

Tipo de artículo: Artículo original  
Temática: Ingeniería y Gestión de software – Inteligencia artificial  
Recibido: 11/03/2013 | Aceptado: 30/04/2013

## **Sistema de Razonamiento Basado en Casos para la identificación de riesgos de software**

### *System of Case-Based Reasoning to support software risk identification*

**Dasiel Cordero Morales**<sup>1\*</sup>, **Yadira Ruiz Constanten**<sup>2</sup>, **Yoanny Torres Rubio**<sup>3</sup>

<sup>1\*</sup> ISEC. Centro Informatización Seguridad Ciudadana. Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Torrens, Boyeros, La Habana, Cuba. CP.: 19370

<sup>2</sup> Facultad 2. Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Torrens, Boyeros, La Habana, Cuba. CP.: 19370. Correo-e: [yadirar@uci.cu](mailto:yadirar@uci.cu)

<sup>3</sup> Tlm. Departamento Desarrollo de componentes. Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Torrens, Boyeros, La Habana, Cuba. CP.: 19370. Correo-e: [ytrubio@uci.cu](mailto:ytrubio@uci.cu)

\*Autor para la correspondencia: [dcordero@uci.cu](mailto:dcordero@uci.cu)

---

#### **Resumen**

Un proyecto constituye una actividad progresiva, emprendida para crear un producto o servicio determinado. Por lo que su desarrollo necesita planificación y control debido a que requiere participación humana. Los proyectos de desarrollo de software se diferencian de otros proyectos en la naturaleza del producto pues el software se desarrolla, no se fabrica y como cualquier actividad humana, incluye la ocurrencia de riesgos. Estos son eventos o condiciones inciertas, que de producirse, afectan tanto positiva como negativamente los objetivos del proyecto. La automatización de los procesos relacionados con la gestión de riesgos garantiza la obtención de resultados que apoyen la toma de decisiones. Los sistemas basados en el conocimiento simulan las cadenas de razonamiento realizadas por expertos para resolver un problema determinado. Son utilizados en diversas áreas y enfocados a variadas temáticas. La complejidad para el desarrollo aplicaciones basadas en el conocimiento para la gestión de riesgos, radica en la forma de representar dicho conocimiento como elemento fundamental, que por su variedad, han surgido diferentes formas

de representarlo siendo las más difundidas los Sistemas Basados en Reglas, las Redes Expertas y el Razonamiento Basado en Casos (RBC). El objetivo de este trabajo es proponer una herramienta que utilizando razonamiento basado en casos se inserte en el proceso de mitigación de riesgos y apoye su tratamiento durante el desarrollo del software. La solución desarrollada combina principalmente un sistema de gestión de información con las técnicas de RBC. Constituye una aplicación web, basada en el modelo Cliente-Servidor desarrollada sobre la plataforma Java.

**Palabras clave:** Conocimiento, mitigación, razonamiento basado en casos, riesgo.

### **Abstract**

*A Project is a progressive activity, made to create some products or services. Therefore its development needs planning and control because it requires the human participation. The software development projects differ from other projects in the nature of the product because the software developed is not manufactured, and like any other human activity, includes the occurrence of risks. Risks are uncertain events or conditions, which can occur, so the objectives of the projects can be affected both positively and negatively. The automation of the process related to software and risk management guarantees the outcome to support decision-making. The Knowledge-based Systems simulate the reasoning chains made by experts to solve some problems. They are used in various areas and are focused on a huge variety of topics. The complexity for developing knowledge-based applications for risk management lies on how to represent this knowledge as main subject, that for its own variety, there have been different ways for representing it, the most widespread are Rule Based Systems, Experts Networks and Case-Based Reasoning (CBR). The objective of this paper is to propose a tool that, by using CBR, takes part in the risk mitigation process and supports their treatment during software development. The solution developed combines an information management system with CBR techniques. It is a web application based on a Client-Server model developed under java platform.*

**Keywords:** Case based reasoