML con los metadatos de los Recursos Educativos Abiertos existentes en el repositorio. Mediante el protocolo REST, el repositorio ofrece también un servicio referente a la revisión de los REA, el

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

cual permite obtener su evaluación (siguiendo el estándar LORI explicado en el capítulo anterior) junto a su identificador único dentro del repositorio. Para más información sobre los métodos brindados en RHODA sobre el protocolo REST remitirse a los Anexos 3 y 4.

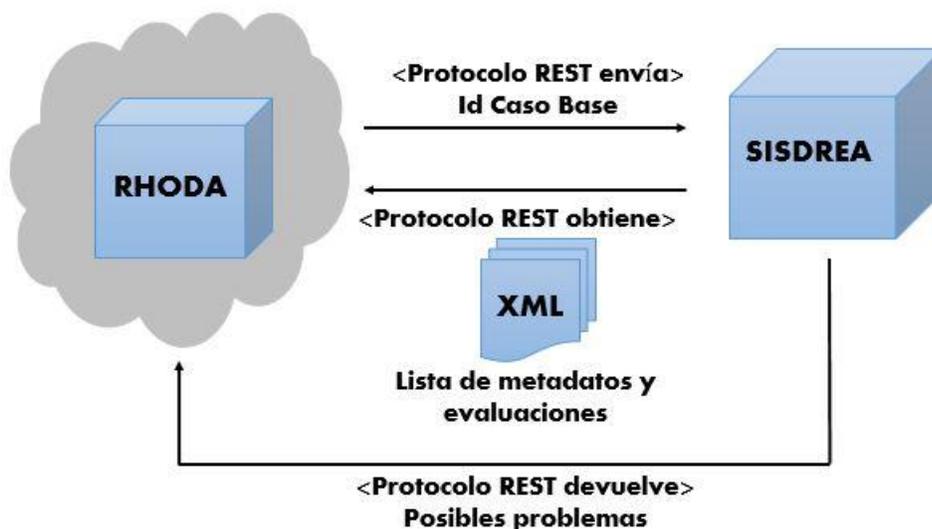


Figura 4: Conexión con el repositorio de recursos digitales RHODA (Elaboración propia)

El repositorio RHODA envía a través del servicio web REST el identificador del recurso defectuoso, luego SISDREA realiza las consultas para obtener los metadatos de los REA así como sus respectivas evaluaciones. Posteriormente devuelve un documento XML con los posibles problemas del recurso base.

2.3 Representación de los casos

Obtenida la información de los REA (dígase los metadatos de los recursos junto a las respectivas evaluaciones) en formato XML, se hace necesario leer y pre-procesar la misma, para ser posteriormente almacenada en una estructura de datos adecuada.

2.3.1 Determinación de los rasgos predictores y objetivos

La base de casos está compuesta por los rasgos predictores y rasgos objetivos. El estándar LOM para la catalogación de metadatos usado como referencia, establece nueve categorías diferentes con varias clasificaciones cada una, para un total de 47 tipos de metadatos diferentes.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

Para la determinación de los rasgos predictores se aplicaron encuestas para extraer información útil a los especialistas encuestados (ver Anexo 6) y a partir de un panel de expertos y la consulta de investigaciones y artículos científicos como la tesis doctoral de (Colomé Cedeño, 2013), se determinó que de estos rasgos, solo 10 tenían la relevancia suficiente para convertirse en los rasgos predictores de la base de casos del sistema, los cuales se describen a continuación usando la terminología *Elemento de metadatos/Categoría: valores admisibles* (Baltasar Fernández y otros, 2010):

1. *Estructura/General*: la estructura interna del material: atómico, colección, networked (en red), jerárquico y lineal.
2. *Nivel de agregación/General*: define la granularidad del material: 1, 2, 3 o 4.
3. *Estado/Ciclo de vida*: el estado de producción del material: borrador, final, revisado o no disponible.
4. *Tipo de interactividad/Educacional*: tipo de interacción soportado por el material: activo, expositivo o mixto.
5. *Tipo de recurso de aprendizaje/Educacional*: especifica el tipo de material: ejercicio, simulación, cuestionario, diagrama, figura, gráfico, diapositiva, tabla, texto narrativo, examen, experimento, declaración de problema, valoración, lectura o multimedia.
6. *Nivel de interactividad/Educacional*: especifica el nivel de interacción del material: muy bajo, bajo, medio, alto o muy alto.
7. *Densidad semántica/Educacional*: una medida subjetiva de la utilidad educativa del material en comparación con su tamaño y/o duración: muy bajo, bajo, medio, alto o muy alto.
8. *Rol del usuario final/Educacional*: determina el papel del usuario final del material: profesor, autor, estudiante o gestor.
9. *Dificultad/Educacional*: grado de dificultad del material: muy fácil, fácil, medio, difícil o muy difícil.
10. *Propósito/Clasificación*: define el propósito de la clasificación: disciplina, idea, prerrequisito, objetivo educacional, accesibilidad, restricciones, nivel educacional, nivel de habilidad, nivel de seguridad o competencia.

Los rasgos objetivos están constituidos por las evaluaciones establecidas para cada una de las categorías del instrumento de revisión de Recursos Educativos Abiertos LORI.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

2.3.2 Cálculo del peso de los rasgos predictores usando la técnica del Triángulo de Fuller

El método del Triángulo de Fuller es un método de comparaciones por parejas para tomar decisiones multicriterio a partir de expertos, se utiliza para obtener la importancia o peso de los rasgos predictores en la determinación del rasgo objetivo (Rodríguez, 2010).

A continuación se muestra el proceso seguido para la determinación de los pesos.

Para el cálculo de los pesos de los rasgos predictores se realiza una enumeración para cada uno de ellos, quedando de la siguiente manera:

1. Estructura
2. Nivel de agregación
3. Estado
4. Tipo de interactividad
5. Tipo de recurso de aprendizaje
6. Nivel de interactividad
7. Densidad semántica
8. Rol del usuario final
9. Dificultad
10. Propósito

Una vez enumerados los pesos se construye el triángulo de Fuller utilizando la enumeración realizada a cada uno de los rasgos predictores, como se muestra a continuación:

Tabla 2: Creación del triángulo de Fuller

1	1	1	1	1	1	1	1	1
<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
	2	2	2	2	2	2	2	2
	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
		3	3	3	3	3	3	3
		<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
			4	4	4	4	4	4
			<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
				5	5	5	5	5

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
6	6	6	6	6
7	<u>8</u>	<u>9</u>	<u>10</u>	
	7	7	7	
	<u>8</u>	<u>9</u>	<u>10</u>	
		8	8	
		<u>9</u>	<u>10</u>	
			9	
			<u>10</u>	

Seguidamente se hace una comparación entre cada uno de los rasgos y se marca con un asterisco el que tiene mayor importancia o relevancia según el experto, como se muestra en la tabla 3:

Tabla 3: Marcación de los rasgos más importantes según criterio del experto

1*	1	1	1	1	1	1*	1	1
<u>2</u>	<u>3*</u>	<u>4*</u>	<u>5*</u>	<u>6*</u>	<u>7*</u>	<u>8</u>	<u>9*</u>	<u>10*</u>
	2*	2	2	2	2	2*	2	2
	<u>3</u>	<u>4*</u>	<u>5*</u>	<u>6*</u>	<u>7*</u>	<u>8</u>	<u>9*</u>	<u>10*</u>
		3	3	3	3	3	3	3
		<u>4*</u>	<u>5*</u>	<u>6*</u>	<u>7*</u>	<u>8*</u>	<u>9*</u>	<u>10*</u>
			4*	4	4	4*	4	4*
			<u>5</u>	<u>6*</u>	<u>7*</u>	<u>8</u>	<u>9*</u>	<u>10</u>
				5*	5*	5*	5	5*
				<u>6</u>	<u>7</u>	<u>8</u>	<u>9*</u>	<u>10</u>
					6*	6*	6*	6*
					<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
						7*	7*	7*
						<u>8</u>	<u>9</u>	<u>10</u>
							8	8
							<u>9*</u>	<u>10*</u>
								9*
								<u>10</u>

Una vez señalados los rasgos más importantes, se calcula el valor de lambda (λ) para cada uno de ellos, representando la cantidad de ocasiones que cada uno es marcado con un asterisco.

$$\lambda_1 = 2, \lambda_2 = 2, \lambda_3 = 1, \lambda_4 = 6, \lambda_5 = 7, \lambda_6 = 8, \lambda_7 = 7, \lambda_8 = 1, \lambda_9 = 7, \lambda_{10} = 4.$$

Luego se procede a calcular el valor de cada rasgo predictor mediante la fórmula:

$$V_t = \lambda_t / N$$

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

Fórmula 4: Fórmula para determinar los pesos de cada rasgo predictor (Quintana y otros, 2004)

Donde N se obtiene a partir de:

$$N = k(k - 1) / 2$$

Fórmula 5: Fórmula para determinar el valor de N (Quintana y otros, 2004)

Donde k representa la cantidad total de rasgos predictores; luego sustituyendo los valores en la fórmula anterior quedaría:

$$N = \frac{10(10 - 1)}{2}$$

Después de obtener el valor de N se calculan los valores para cada uno de los rasgos, obteniéndose los siguientes resultados:

$$\begin{array}{lll} V_1 = \frac{2}{45} = 0,044 & V_2 = \frac{2}{45} = 0,044 & V_3 = \frac{1}{45} = 0,022 \\ V_4 = \frac{6}{45} = 0,13 & V_5 = \frac{7}{45} = 0,15 & V_6 = \frac{8}{45} = 0,17 \\ V_7 = \frac{7}{45} = 0,15 & V_8 = \frac{1}{45} = 0,022 & V_9 = \frac{7}{45} = 0,15 \\ & V_{10} = \frac{4}{45} = 0,088 & \end{array}$$

Obtenidos estos valores, quedan definidos los pesos para cada uno de los rasgos predictores.

2.4 Módulo de recuperación

Una de las partes más importantes en todo sistema SBC es la recuperación de los casos más relevantes. Para ello cada sistema utiliza diferentes técnicas y metodologías con el objetivo de obtener los casos de la manera más eficaz y rápida posible. Para la construcción del módulo de recuperación se hizo uso del algoritmo KNN.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

2.4.1 Aplicación del algoritmo K-Nearest Neighbor (KNN)

El modelo de KNN en la tarea de clasificación es una aproximación local que consiste en asignar a una entrada X la salida Y más común entre sus K vecinos más próximos.

Sea $X = \{x^i = (x_1^i, \dots, x_p^i), i = 1, \dots, N\}$ una colección de ejemplos de entrenamiento de dimensión p y sea $C = \{C_1, \dots, C_m\}$ un conjunto de M clases. Suponiendo que en cada ejemplo x^i existe una etiqueta de clase $L_i \in \{1, \dots, M\}$ que indica con certeza su pertenencia a una clase en C . Suponiendo también que x^S es un nuevo ejemplo para clasificar. Entonces, clasificar x^S , corresponde a asignarle una de las clases en C , esto es, decidir entre un conjunto de M hipótesis: $x^S \in C_q, q = 1, \dots, M$. Sea Φ^S el conjunto de los K -vecinos más próximos de x^S en X . Para cualquier $x^i \in \Phi^S$, el conocimiento de $L_i = q$ puede considerarse como una evidencia que aumenta la confianza de que x^S también pertenece a C_q . No obstante, esta evidencia no proporciona por sí misma 100 % de certeza (Bregón y otros, 2005).

En la situación descrita en la presente investigación, el algoritmo recibe como entrada los metadatos de un recurso defectuoso sin evaluación y realiza una búsqueda en la base de casos de recursos ya evaluados para encontrar los más semejantes. Posteriormente se toma la evaluación de estos para inferir la del recurso entrado (o caso base). Entiéndase por evaluación a la valoración del recurso, resultado del proceso de revisión del repositorio atendiendo a los indicadores del estándar LORI.

¿Cómo elegir el parámetro K (cantidad de casos similares a recuperar)? Si K es demasiado pequeño, entonces el resultado es muy sensible a puntos ruidosos, pero si K es demasiado grande, entonces la vecindad del punto desconocido incluirá muchos puntos de otras clases lejanas. Por tanto K debe ser grande para disminuir la probabilidad de una mala clasificación, pero pequeño en comparación con el número de puntos (Cárdenas Montes, 2014).

Dado que la base de recursos educativos con la que se cuenta es actualmente de solo 150, en los experimentos fueron probados valores pequeños de K , $K = 1, 3, 5$ y 7 ; y se encontró empíricamente que $K = 5$ proporcionó los mejores resultados para el conjunto de datos, aunque este valor pudiera revalorarse en caso de aumentar significativamente la cantidad de elementos en base de casos.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

2.4.2 Descripción del proceso de recuperación

La recuperación de los casos más similares se realiza de la siguiente manera:

1. Se calcula la distancia máxima teniendo en cuenta los atributos que describen el caso base y sus pesos como se muestra en la Fórmula 6, donde n es la cantidad de rasgos del caso base, w_i es el peso para el rasgo x_i , x_n es el mayor valor que puede tomar el rasgo x_i y x_{mi} es el menor valor que admite el rasgo x_i .

$$dist\ max = \sqrt{\sum_{i=0}^n w_i * (x_n - x_{mi})^2}$$

Fórmula 6: Fórmula para determinar la distancia máxima (Giménez Arjona, 2006)

2. Posteriormente se calcula la distancia entre el caso base y los casos fuente mediante el algoritmo KNN usando la fórmula euclidiana ponderada mostrada en la Fórmula 7, donde W_k constituye el peso del rasgo k y las expresiones $x_k(i)$ y $x_k(j)$ representan los valores del rasgo k para los objetos i y j respectivamente.

$$d(i, j) = \sqrt{\sum_{k=1}^p W_k (x_k(i) - x_k(j))^2}$$

Fórmula 7: Distancia euclidiana ponderada

3. Seguidamente se divide la distancia obtenida entre la distancia máxima (a modo de normalización del proceso) como se muestra en la Fórmula 8.

$$Similitud = \frac{dist\ C_x C_y}{dist\ max}$$

Fórmula 8: Fórmula para determinar el porcentaje de similitud de las respuestas encontradas (Giménez Arjona, 2006)

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

4. Se crean porcentajes de similitud entre la distancia del caso base y los k vecinos (casos fuentes) más próximos.
5. Por último se ordenan los valores y se seleccionan los cinco recursos con mayor porcentaje de similitud ($K=5$).

Para comparar los rasgos se utilizan las siguientes fórmulas teniendo en cuenta el tipo de dato de los mismos:

- ✓ Datos no numéricos (Borrajo Millán, 2014): $sim(x, y) = \begin{cases} 0 & \text{si } x = y \\ 1 & \text{si } x \neq y \end{cases}$
- ✓ Datos numéricos (Gutiérrez Martínez y otros, 2002): $sim(x_i, y_i) = 1 - \frac{|x_i - y_i|}{max_i - min_i}$

2.5 Módulo de adaptación

Una vez obtenidos o recuperados los cinco recursos más parecidos al caso base, se procede a la siguiente fase del ciclo de funcionamiento de un SBC: la adaptación.

Lo que se desea es poder determinar, con la mayor exactitud posible, cuál de las categorías definidas por el instrumento de evaluación LORI es la más afectada en el recurso educativo defectuoso que se tiene como entrada y de esta forma poder proponer los posibles problemas que pueda presentar dicho recurso. En este sentido, cada recurso de la base de casos posee una evaluación para cada uno de los indicadores definidos por LORI. Según establece dicho instrumento, estas evaluaciones pueden ser: excelente, muy bien, bien, regular, mal y no procede.

Para poder inferir las categorías más afectadas del recurso defectuoso que se tiene como base, es necesario partir de las categorías más afectadas de los cinco recursos más parecidos obtenidos como salida del proceso de recuperación. De estos recursos se tiene también un porcentaje de similitud respecto al caso base, valor que se tiene en cuenta para determinar la repercusión o utilidad del propio recurso en la respuesta obtenida por el proceso de adaptación.

Hasta este punto se puede analizar que los principales factores a tener en cuenta para ofrecer una respuesta útil son: las evaluaciones de los cinco recursos recuperados y el porcentaje de similitud de los mismos. Teniendo en cuenta que SISDREA formula problemas a partir de categorías afectadas, se propone

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

determinar numéricamente el nivel de afectación que posee cada categoría y de esta forma poder combinarlo con el porcentaje de similitud que se conoce de los recursos que estas describen.

Partiendo de que los valores con los que se evalúa son imprecisos numéricamente respecto a una afectación concisa de la categoría valorada, se decide aplicar los preceptos de la Lógica Difusa (Corzo, 2001) para obtener valores de pertenencia a la clasificación de *mal* (categoría afectada) y de esta forma poder realizar cálculos numéricos y obtener un desempeño más útil y preciso.

Para realizar la asignación de los grados de pertenencia a la evaluación de *mal* para cada indicador de LORI se sigue la siguiente distribución obtenida a partir del criterio de expertos:

- ✓ Excelente: valor de pertenencia=0
- ✓ Muy bien: valor de pertenencia= 0.1
- ✓ Bien: valor de pertenencia = 0.3
- ✓ Regular: valor de pertenencia = 0.6
- ✓ Mal: valor de pertenencia = 1

En caso de que el indicador o categoría haya sido especificado como *no procede*, se ignora el aporte de esta y por tanto no es tomada en cuenta en la realización de los cálculos numéricos consiguientes.

Una vez obtenidos los valores de pertenencia a la clasificación de *mal* para cada categoría se multiplican por el porcentaje de similitud del recurso. De esta manera se logra integrar de forma numérica la derivación útil que aporta el recurso educativo en cuanto a su porcentaje de similitud y las categorías afectadas que este posee.

Al aplicar el procedimiento a cada recurso recuperado se obtienen los valores parciales para cada categoría. Luego, para integrar la información aportada por cada valor se calcula la media aritmética de los mismos, obteniéndose así un solo valor por cada categoría de LORI capaz de describir el nivel de afectación de la misma, teniendo en cuenta la información útil contenida en los cinco recursos educativos resultados del proceso de Recuperación. Siendo estos valores directamente proporcionales con el grado de afectación de la categoría, los mismos son ordenados y se selecciona el mayor valor, obteniéndose así la presunta categoría más afectada luego de procesar la información de los cinco recursos más similares.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

A partir de la selección de la categoría más afectada, se obtienen los posibles problemas que pudieran estar atentando contra la integridad y calidad del REA. Dichos problemas son ofrecidos como salida al proceso de inferencia realizado por SISDREA.

El procedimiento de adaptación descrito constituye el resultado de un perfeccionamiento continuo e intuitivo, donde secuencialmente se fueron reajustando los parámetros obtenidos en los recursos recuperados y las operaciones realizadas, a fin de obtener mejores desempeños en las predicciones realizadas por el sistema.

2.6 Interfaz de E/S

Explicada la estructura y funcionamiento del CBR que implementa SISDREA, solo queda definir los elementos relacionados con la interfaz de E/S del sistema, que permitirá recibir el recurso defectuoso y mostrar la respuesta correspondiente como resultado del proceso de inferencia. Como se ha explicado anteriormente, SISDREA está orientado a ser útil en la realización del proceso de revisión de los REA defectuosos, este proceso se realiza a través de una interfaz visual en el repositorio RHODA, no siendo objetivo contar con una interfaz visual en el sistema para mostrar las salidas de la inferencia.

Por tal motivo se define la publicación de un servicio web sustentado sobre el protocolo REST, que podrá ser consumido por el repositorio RHODA. Dicho servicio devolverá mediante un archivo XML, y a partir de un REA defectuoso indicado por el repositorio, una propuesta de posibles problemas que deberán ser mostrados al diseñador del aprendizaje a través de la interfaz visual del proceso de revisión del recurso.

En la siguiente tabla se muestran las especificaciones del servicio web propuesto:

Tabla 4: Especificaciones del servicio web propuesto

Método	Tipo	Parámetros de entrada	Salida
getProblemsInference	GET	Identifier: identificador único del recurso educativo en la base de datos del repositorio de RHODA.	XML plano con la propuesta de problemas obtenidos como salida de sistema, tomando como caso base el recurso que cumple con el parámetro identifier.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

El archivo XML de salida tendrá la siguiente estructura:

- ✓ Nodo raíz: etiqueta *<instrumento>*, atributo nombre donde se especifica el nombre del instrumento de revisión utilizado.
- ✓ 1er subnodo hijo: etiqueta *<indicador>*, nombre del indicador afectado.
- ✓ 2do subnodo hijo: etiqueta *<características>*, características del indicador afectado.
- ✓ 2do subnodo hijo: etiqueta *<descripción>*, descripción del contexto idóneo para la no afectación del indicador.
- ✓ 2do subnodo hijo: etiqueta *<ejemplo>*, ejemplo donde se evidencia la afectación del indicador.
- ✓ 2do subnodo hijo: etiqueta *<problema>*, el problema que afecta al indicador.

Ejemplo de respuesta en formato XML:

```
<?xml versión="1.0" encoding="UTF-8"?>
<Instrumento_para_la_evaluación_de_objetos_de_aprendizaje nombre="LORI_ESP">
  <indicador3 nombre="Feedback (Retroalimentación) y Adaptabilidad">
    <características value="Contenido adaptativo o feedback en función de la respuesta de
cada alumno/a y su estilo de aprendizaje"/>
    <problema value="No aporta feedback relativo a la calidad o corrección de las respuestas
del alumnado"/>
    <problema value="No mantiene un registro de las respuestas o del estilo de aprendizaje del
alumno/a de cara a adaptar las siguientes presentaciones de contenido"/>
    <problema value="No hay simulación o un conjunto de herramientas que se adapten en
función de las respuestas del alumno/a"/>
  </indicador3>
</Instrumento_para_la_evaluación_de_objetos_de_aprendizaje>
```

Conclusiones parciales

En el presente capítulo se indagó en los elementos relacionados con la propuesta de solución, especificándose cada una de las fases y componentes de la misma. Se obtuvieron los rasgos predictores usando el criterio de expertos así como sus respectivos pesos utilizando la técnica del triángulo de Fuller.

Capítulo 2: Propuesta de solución para el sistema basado en casos SISDREA

Para calcular los porcentajes de similitud entre varios REA se empleó el algoritmo K-Nearest Neighbor obteniéndose como salida los casos más semejantes con los posibles problemas que pueda contener el caso base. Para el módulo de adaptación se hizo uso de los principios de la lógica difusa para determinar los posibles problemas de un REA defectuoso obtenido del repositorio RHODA.

Capítulo 3: Implementación y validación del Sistema Basado en Casos SISDREA

En este capítulo se describen los elementos establecidos por el proceso de desarrollo empleado que responden a la implementación de la propuesta de solución, especificándose las clases utilizadas en cada uno de los módulos descritos en el capítulo anterior. También se realizan pruebas para validar el correcto funcionamiento de la solución propuesta y medir los niveles de satisfacción logrados con su utilización por parte de un grupo de usuarios.

3.1 Implementación

En el marco de la investigación se toma como punto de partida el análisis anteriormente realizado para especificar las clases definidas en la implementación de los diferentes módulos de SISDREA, así como los estándares de codificación utilizados para contribuir al entendimiento del código por futuros desarrolladores que deseen ampliar o reutilizar los módulos del sistema.

3.1.1 Estándares de codificación

Los estándares de codificación son reglas o patrones de escritura del código de programación que permiten tener una programación homogénea y que regula la realización de ciertos procesos, de esta forma el sistema resulta fácil de entender y mantener, independientemente de su autor. En la implementación de SISDREA se utilizaron varios estándares para favorecer la legibilidad y organización al código, simplificando esfuerzos a la hora de darle mantenimiento y seguimiento a la aplicación.

- **Identación, llaves de apertura y cierre y tamaño de las líneas**

Se utiliza una indentación sin tabulaciones con un equivalente a cuatro espacios. El uso de las llaves “{ }” se realiza de la siguiente forma: la de apertura al final de la línea de código y la de cierre en una nueva línea y la longitud de las líneas es aproximadamente de 75 a 80 caracteres para mantener la legibilidad del código.

A continuación se muestra un ejemplo:

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

```
34  □ public SimilarityDescription getBestMatch() {  
35      SimilarityDescription descriptor  
36          = (SimilarityDescription) descriptions.get(0);  
37  
38      return descriptor;  
39  } //--- getBestMatch
```

Figura 5: Uso de la indentación y las llaves (Elaboración propia)

- **Convención de nomenclatura**

Para la implementación se aplicaron los estándares de codificación y nomenclatura recomendados por la Sun Microsystems (Java, 2010) como se muestra en la Figura 6, a continuación se describen algunos de estos.

- ✓ *Clases*: la primera letra de los nombres de clase debe de ser mayúscula, si son varias palabras se debe de intercalar entre mayúsculas y minúsculas, este mecanismo de nombre es llamado *camelCase* (por ejemplo: *SimilarItems*, *LoriEvaluation* y *REA*).
- ✓ *Interfaces*: se aplica la misma definición de clases, pero se debe colocar la terminación “able” al final del nombre de las clases (por ejemplo *Runnable*, *Serializable*).
- ✓ *Métodos*: la primera letra de los métodos de clase debe ser minúscula, si son varias palabras se debe intercalar entre minúsculas y mayúsculas, para el caso de los métodos de clases se aplica el mecanismo del *camelCase* (por ejemplo *getSimilarEvaluation*, *computeSimilarity* y *getSimilarEvaluation*).
- ✓ *Variables*: para las variables se procede similar a los métodos, donde la primera letra es minúscula y las demás se deben de guiar por el mecanismo de *camelCase*, lo que es importante de destacar, es que los nombres de las variables, además de cumplir lo anterior, deben ser cortos y descriptivos. (por ejemplo: título, evaluation y casoBase).
- ✓ *Constantes*: su nombre debe ser escrito completamente en mayúsculas, y para la separación de palabras se debe usar el *underscore*/guión bajo (*_*). (por ejemplo *MAX_SUMA* y *VALOR_DISTANCIA*).

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

```
public class SimilarItems {
17 //-----
18 // Attributes
19 //-----
20 //--- Sorted list of SimilarityDescription objects
21 private ArrayList descriptions = new ArrayList();
22
23 //-----
24 // Methods
25 //-----
26 protected void add(SimilarityDescription descriptor) {
27     descriptions.add(descriptor);
28 } //--- add
29
30 public ArrayList getDescriptions() {
31     return descriptions;
32 }
33
34 public SimilarityDescription getBestMatch() {
35     SimilarityDescription descriptor
36         = (SimilarityDescription) descriptions.get(0);
37
38     return descriptor;
39 } //--- getBestMatch
```

Figura 6: Uso de estándares para la nomenclatura (Elaboración propia)

- **Estructuras de control**

Las estructuras de control utilizan siempre llaves de apertura y cierre “{}”, incluso en situaciones en las que técnicamente son opcionales. Con esto se aumenta la legibilidad del código y se disminuye la probabilidad de errores lógicos. Incluyen *if*, *for*, *foreach*, *while* y *switch*.

```
68 while (targetValueList.hasNext()) {
69     LomStandar_Rasgo targetScore
70         = (LomStandar_Rasgo) targetValueList.next();
71     float delta = 1 * targetScore.getPeso();
72     float squaredDelta = (delta * delta);
73     sum += squaredDelta;
74 }
```

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

Figura 7: Uso de estándares para la definición de estructuras de control (Elaboración propia)

El uso de estándares para la codificación de SISDREA facilita que nuevos desarrolladores puedan implementar otros cambios en el sistema, además de optimizar el código fuente y permitir un fácil mantenimiento del mismo en caso de ser requerido en posteriores momentos de actualización y soporte.

3.1.2 Pre-procesamiento de la información. Librería *LomStandarInterpreter*

Cuando el repositorio RHODA especifica un REA defectuoso, SISDREA se conecta al mismo y obtiene la información de los recursos que conformarán la base de casos. Luego de obtenida la información de los REA (dígase los metadatos de los recursos junto a sus respectivas evaluaciones) en formato XML, se hace necesario leer y pre-procesar la misma para ser posteriormente almacenada en una estructura de datos adecuada. Para la ejecución de estas acciones se implementó una librería en el lenguaje Java nombrada *LomStandarInterpreter*, capaz de interpretar los archivos XML con las informaciones de los metadatos y las evaluaciones y modelarlos en una estructura de datos del propio lenguaje utilizando el paradigma de programación orientada a objetos. Realizar las acciones de estandarización y pre-procesamiento de la información conlleva el uso por parte de esta librería de las bondades del polimorfismo y la herencia de clases definidas por Java a través del método abstracto *Standardize* mostrado en la Figura 8.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

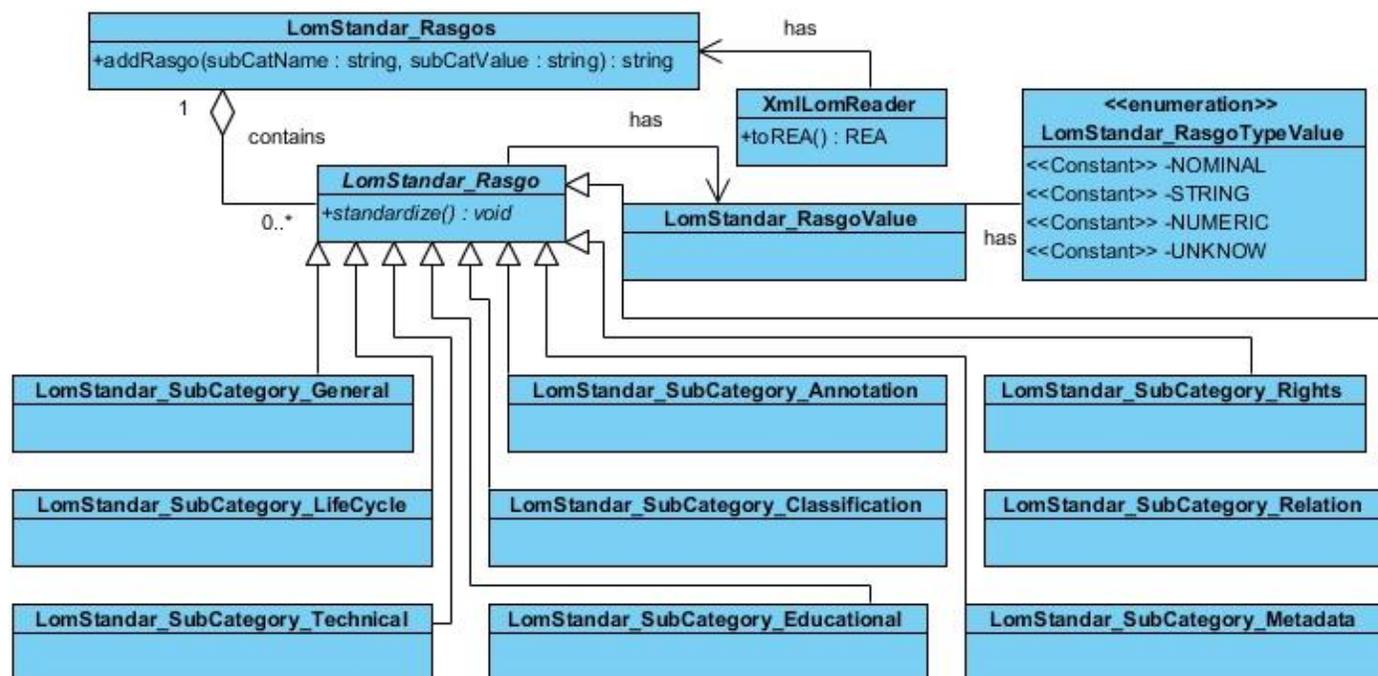


Figura 8: Relación entre las clases de la librería *LomStandarInterpreter* (Elaboración propia)

Para obtener la información contenida en los archivos XML se utilizó la librería JDOM 2.0.6 (Hunter y otros, 2015), la cual provee de una solución completa para el acceso, manipulación y creación de datos XML desde el código de Java.

El pre-procesamiento de la información obtenida consta de dos procesos secuenciales fundamentales:

- ✓ Reconocimiento de los rasgos y categorías definidas en LOM en su versión 1.0.
- ✓ Estandarización de los valores encontrados en los rasgos.

El reconocimiento de los rasgos permitirá posteriormente la estandarización, lo que conlleva a la siguiente interrogante, ¿por qué se hace necesario estandarizar los datos? El aprendizaje automatizado, y más específicamente el razonamiento basado en casos, utilizan una serie de algoritmos y fórmulas matemáticas para el cálculo de la semejanza entre casos. Para el correcto funcionamiento de estos algoritmos, tradicionalmente se han centrado en tipos de datos escalares (nominales y numéricos) (Ramírez Cabrera, 2013). Esto está dado porque se hace complicado comparar cadenas de texto redactadas libremente por el usuario, que pudieran contener por ejemplo, palabras que signifiquen lo mismo, pero escritas en diferentes

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

idiomas. Para estandarizar los datos se realizan las siguientes acciones sobre los rasgos, que una vez reconocidos, se cuenta con el vocabulario LOM para la descripción de los mismos (rasgos con valores escalares):

- ✓ Se lleva el tamaño de la letra a minúscula.
- ✓ Se ignoran los caracteres extraños y espacios.
- ✓ Se traducen los datos de texto al idioma español en caso de que se encuentre en inglés.
- ✓ Se compara el valor obtenido con los referidos en el estándar LOM hasta encontrar una coincidencia, si no se encuentran coincidencias se ignora el rasgo.

En caso de que se trate específicamente de un rasgo cuyo valor sea numérico (por ejemplo la versión del recurso, el tamaño en MB (Megabyte) o el tiempo de duración del mismo (en minutos)) se parsea el texto hasta reconocer el número contenido. Si el valor del rasgo no es más que una cadena de texto redactada libremente, entonces se almacena tal y como es.

En el Anexo 5 se muestra una tabla con los 47 rasgos contenidos en las nueve categorías definidas por LOM donde se describe el tipo de datos que admite cada uno con su correspondiente vocabulario, si existe. Los tipos de datos definidos son: numérico, nominal y string (entiéndase string por cadenas de texto no nominales).

Es importante destacar que a pesar de que la librería *LomStandardInterpreter* es capaz de reconocer todos los rasgos de metadatos de LOM, una vez identificados los rasgos predictores y objetivos a ser tomados en cuenta, serán solo estos los que continuarán el proceso de almacenamiento en la base de casos, los demás rasgos serán ignorados.

3.1.3 Estructuración de la base de casos

Identificados los rasgos de metadatos de un REA y estandarizados sus valores se procede al almacenamiento de estos, junto con la evaluación del recurso, en una estructura de datos que favorezca su acceso cuando sea necesario. Dicha estructura será la encargada de representar los casos (Recursos Educativos Abiertos) que van a conformar la base de casos.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

Existen varias formas de representación de casos de un SBC, entre las que se encuentran (Chaveco Bustamante y otros, 2013):

- ✓ *Memoria plana de pares atributo-valor*: a menudo la estructura de caso simple es suficiente para resolver el problema. Además cuenta con gran facilidad de almacenamiento y recuperación en un SBC.
- ✓ *Memoria Jerárquica*: en casos más complejos es conveniente utilizar representaciones en forma de grafos: conjunto de nodos y arcos.

Teniendo en cuenta que el sistema almacenará casos sencillos, compuestos básicamente por metadatos de los recursos, se ha decidido utilizar la representación plana, la cual tiene varias ventajas, entre ellas se puede destacar que el almacenamiento de casos se realiza de forma muy sencilla. La representación plana necesita menos recursos para ejecutarse que una representación en forma de grafo.

Dicha representación se complementó con la utilización de las interfaces de Java *LinkedHashMap*, para el almacenamiento de los rasgos dentro de cada recurso y el *LinkedList* para el almacenamiento de cada recurso en la base de casos.

Se seleccionó la estructura de *LinkedHashMap* pues permite representar una estructura de datos para almacenar pares “clave/valor”; de tal manera que para una clave solamente se tiene un valor. Esta estructura de datos también es conocida en otros lenguajes de programación como “Diccionarios”. Es especialmente rápida a la hora de agregar elementos y acceder a los mismos.

Por su parte la estructura *LinkedList* es una de las implementaciones para listas más utilizadas, diseñada para obtener un mejor rendimiento cuando se insertan o eliminan muchos elementos de la mitad de la colección. Este tipo de lista como su nombre indica, utiliza una lista doblemente enlazada internamente, en lugar de un *array* (arreglo). Esta estructura guarda los objetos respetando el orden en que son insertados y son sencillas de recorrer usando ciclos (González Duque, 2015).

3.1.4 Clases definidas para el módulo de recuperación

Seguidamente se describe la estructura de clases implementadas en el lenguaje Java para garantizar el correcto desempeño del módulo de recuperación.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

La clase principal del módulo de recuperación es la llamada *SimilarityEngine*, dicha clase es el motor de inferencia del CBR. Esta hace uso de varios objetos definidos como se muestra en la Figura 9. REAs es el objeto que contiene la colección de objetos REA (recurso) que constituyen la base de casos. Cada objeto REA posee una clase *LoriEvaluation* mediante la cual se gestionan las evaluaciones del recurso realizadas usando el instrumento LORI. Además posee un objeto *LomStandar_Rasgos*, el cual constituye una colección de objetos *LomStandar_Rasgo* (rasgo de metadato).

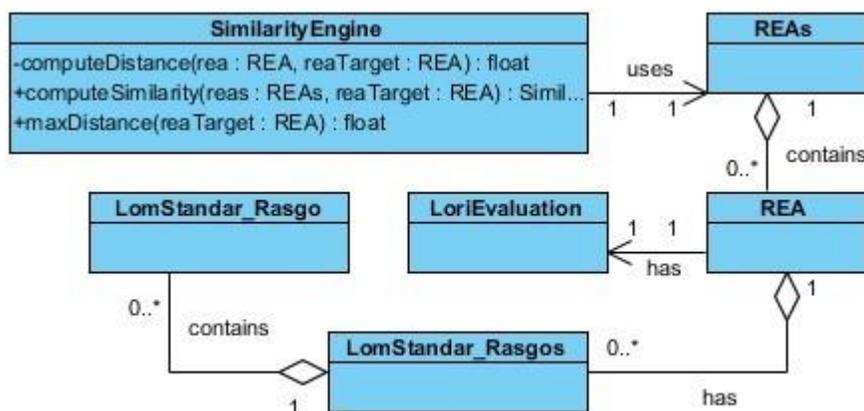


Figura 9: Relación entre las clases del módulo de recuperación (Elaboración propia)

Para ofrecer la salida de los recursos más similares se utiliza el objeto *SimilarItems*, el cual es una colección de objetos *SimilarityDescription*. Estos últimos contienen, como se muestra en la Figura 10, la información del recurso junto a su porcentaje de similitud.

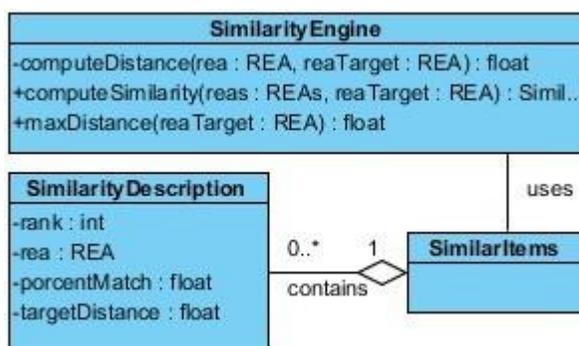


Figura 10: Relación entre las clases encargadas de ofrecer como salida los recursos más similares (Elaboración propia)

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

3.1.5 Clases definidas para el módulo de adaptación

El procedimiento de adaptación descrito en la propuesta de solución fue implementado a través de la clase *FuzzyAdaptation* mostrada en la Figura 11, mientras que el código de la misma puede ser encontrado en el Anexo 7.

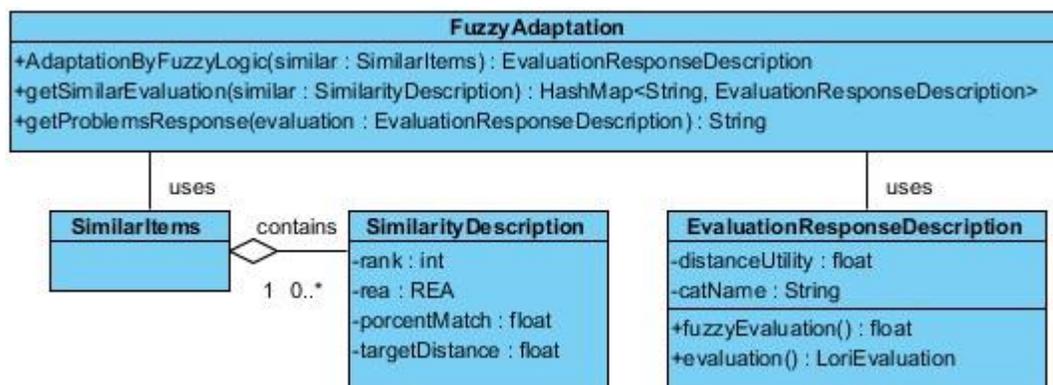


Figura 11: Clase FuzzyAdaptation (Elaboración propia)

3.2 Validación

La validación es una de las etapas más importantes en el análisis del comportamiento de un sistema experto. El término validación se refiere a un análisis de la calidad del sistema inteligente en su entorno real de trabajo, lo que permite determinar si el producto desarrollado satisface convenientemente las expectativas inicialmente depositadas.

Con el objetivo de realizar la validación de SISDREA, se hace uso de la técnica de ladov para medir el nivel de satisfacción que podrían tener los diseñadores del aprendizaje con el uso de SISDREA. Seguidamente se muestran los resultados de la aplicación de la técnica de validación cruzada y las pruebas unitarias realizadas al sistema.

3.2.1 Aplicación de la técnica de ladov

La técnica de ladov en su versión original fue creada por (Kuzmina, 1970) para el estudio de la satisfacción por la profesión en carreras pedagógicas. Dicha técnica está conformada por cinco preguntas: tres cerradas

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

y dos abiertas y constituye una vía indirecta para el estudio de la satisfacción, ya que los criterios que se utilizan se fundamentan en las relaciones que se establecen entre tres preguntas cerradas que se intercalan dentro de un cuestionario y cuya relación el sujeto desconoce. Estas tres preguntas se relacionan a través de lo que se denomina el "Cuadro Lógico de Iadov" (López Rodríguez y otros, 2002).

La presente investigación está destinada fundamentalmente a los diseñadores del aprendizaje los cuales deberán emitir su criterio sobre el funcionamiento de SISDREA, validando que las respuestas ofrecidas sean lo más acertadas posible de acuerdo a su criterio profesional.

Tabla 5: Cuadro Lógico de Iadov

3-¿Cuál es su opinión sobre el sistema basado en casos SISDREA utilizado en el proceso de revisión de los REA para la identificación de problemas en sus diseños?	1-¿Considera usted factible la utilización de un sistema experto en el proceso de revisión de los REA?								
	No			No sé			Sí		
	2-¿Considera usted factible la utilización de un sistema basado en casos para la identificación de problemas en el diseño de los REA?								
	Sí	No sé	No	Sí	No sé	No	Sí	No sé	No
Me gusta mucho.	1	2	6	2	2	6	6	6	6
Me gusta más de lo que me disgusta.	2	2	3	2	3	3	6	3	6
Me es indiferente.	3	3	3	3	3	3	3	3	3
Me disgusta más de lo que me gusta.	6	3	6	3	4	4	3	4	4
No me gusta en lo absoluto.	6	6	6	6	4	4	6	4	5
No sé.	2	3	6	3	3	3	6	3	4

El número resultante de la interrelación de las tres preguntas indica la posición de cada sujeto en la escala de satisfacción siguiente (López Rodríguez y otros, 2002):

1. Clara satisfacción
2. Más satisfecho que insatisfecho

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

3. No definida o contradictoria
4. Más insatisfecho que satisfecho
5. Clara insatisfacción

Para obtener el índice de satisfacción grupal (ISG) se trabaja con los diferentes niveles de satisfacción que se expresan en la escala numérica que oscila entre -1 y +1 de la siguiente forma:

Tabla 6: Niveles de satisfacción

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

El índice de satisfacción grupal se calcula de la siguiente forma:

$$ISG = \frac{A(+1) + B(+0.5) + C(0) + D(-0.5) + E(-1)}{N}$$

Fórmula 9: Fórmula para calcular el ISG

Para esta fórmula A, B, C, D y E; representan el número de sujetos con índice individual (1; 2; 3 o 6; 4; 5) y donde *N* representa el número total de sujetos del grupo.

Al calcular el ISG a diez especialistas en el diseño de REA se obtuvo según la Fórmula 9 el siguiente ISG:

$$ISG = \frac{8(+1) + 1(+0,5) + 1(0) + 0(-0,5) + 0(-1)}{10}$$

$$ISG = 0,85$$

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

El índice grupal arroja valores entre -1 y +1. Los valores que se encuentran comprendidos entre -1 y -0,5 indican insatisfacción, los comprendidos entre -0,49 y +0,49 evidencian contradicción y los que se enmarcan entre 0,5 y 1 indican que existe satisfacción.

Al representar estos valores gráficamente en un eje se aprecian de la siguiente manera:



Figura 12: Línea de satisfacción

El valor del ISG obtenido al aplicar la técnica de ladov es 0.85 y se encuentra en el intervalo de satisfacción, por lo que se puede concluir que existe satisfacción al utilizar SISDREA para detectar problemas en el diseño de REA.

3.2.2 Aplicación de la técnica Validación Cruzada (Cross-Validation)

La Validación Cruzada o *Cross-Validation* es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Para operar divide la muestra en un número de submuestras, generando luego los modelos de árbol, que no incluyen los datos de cada submuestra (IBM Corporation, 2012).

De esta forma, en la validación cruzada *K-fold* (Anastasiadis y otros, 2003), un conjunto de datos D es dividido aleatoriamente en k subconjuntos mutuamente excluyentes D_1, \dots, D_k , el clasificador es entrenado y probado k veces; cada vez $t \in 1, 2, \dots, k$ es entrenado sobre $D_i, i = 1, \dots, k$, con $i \neq t$ y probado sobre D_t . La estimación de la precisión de la validación cruzada es el número global de clasificaciones correctas dividido por el número de ejemplos del conjunto de datos.

Para la aplicación de la validación cruzada se siguieron los siguientes pasos:

1. Se tomaron 140 REA del repositorio RHODA como conjunto de entrenamiento de 150 existentes.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

2. Luego se separó el conjunto de entrenamiento en k *folds*, con $k = 2$.
3. Se combinaron los diferentes *fold* para obtener diferentes pruebas de entrenamiento y test.
4. Por último se registró la media de los errores de los k *folds*.

Al concluir la aplicación de la validación cruzada a SISDREA se obtuvo un total de 93,33% de clasificaciones correctas, lo que se considera como un desempeño aceptable del sistema.

3.2.3 Aplicación de las pruebas unitarias

Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representan una revisión final de las especificaciones del diseño y de la codificación. Las comprobaciones manuales o automatizadas desarrolladas por los programadores de software conforman las llamadas pruebas de unidad o pruebas unitarias, las cuales se realizan para verificar que el código correspondiente a un módulo concreto se comporte de la manera que se espera, teniendo como objetivo aislar cada parte del programa y mostrar que las partes individuales son correctas.

Las pruebas unitarias aportan un grupo importante de beneficios, dentro de los que destacan ofrecer al programador una inmediata retroalimentación de cómo está realizando su trabajo y posibilitarle la realización de cambios de forma segura.

A los módulos que componen SISDREA obtenidos como resultado de la presente investigación, se le realizaron las pruebas unitarias correspondientes a fin de validar sus principales funcionalidades y clases. Para su aplicación se utilizó una técnica automatizada de software, la cual permite realizar pruebas al código del sistema utilizando la herramienta *jUnit*, la cual está diseñada especialmente para implementar y automatizar la realización de pruebas de unidad en el lenguaje de programación Java.

A medida que se fueron realizando las pruebas correspondientes se detectaron algunos errores en el código, los cuales se resolvieron mientras se desarrollaban las funcionalidades. Una vez realizada la corrección de los errores detectados en pequeñas iteraciones, los métodos de cada clase se ejecutaron de forma correcta, obteniendo de ellos los resultados esperados, como se muestra a continuación.

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

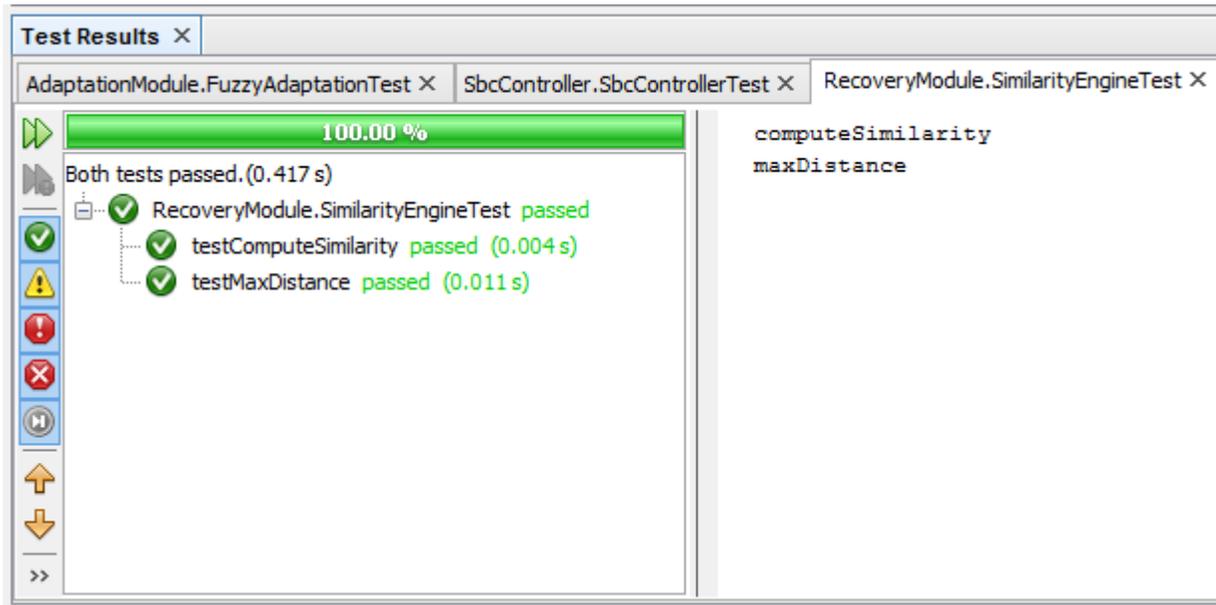


Figura 13: Resultados de la prueba unitaria realizada al módulo de Recuperación

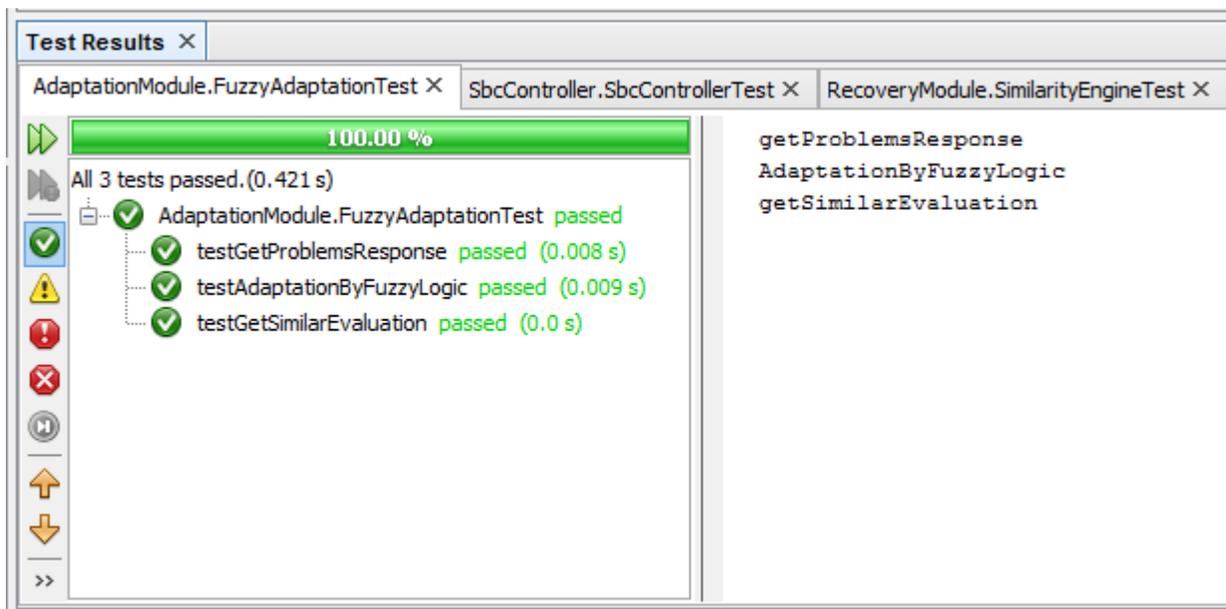


Figura 14: Resultados de la prueba unitaria realizada al módulo de Adaptación

Capítulo 3: Implementación y validación del sistema basado en casos SISDREA

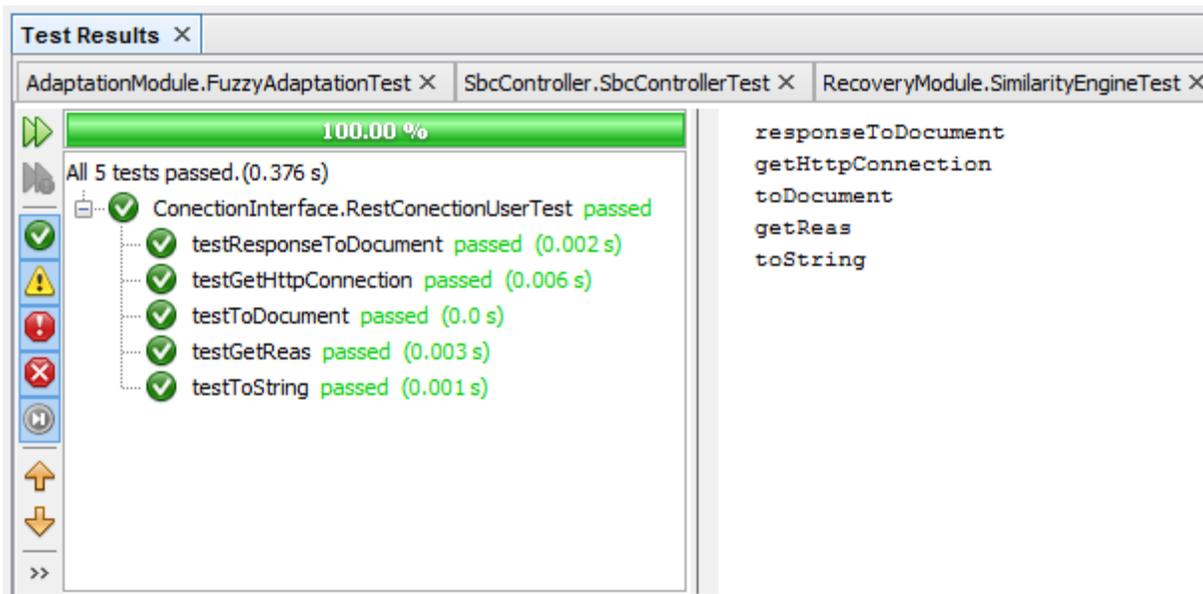


Figura 15: Resultados de la prueba unitaria realizada a la Interfaz de E/S

Conclusiones parciales

Al concluir este capítulo se analizó la estructura e implementación de las principales clases que se utilizaron para el desarrollo de cada uno de los módulos que componen la propuesta de solución. Se implementó una librería llamada *LomStandardInterpreter* capaz de procesar la información acerca de los recursos educativos y sus evaluaciones, obtenida del repositorio RHODA.

Con la realización de pruebas de validación a SISDREA se obtuvo un nivel de satisfacción adecuado de acuerdo a la técnica de ladov, la cual se llevó a cabo mediante la encuesta a especialistas en el trabajo con REA de la Universidad de las Ciencias Informáticas. Para medir el nivel de exactitud en las respuestas de SISDREA, se utilizó la técnica de Validación Cruzada, la cual arrojó como resultado un 93,33% de clasificaciones correctas de problemas en un REA procesado por el sistema. Se realizaron además pruebas unitarias donde se validó que el código correspondiente a los módulos de Recuperación, Adaptación e Interfaz de E/S, funciona correctamente según lo esperado.

Conclusiones

Con la realización del presente trabajo de diploma se ha cumplido con el objetivo general propuesto, así como con los objetivos específicos, obteniéndose las siguientes conclusiones:

- ✓ Con el estudio de los estándares y especificaciones relacionadas con el diseño de Recursos Educativos Abiertos se determinaron los 10 atributos del estándar LOM, seleccionados por criterio de expertos, que son empleados como los rasgos predictores de la base de casos de SISDREA. Por otro lado al estudiar los instrumentos de evaluación de Recursos Educativos Abiertos existentes se determinó que es posible obtener de estos los problemas más comunes que se cometen en el diseño de REA para ser usados como rasgos objetivos del sistema, tomándose como referencia el instrumento LORI.
- ✓ El estudio de los principales sistemas expertos, evidenció que los mismos no poseen las características necesarias para identificar posibles problemas en los recursos educativos abiertos, además de que sus máquinas de inferencia no pueden ser adaptadas para tal fin.
- ✓ Se implementó un Sistema Basado en Casos para la identificación de problemas en los diseños de los Recursos Educativos Abiertos, usando el algoritmo KNN con la utilización de la distancia Euclidiana y el uso de la lógica difusa para adaptar las soluciones. Incluyó además el desarrollo de la librería *LomStandardInterpreter* para procesar los metadatos obtenidos del repositorio RHODA, convirtiéndolos en objetos y logrando que SISDREA los interprete en idioma español o inglés.
- ✓ Se validó la propuesta de solución usando la técnica de ladov con la que se evidenció una clara satisfacción por parte de los diseñadores de aprendizaje, mientras que con la técnica de validación cruzada se obtuvo un 93,33% de clasificaciones correctas, lo que equivale a un desempeño aceptable del sistema. Con la realización de pruebas unitarias se validó el comportamiento satisfactorio del código de la aplicación.

Recomendaciones

Al finalizar la presente investigación se recomienda:

- ✓ Implementar la interpretación de otros estándares de catalogación de metadatos, además de LOM, para una mejor integración de SISDREA con repositorios de REA.
- ✓ Integrar otros algoritmos de recuperación y adaptación a SISDREA que permitan una mayor versatilidad al procesar diversas bases de casos.

Referencias bibliográficas

1. (Agudo, 2001) **Agudo, B. 2001.** *Introducción al razonamiento basado en casos*. España: Universidad Complutense de Madrid, 2001.
1. (Anastasiadis y otros, 2003) **Anastasiadis, A. D. , Magoulas, G. D. y Liu, X. . 2003.** *Classification of Protein Localisation Patterns via Supervised Neural Network Learning*. Berlin, Alemania : s.n., 2003.
2. (Atkins y otros, 2007) **Atkins, Daniel, Brown, John y Hammond, Allen. 2007.** The William and Flora Hewlett Foundation. [En línea] 2007. [Citado el: 3 de Noviembre de 2014.]
http://www.hewlett.org/uploads/files/Hewlett_OER_report.pdf.
3. (Baltasar Fernández y otros, 2010) **Baltasar Fernández, Manjón, y otros. 2010.** *Uso de estándares aplicados a TIC en Educación, ep. 5*. Ministerio de Educación y Ciencia. España: s.n., 2010.
4. (Barberá Gregori y otros, 2005) *El uso educativo de las aulas virtuales emergentes en la educación superior.* **Barberá Gregori , Elena y Badia Garganté, Antoni . 2005.** 2, 2005, Vol. 2. 1698-580X.
5. (Barroso, 1994) ENS-AI: Un sistema experto para la enseñanza. **Barroso, Clara. 1994.** Santa Cruz de Tenerife, España: s.n., 1994, Vol. VI.
6. (Begaña Vitoriano y otros, 2003) **Begaña Vitoriano, Antonio Omaña y Ortuño, Teresa. 2003.** *SEDD, Un Sistema Experto para el Diagnóstico en Desastres y Emergencias*. Madrid, España: s.n., 2003.
7. (Belmonte Fernández, 2005) **Belmonte Fernández, Oscar. 2005.** *Introducción al lenguaje de programación Java: Una guía básica*. 2005.

8. (Bregón y otros, 2005) **Bregón, Anibal , y otros. 2005.** *Un sistema de razonamiento basado en casos para la clasificación de fallos en sistemas dinámicos.* Valladolid : s.n., 2005. 84-9732-449-8.
9. (Campos Perales y otros, 2011) *La formación del profesional desde una concepción personalizada del proceso de aprendizaje.* **Campos Perales, Vilma y Moya Ricardo, Raubel. 2001.** 28, Guantánamo: s.n., 2011, Vol. 3.
10. (Cañizares González, 2012) **Cañizares González, Roxana. 2012.** *REPOSITORIO DE RECURSOS EDUCATIVOS PARA LAS INSTITUCIONES DE EDUCACIÓN SUPERIOR.* La Habana, Cuba: s.n., 2012.
11. (Cárdenas Montes, 2014) **Cárdenas Montes, Miguel. 2014.** *K-Nearest Neighbors. Gráficas, estadística y minería de datos con python.* Madrid, España: s.n., 2014.
12. (Carlos C. y otros, 2002) **Carlos C., Amano, Anderson, Thor y Barkman, Patricia. 2002.** *Estándar para Metadatos de Objetos Educativos.* Instituto de Ingenieros Electrónicos y Electrónicos (IEEE). New York, NY 10016-5997, USA: s.n., 2002.
13. (Castro, 2015) **Castro, Luis. 2015.** [En línea] 2015. [Citado el: 27 de Mayo de 2015.] <http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Xml.htm>.
14. (Celaya Ramírez y otros, 2010) *Revista mexicana de investigación educativa.* **Celaya Ramírez, Rosario, Lozano Martínez, Fernando y Ramírez Montoya, María Soledad. 2010.** 45, México: s.n., 2010, Vol. 15. 1405-6666.
15. (Charnelli, 2013) **Charnelli, María Emilia. 2013.** *Integrando repositorios digitales de recursos educativos abiertos con plataformas virtuales de aprendizaje.* Argentina: s.n., 2013.
16. (Chaveco Bustamante y otros, 2013) **Chaveco Bustamante, Yordi , Roque Hernández, Yoslenys y Vilches Pupo, Jesús . 2013.** *Sistema web inteligente apoyado en mapas*

conceptuales para la asignatura de Álgebra Lineal en la Universidad de las Ciencias Informáticas. La Habana : s.n., 2013.

17. (Chipia Lobo, 2015) **Chipia Lobo, Joan Fernando. 2015.** Slideshare. *Slideshare*. [En línea] 2015. [Citado el: 7 de Febrero de 2015.] <http://es.slideshare.net/JoanFernandoChipia/analisis-de-la-importancia-del-uso-de-metodologas-de-desarrollo-y-mtricas-de-calidad-de-software-en-aplicaciones-educativas-3565076>.
18. (Codina, 2000) *Evaluación de recursos digitales en línea*. **Codina, L. 2000.** 1, 2000, Vol. 23. 0210-0614.
19. (Colomé Cedeño, 2013) **Colomé Cedeño, Dunia María. 2013.** *Ambiente de trabajo para la producción de Objetos de Aprendizaje en la Educación superior*. La Habana: s.n., 2013.
20. (Cooperman, 2014) **Cooperman, Larry. 2014.** *La promesa de la educación abierta es la educación superior universal*. 2014.
21. (Corona Prendes y otros, 2013) **Corona Prendes, Frank David y Trujillo Irragorri, Luis Enrique. 2013.** *SISTEMA BASADO EN CASOPARA PREDECIR LA OCURRENCIA DE REACCIONES ADVERSAS A MEDICAMENTOS EN LA CONSULTA MÉDICA*. La Habana : s.n., 2013.
22. (Corzo, 2001) **Corzo, Yuliana . 2001.** *La Lógica Difusa*. Porlamar, Venezuela : s.n., 2001.
23. (Cruz Rojas y otros, 2009) **Cruz Rojas, Orlando y Puig González, Claudio. 2009.** *Sistema Basado en Casos para la asistencia al proceso de soporte y ayuda en la gestión de incidencias tecnológicas en la Universidad de las Ciencias Informáticas* . La Habana, Cuba : s.n., 2009.

24. (D' Negri y otros, 2006) **D' Negri, Carlos Eduardo y De Vito, Eduardo Luis. 2006.** *Introducción al razonamiento aproximado: lógica difusa.* Argentina : s.n., 2006. 126-136.
25. (de Vergara, 2001) **de Vergara, Evergisto. 2001.** *Seminario de Interoperabilidad.* 2001.
26. (Domínguez Gómez y otros, 2012) Revista Cubana de Informática Médica. **de la Torre Navarro, Lilia María y Domínguez Gómez, José. 2012.** 1, Ciudad de la Habana: s.n., 2012, Vol. IV. 1684-1859.
27. (Edel Navarro, 2015) *El concepto de enseñanza aprendizaje.* **Edel Navarro, Dr. Rubén. 2015.** 2015. 1579-0223.
28. (Fdez-Riverola y otros, 2003) **Fdez-Riverola, F. y Corchado, J. M. . 2003.** *Employing TSK Fuzzy models to automate the revision stage of a CBR system.* Salamanca : s.n., 2003.
29. (Febles Díaz, 2012) SI-Holmes: aplicación compuesta para la gestión de sistemas basados en conocimiento. **Febles Díaz, Orestes, y otros. 2012.** 3, La Habana, Cuba: s.n., 2012, Vol. 23. 1024-9435.
30. (Gema Santos y otros, 2012) *Recursos educativos abiertos: Repositorios y uso.* **Gema Santos, Hermosa, Ferran-Ferrer, Núria y Abadal, Ernest. 2012.** 2, Barcelona: El profesional de la información, 2012, Vol. 21. 1386-6710.
31. (Giménez Arjona, 2006) **Giménez Arjona, Manuel Miguel. 2006.** Estudio para la implementación de un sistema de razonamiento basado en casos. Tarragona, España: s.n., 2006.
32. (González Duque, 2015) **González Duque, Raúl . 2015.** Mundo geek. [En línea] 2015. [Citado el: 28 de Mayo de 2015.] <http://mundogeek.net/archivos/2009/03/16/colecciones-en-java/>.

33. (Guioteca, 2011) **Guioteca, 2011.** Guioteca. [En línea] 2011. [Citado el: 5 de Febrero de 2015.] <http://www.guioteca.com/educacion-secundaria/ensenanza-aprendizaje-3-etapas-clave-para-tener-exito/>.
34. (Hansen y otros, 2000) **Hansen, Bjarne K. y Riordan, Denis . 2000.** *Weather Prediction Using Case-Based Reasoning.* Canada : s.n., 2000.
35. (Hernández Orallo y otros, 2004) **Hernández Orallo, José Quintana y Ramírez Ferry Ramírez, César José. 2004.** *Introducción a la minería de datos.* Madrid, España : s.n., 2004.
36. (Hunter y otros, 2015) **Hunter, Jason y McLaughlin, Brett . 2015.** JDOM v2.0.6 API Specification. [En línea] 2015. [Citado el: 31 de Mayo de 2015.] <http://www.jdom.org/docs/apidocs/>.
37. (IBM Corporation, 2012) **Corporation, IBM. 2012.** IBM Knowledge Center. [En línea] 2012. [Citado el: 18 de junio de 2015.] http://www-01.ibm.com/support/knowledgecenter/SSLVMB_21.0.0/com.ibm.spss.statistics.help/idh_idd_tree_validation.htm?lang=es.
38. (Izquierdo, 2008) **Izquierdo, Susuna. 2008.** Abartia Team. [En línea] 2008. [Citado el: 16 de Abril de 2015.] <http://www.abartiateam.com/interoperabilidad>.
2. (Java, 2010) **Java. 2010.** Java Foundation. [Online] Julio 2, 2010. [Cited: Junio 11, 2015.] <http://javafoundations.blogspot.com/2010/07/java-estandares-de-programacion.html>.
39. (Kuramoto, 2010) **Kuramoto, H. 2010.** *Ciencia Abierta. Un desafío regional.* Buenos Aires: s.n., 2010.
40. (López Rodríguez y otros, 2002) La técnica de ladov. **López Rodríguez, Alejandro y González Maura, Viviana. 2002.** 47, Buenos Aires, Argentina: s.n., 2002.

41. (Lozano y otros, 2005) **Lozano, Laura y Fernández, Javier. 2005.** Razonamiento Basado en Casos: Una Visión General. Valladolid, España: s.n., 2005.
42. (M. Boneu, 2007) *Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos.* **M. Boneu, Jose. 2007.** 1, 2007, Vol. 4. 1698-580x.
43. (M. Boneu, 2007) Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos. **M. Boneu, Josep. 2007.** 1, 2007, Vol. 4. issn 1698-580x.
44. (Martínez Sánchez y otros, 2008) **Martínez Sánchez, Natalia, y otros. 2008.** *El Razonamiento Basado en Casos en el ámbito de la Enseñanza/Aprendizaje.* s.l.: Revista de Informática Educativa y Medios Audiovisuales, 2008. 1667-8338.
45. (Menéndez Domínguez, y otros) **Menéndez Domínguez, Victor Hugo y Prieto Méndez, Manuel Emilio.** *Herramientas de Gestión para Objetos de Aprendizaje.* AGORA.
46. (Molina López y otros, 2006) **Molina López, José Manuel y García Herrero, Jesús . 2006.** *Técnicas de análisis de datos.* 2006.
47. (Mortera Jiménez, 2013) **Mortera Jiménez, Fernando Jorge. 2013.** *Buenas prácticas para el uso académico de Recursos Educativos Abiertos y Objetos de Aprendizaje.* Monterrey, Costa Rica: s.n., 2013.
48. (Navarro Marset, 2007) **Navarro Marset, Rafael. 2007.** *REST vs Web Services.* 2007.
49. (Nesbit y otros, 2003) **Nesbit, John y Belfer, Karen. 2003.** Learning Object Review Instrument. 2003.
50. (Netbeans, 2014) **Netbeans. 2014.** Desarrolloweb.com. *Desarrolloweb.com.* [En línea] 2014. [Citado el: 7 de Febrero de 2015.]

- <http://www.desarrolloweb.com/actualidad/netbeans-6-9-final-liberado-3613.html>.
51. (Oracle Corporation and/or its affiliates, 2015) **Oracle Corporation and/or its affiliates. 2015.** NetBeans. *NetBeans*. [En línea] 2015. [Citado el: 6 de Abril de 2015.] <https://netbeans.org/community/releases/80/>.
52. (Peña Ayala, 2006) **Peña Ayala, Alejandro. 2006.** *Sistemas basados en Conocimiento: Una Base para su Concepción y desarrollo*. Mexico, D.F.: Dirección de Publicaciones. Revillagigedo 83, Centro Histórico, 06070, México, D.F., 2006. 970-94797-4-1.
53. (Pereira y otros, 2010) **Pereira, B., Oliveira, I. D. y Campos, B. D. 2010.** Contenidos Educativos Digitales. [En línea] 2010. [Citado el: 18 de Abril de 2015.] [http://www.iiisci.org/journal/CV\\$/risici/pdfs/NK516OL.pdf](http://www.iiisci.org/journal/CV$/risici/pdfs/NK516OL.pdf).
54. (Pérez Bello, 2002) *Aplicaciones de la Inteligencia Artificial*. **Pérez Bello, Rafael E. 2002.** Guadalajara, Jalisco: Ediciones de la Noche, 2002. 970-27-0177-5.
55. (Peris Soler y otros, 2004) **Peris Soler, Pablo y Belenguer Navarro, Noemí. 2004.** Servicios Web. Valencia: s.n., 2004.
56. (Quintana y otros, 2004) **Quintana, Mariledy y Marrero, Mercedes. 2004.** *La gestión ambiental dentro de la planificación estratégica de la ciudad de Matanzas*. Matanzas: s.n., 2004.
57. (Ramírez Cabrera, 2013) **Ramírez Cabrera, José Luis . 2013.** *Análisis comparativo DBDT vs otros Algoritmos para el manejo de datos no escalares*. Valencia, España : s.n., 2013.
58. (Reino Romero, 2000) **Reino Romero, Alfredo . 2000.** Introducción a XML en castellano. [En línea] 2000. [Citado el: 27 de Mayo de 2015.] <http://www.asptutor.com/zip/xml.pdf>.

59. (Reyes García y otros, 2014) **Reyes García, Jorge y Kindelán Colmenero, Julio César. 2014.** *SISTEMA BASADO EN REGLAS PARA DETECTAR INTERACCIÓN MEDICAMENTOSA DESDE LA.* La Habana : s.n., 2014.
60. (Risso Patrón, 2003) **Risso Patrón, Hernán José María. 2003.** *La Interoperabilidad.* Buenos Aires: s.n., 2003.
61. (ROA, 2010) **ROA. 2010.** Repositorio de Objetos de Aprendizaje. [En línea] 2010. [Citado el: 26 de Mayo de 2015.] roa.uci.cu.
62. (Rodríguez, 2010) *Sistema experto basado en casos para el diagnóstico de la hipertensión arterial.* **Cuadrado Rodríguez, Santiago. 2010.** 60, Colombia: Revista de Ingeniería de la Universidad de Antioquía, 2010. 0120-6230.
63. (Romero y otros, 2012) *Uso de las TIC en el proceso de enseñanza aprendizaje.* **Romero, Sandy y Araujo, Dory. 2012.** 1, Colombia: s.n., 2012, Vol. 11.
64. (Sánchez Guerrero, 2009) **Sánchez Guerrero, María de Lourdes. 2009.** *Sistema de Aprendizaje Inteligente con Objetos de Aprendizaje. "ProgEst".* Mexico, D.F.: s.n., 2009.
65. (Serrano Islas y otros, 2013) *Objetos de Aprendizaje.* **Serrano Islas, María de los Ángeles. 2013.** 2013.
66. (Takagi y otros, 1985) **Takagi, T. y Sugeno, M. . 1985.** *Fuzzy identification of systems and its application to modeling and control.* 1985.
67. (Toll Palma y otros, 2011) *La Calidad de los Objetos de Aprendizaje producidos en la Universidad de las Ciencias Informáticas.* **Toll Palma, Yuniet del Carmen, y otros. 2011.** 36, 2011. 1135-9250.

68. (UNESCO, 2002) **UNESCO. 2002.** Forum on the impact of open courseware for higher. [En línea] 2002. [Citado el: 3 de Noviembre de 2014.] <http://www.wcet.info/resources/publications/unescofinalreport.pdf>.
69. (Vega, 2000) *Razonamiento basado en casos aplicado al diagnóstico de generadores eléctricos.* **Vega, H. Octavio de la Torre y López A., Ernesto. 2000.** México: Boletín iie, 2000.
70. (Velázquez López, 2010) **Velázquez López, Luisa. 2010.** Estructura de un Sistema Experto. [En línea] Instituto de Investigaciones en Informática, 19 de Agosto de 2010. [Citado el: 9 de Febrero de 2015.] http://iii.informatica.edu.bo/index.php?option=com_content&view=article&id=106:eduse&catid=43:revista2007&Itemid=60.
71. (Vílchez Quesada, 2007) **Vílchez Quesada, Enrique. 2007.** Cuadernos de Investigación y Formación en educación matemática. *Sistemas Expertos para la Enseñanza y el Aprendizaje de la matemática en la Educación Superior.* 2007. 3.

Anexos

Anexo 1: Instrumentos de evaluación de recursos educativos

Instrumento de evaluación	Indicadores de evaluación
LORI	Calidad de contenido, alineamiento de objetivos de aprendizaje, retroalimentación y adaptación, motivación, diseño de presentación, usabilidad en interacción, accesibilidad, reusabilidad y cumplimiento de estándares.
MERLOT	Calidad de contenido, efectividad potencial y facilidad de uso.
Evaluación Pedagógica de Reeves	Epistemológica, filosofía pedagógica, sustento psicológico, orientación a objetivos, validez experimental, rol del instructor, flexibilidad de programa, valor del error, motivación, adaptación a diferencia a individuales, control de aprendizaje, actividades de usuario, aprendizaje cooperativo y sensibilidad cultural.
HEODAR	Criterios Pedagógicos: categoría psicopedagógica y categoría didáctico curricular; Criterios de Usabilidad: diseño de interfaz y diseño de navegación
COdA	Objetivos y coherencia didáctica, calidad de los contenidos, capacidad de generar reflexión, crítica e innovación, interactividad y adaptabilidad, motivación, formato y diseño, usabilidad, accesibilidad, reusabilidad e interoperabilidad

Anexo 2: Categorías del instrumento de revisión de recursos educativos LORI con sus respectivos problemas de diseño.

Categorías	Problemas de diseño
Calidad de los contenidos	✓ El contenido es incorrecto.

	<ul style="list-style-type: none"> ✓ El contenido presenta omisiones o sesgos. ✓ El nivel de detalle es inadecuado. ✓ Las presentaciones no refuerzan los puntos clave ni las ideas significativas. ✓ La información muestra sesgos en la representación de grupos étnicos o culturas.
Adecuación de los objetivos de aprendizaje	<ul style="list-style-type: none"> ✓ Los objetivos de aprendizaje no son evidentes. ✓ No existe una relación entre los contenidos, las actividades y las evaluaciones. ✓ Los objetivos no son apropiados para el perfil de alumnado al que se dirige.
Feedback (Retroalimentación) y Adaptabilidad	<ul style="list-style-type: none"> ✓ No aporta <i>feedback</i> relativo a la calidad o corrección de las respuestas del alumnado. ✓ No mantiene un registro de las respuestas o del estilo de aprendizaje del alumno/a de cara a adaptar las siguientes presentaciones de contenido. ✓ No hay simulación o un conjunto de herramientas que se adapten en función de las respuestas del alumno/a.
Motivación	<ul style="list-style-type: none"> ✓ El contenido no es relevante en la vida del alumnado. ✓ Las actividades son demasiado fáciles o demasiado difíciles para percibir el interés del alumnado. ✓ Las características del objeto que suponen captar el interés del alumno son una distracción que interfiere con el aprendizaje. ✓ El objeto no varía en el tono, no hay muestras de humor o novedad. ✓ La interacción del alumno/a con el objeto no presenta consecuencias interesantes.
Diseño y presentación	<ul style="list-style-type: none"> ✓ La fuente seleccionada o el tamaño de la misma reduce notablemente la velocidad de lectura. ✓ Cierta información que es necesaria para la comprensión del objeto de aprendizaje es ilegible.

	<ul style="list-style-type: none"> ✓ La calidad del vídeo o audio es insuficiente para el aprendizaje. ✓ La selección de colores, imágenes o sonidos interfieren con los objetivos de aprendizaje. ✓ El diseño de la información provoca un sobreesfuerzo innecesario para el procesamiento de la misma. ✓ No hay suficientes encabezados o no son significativos para el alumno/a.
Usabilidad	<ul style="list-style-type: none"> ✓ No hay posibilidad de interactividad, solo hay texto y/o imágenes estáticas. ✓ Varios enlaces o botones no funcionan. ✓ Hay una demora excesiva en la navegación. ✓ El funcionamiento de la interfaz no es intuitivo y tampoco hay instrucciones. ✓ El funcionamiento de la interfaz es inconsistente e impredecible.
Accesibilidad	<ul style="list-style-type: none"> ✓ Los videos no tienen subtítulos. ✓ No hay transcripciones para los archivos de audio. ✓ Faltan etiquetas en las imágenes. ✓ La comprensión de los gráficos requiere que el alumno/a pueda percibir los colores.
Reusabilidad	<ul style="list-style-type: none"> ✓ Se refiere al módulo, curso o docente para el que fue diseñado originalmente. ✓ Su uso depende críticamente de recursos de aprendizaje específicos y externos. ✓ Sólo puede ser utilizado por un grupo pequeño de alumnos/as con un nivel de conocimiento previo alto y específico.
Cumplimiento de estándares	<ul style="list-style-type: none"> ✓ No se provee suficiente metadata o no está formateado de acuerdo al estándar de objetos de aprendizaje del IEEE. ✓ Fallan todos los tests de cumplimiento de directrices del W3C y SCORM (directrices de accesibilidad no incluidas).

Anexo 3: Servicios de recolección de metadatos brindados por el repositorio RHODA usando la URL <http://10.55.11.230/roa/web/roa.php/interoperability/request> y el protocolo REST.

Nombre del método	Entrada	Salida
Identify	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=Identify	XML plano con la información del servidor.
ListRecords	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=ListRecords&metadataPrefix=oai_dc&from=2015-04-27&until=2015-05-27	XML plano con un listado de los metadatos de los Objetos de aprendizaje, que cumplen con los parámetros de selección (<i>set, from, until</i> y <i>metadataPrefix</i>).
ListSets	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=ListSets&identifier=oai:Rhoda_1587	XML plano con un listado de las colecciones soportadas por el repositorio.
GetRecord	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=GetRecord&metadataPrefix=oai_dc&identifier=oai:Rhoda_1587	XML plano con los metadatos de un recurso que cumple con los parámetros (<i>identifier</i> y <i>metadataPrefix</i>).
ListMetadata Records	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=ListMetadataFormats&identifier=oai:Rhoda_1587	XML plano con un listado de los formatos de catalogación de Objetos de aprendizaje soportados por el repositorio.
ListIdentifiers	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=ListIdentifiers&metadataPrefix=oai_dc&from=fecha&until=fecha	XML plano con un listado de las cabeceras de los Objetos de aprendizaje, que cumplen con los parámetros de selección (<i>set, from, until</i> y <i>metadataPrefix</i>).

Anexo 4: Servicios de revisión brindados por el repositorio RHODA usando la URL <http://10.55.11.230/roa/web/roa.php/interoperability/request> y el protocolo REST.

Nombre del método	Entrada	Salida
-------------------	---------	--------

GetRecordReview	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=GetRecordReview&identifier=oai:RHODA_1587	XML plano con los valores de revisión del recurso que cumple con el parámetro <i>identifier</i>
ListRecordsReview	http://10.55.11.230/roa/web/roa.php/interoperability/request?verb=ListRecordsReview&from=2015-04-27&until=2015-05-27	XML plano con un listado de las revisiones, que cumplen con los parámetros de selección (<i>from, until</i>)

Anexo 5: Esquema de metadatos del estándar LOM.

Categoría	Elementos de metadatos	Explicación	Tipo de dato
General	Identificador	Identificador descriptivo del material educativo. Su valor debe identificar unívocamente el material en su contexto educativo.	Numeric
	Título	Nombre descriptivo del material educativo.	String
	Entrada en catálogo	Entrada en un determinado catálogo. El valor para este metadato debe ser un par formado por un nombre de catálogo, así como por el nombre de la entrada en dicho catálogo. Este metadato puede especificarse para seleccionar el recurso cuando éste se encuentra indexado en un catálogo externo.	String
	Idioma	El idioma primario utilizado en el material para comunicarse con los potenciales consumidores del mismo.	String
	Descripción	Texto describiendo el contenido del material.	String
	Palabra clave	Colección de frases que representan palabras clave sobre el material.	String
	Covertura	Eventos temporales, culturales o geográficos asociados con el material.	1823

	Estructura	La estructura interna del material.	colección, mixta, lineal, jerárquica, en_red, ramificada, compartimentada, atómica
	Nivel de agregación	<p>Define la granularidad del material. Ejemplo:</p> <ol style="list-style-type: none"> 1. Representa el nivel más pequeño de agregación (el aplicable a material aparentemente indivisible, como una imagen, un archivo PDF, etc.). 2. Colección de materiales atómicos (por ejemplo, un archivo HTML junto con las imágenes referidas desde el mismo). 3. Una colección de dos o más materiales de nivel 2 (por ejemplo, una web formada por múltiples documentos HTML). 4. El nivel mayor de granularidad (por ejemplo, un conjunto de cursos que conducen a la obtención de un grado). 	Numeric
Ciclo de vida	Versión	La edición o versión del material.	Numeric
	Estado	El estado de producción del material.	borrador, final, revisado, no_disponible
	Participantes	Introduce información acerca de un contribuyente a la producción del material. De esta forma, un mismo material puede tener asociados múltiples contribuyentes.	String
Meta-metadatos	Identificador	Identificador del conjunto de metadatos para el recurso. Este identificador puede utilizarse para	Numeric

		seleccionar el conjunto de metadatos, cuando éste se encuentra almacenado externamente.	
	Catálogo de entrada	Un catálogo y una entrada en dicho catálogo en el que el conjunto de metadatos para el recurso reside. Esto permite seleccionar los metadatos de un catálogo externo.	String
	Participantes	Contribuyente a la elaboración de estos metadatos. Para cada contribuyente es posible especificar, al igual que en la categoría lifecycle, el rol, la identidad y la fecha. LOM proporciona un vocabulario controlado para el rol, que, en este caso, puede ser: creador y validador.	String
	Esquema de metadatos	Esquema de metadatos utilizado	String
	Idioma	El idioma por defecto utilizado para proporcionar los metadatos.	String
Requisitos técnicos	Formato	Formato del material. Dado que el material no tiene por qué ser atómico, es posible que integre múltiples formatos (por ejemplo, una página web puede integrar un documento HTML con un conjunto de imágenes JPG), por lo que un mismo material puede exhibir múltiples metadatos format. Una manera adecuada de describir los formatos es mediante su denominación MIME (ver MIME 1996).	String
	Tamaño	Tamaño en bytes del material.	Numeric
	Localización	Forma de localizar al material (por ejemplo, una URL, o una descripción textual acerca de cómo llevar a cabo dicha localización).	String
	Requisito	Plataforma informática necesaria para utilizar este material.	String

	Indicaciones de instalación	Notas de instalación para el recurso.	String
	Otros requisitos de plataforma	Otros requisitos software y hardware.	String
	Duración	Duración (únicamente para material para el que tenga sentido una duración en su reproducción, como, por ejemplo, un video o una presentación Flash)	Numeric
Educativa	Tipo de interacción	Tipo de interacción soportado por el material.	interactivo, pasivos, mixto
	Tipo de recurso educativo	Especifica el tipo de material (por ejemplo, ejercicio, figura, etc.). Un mismo material puede tener distintos tipos asociados.	ejercicio, simulación, cuestionario, diagrama, figura, gráfico, índice, diapositiva, tabla, texto narrativo, examen, experimento, enunciado de problema, autoevaluación.
	Nivel de interacción	Especifica el nivel de interacción del material.	muy_bajo, bajo, medio, muy_alto, alto
	Densidad semántica	Una medida subjetiva de la utilidad educativa del material en comparación con su tamaño y/o duración.	muy_bajo, bajo, medio, muy_alto, alto
	Papel jugado por el supuesto usuario	Determina el papel del usuario final del material.	maestro, autor, aprendiz, gestor.

	Contexto	El entorno educativo típico en el que se usará el material.	educación primaria, educación secundaria, educación superior, primer ciclo universitario, segundo ciclo universitario, postgrado, primer ciclo de escuela técnica, segundo ciclo de escuela técnica, formación profesional, formación continua, formación vocacional.
	Segmento de edad típico	Rango de edades típico de los usuarios a los que va dirigido el material.	adolescente, adulto, joven
	Dificultad	Grado de dificultad del material.	muy_fácil, fácil, medio, muy_difícil, difícil
	Tiempo típico de aprendizaje	Tiempo de aprendizaje típico asociado con el material.	Numeric
	Descripción	Comentarios sobre el uso del material desde un punto de vista pedagógico.	String
	Idioma	Idioma del usuario final.	String
Derechos de uso	Coste	Establece si el recurso es o no de pago.	sí, no
	derechos de copia y otras restricciones	Establece si el recurso está o no sujeto a derechos de copia y otras restricciones.	sí, no
	descripción	Comentarios sobre las condiciones y derechos de uso de este recurso	String

Relaciones	Tipo de relación	LOM propone el siguiente vocabulario controlado para caracterizar dicha clase: isPartOf (el material es parte de otro más complejo), hasPart (el material tiene a otro como parte integrante), isVersionOf (el material es una versión de otro), hasVersion (el material tiene a otro como una versión), isFormatOf (el material es la descripción de un formato de otro material), hasFormat (el material tiene a otro como formato), references (el material refiere al otro), isReferencedBy (el material está referido por el otro), isBasedOn (el material está basado en otro), isBasisFor (el material es la base de otro), requires (el material requiere la presencia de otro), isRequiredBy (el material es requerido por otro).	isPartOf, hasPart, isVersionOf, isFormatOf, hasFormat, references, isReferencedBy, isBasedOn, isBasisFor, requires, isRequiredBy
Anotación	Entidad	El anotador que realiza la anotación.	String
	Fecha	La fecha de la anotación.	String
	Descripción	El texto en sí de la anotación	String
Clasificación	Propósito	El propósito de la clasificación.	disciplina, idea, prerrequisito, objetivo educativo, restricciones de acceso, nivel educativo, nivel de destreza, nivel de seguridad.
	Ruta en un sistema de clasificación	Una serie de rutas en distintas taxonomías.	String
	Descripción	Una descripción textual del material relativa al propósito de clasificación establecido.	String

	Palabras clave	Un conjunto de palabras clave relativas al propósito de clasificación establecido.	String
--	----------------	------------------------------------------------------------------------------------	--------

Anexo 6: Encuesta realizada a especialistas para asignar valores numéricos a las relaciones de comparación entre los posibles valores discretos que puede adoptar determinada categoría del estándar de metadatos LOM.

✓ **Nivel de agregación**

Valores: 1, 2, 3, 4

\\%-simil.\\	1	2	3	4
1				
2				
3				
4				

✓ **Estado**

Valores: borrador, final, revisado, no disponible

\\%-simil.\\	borrador	final	revisado	no disponible
borrador				
final				
revisado				
no disponible				

✓ **Tipo de interacción**

Valores: activo, expositivo, mixto, indefinido

\\%-simil.\\	activo	expositivo	mixto	indefinido
activo				
expositivo				
mixto				
indefinido				

✓ **Tipo de recurso educativo**

Valores: ejercicio (1), simulación (2), cuestionario (3), diagrama (4), figura (5), gráfico (6), índice (7), diapositiva (8), tabla (9), texto narrativo (10), examen (11), experimento (12), enunciado de problema (13), autoevaluación (14)

\\%-simil.\\	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														

✓ **Nivel de interacción**

Valores: muy bajo, bajo, medio, alto, muy alto

\\%-simil.\\	muy bajo	bajo	medio	alto	muy alto
muy bajo					
bajo					
medio					
alto					
muy alto					

✓ **Densidad semántica**

Valores: muy bajo, bajo, medio, alto, muy alto

\\%-simil.\\	muy bajo	bajo	medio	alto	muy alto
muy bajo					
bajo					

medio					
alto					
muy alto					

✓ **Papel jugado por el supuesto usuario**

Valores: maestro, autor, aprendiz, gestor.

\\%-simil.\\	maestro	autor	aprendiz	gestor
maestro				
autor				
aprendiz				
gestor				

✓ **Contexto**

Valores: educación primaria (1), educación secundaria (2), educación superior (3), primer ciclo universitario (4), segundo ciclo universitario (5), postgrado (6), primer ciclo de escuela técnica (7), segundo ciclo de escuela técnica (8), formación profesional (9), formación continua (10), formación vocacional (11)

\\%-simil.\\	1	2	3	4	5	6	7	8	9	10	11
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											

✓ **Dificultad**

Valores: muy fácil, fácil, medio, difícil, muy difícil

\\%-simil.\\	muy fácil	fácil	medio	difícil	muy difícil
muy fácil					

fácil					
medio					
difícil					
muy difícil					

✓ **Tipo de relación**

Valores: esParteDe (1), fueParte (2), esVersionDe (3), fueVersion (4), esFormatoDe (5), fueFormato (6), referencias (7), esReferenciadoPor (8), esBasadoEn (9), esBasePara (10), requiere (11), esRequeridoPor (12)

\\%-simil.\\	1	2	3	4	5	6	7	8	9	10	11	12
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												

✓ **Propósito**

Valores: disciplina (1), idea (2), prerrequisito (3), objetivo educativo (4), restricciones de acceso (5), nivel educativo (6), nivel de destreza (7), nivel de seguridad (8)

\\%-simil.\\	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								
7								

Anexo 7: Implementación en lenguaje Java de la clase *FuzzyAdaptation*.

```
import CaseBaseModel.LoriEvaluation;
import CaseBaseModel.REA;
import RecoveryModule.SimilarItems;
import RecoveryModule.SimilarityDescription;
import java.io.IOException;
import java.util.Collections;
import java.util.HashMap;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.logging.Level;
import java.util.logging.Logger;
import org.jdom2.JDOMException;

public class FuzzyAdaptation {

    public EvaluationResponseDescription AdaptationByFuzzyLogic(SimilarItems similares) {
        List<HashMap<String, EvaluationResponseDescription>> primeros5 = new LinkedList<>();

        for (int i = 1; i < 6; i++) {
            primeros5.add(getSimilarEvaluation(similares.getByRelativeRank(i)));
        }
    }
}
```

```
}  
LoriEvaluation lori = new LoriEvaluation();  
LinkedList<EvaluationResponseDescription> _final = new LinkedList<>();  
  
float sum = 0;  
int cont = 0;  
float avg = 0;  
String cat;  
Iterator it = primeros5.iterator();  
for (int i = 0; i < 9; i++) {  
    cat = lori.getCategoryName(i);  
    while (it.hasNext()) {  
        HashMap<String, EvaluationResponseDescription> rea = (HashMap<String,  
EvaluationResponseDescription>) it.next();  
        if (rea.get(cat) != null) {  
            float cert = (float) rea.get(cat).getCertidumbre();  
            sum += cert;  
            cont++;  
        }  
    }  
}  
avg = sum / cont;  
_final.add(new EvaluationResponseDescription(cat, LoriEvaluation.evaluation.MAL, avg));  
sum = 0;  
cont = 0;  
avg = 0;  
it = primeros5.iterator();
```

```
    }
    Collections.sort(_final);
    return _final.getLast();
}

public HashMap<String, EvaluationResponseDescription> getSimilarEvaluation(SimilarityDescription
similar) {
    REA k_1 = similar.getItem();
    LoriEvaluation evaluation = k_1.getEvaluation();
    float similaridad = similar.getPercentSimilarity();

    HashMap<String, EvaluationResponseDescription> ret = new LinkedHashMap<>();
    Set keys = evaluation.getCategories().keySet();
    Iterator it = keys.iterator();

    while (it.hasNext()) {
        Integer key = (Integer) it.next();
        String cat = evaluation.getCategoryName(key);
        LoriEvaluation.evaluation eval = evaluation.getEvaluation(key);
        ret.put(cat, new EvaluationResponseDescription(cat, eval, similaridad));
    }
    return ret;
}

public String getProblemsResponse(EvaluationResponseDescription eval) {
```

```
String cat = eval.getCatName();
float similitud = eval.getCertidumbre();
ProblemsDB problemas = null;
try {
    problemas = new ProblemsDB();
} catch (JDOMException ex) {
    Logger.getLogger(AdaptationByReInstantiation.class.getName()).log(Level.SEVERE, null, ex);
} catch (IOException ex) {
    Logger.getLogger(AdaptationByReInstantiation.class.getName()).log(Level.SEVERE, null, ex);
}
LoriEvaluation aux = new LoriEvaluation();
String resp = "Utilidad de recomendación según Adaptación por lógica fuzzy: " + similitud + "\n";
resp += problemas.getCategoryProblems(aux.getCatNameIndex(cat));

return resp;
}
}
```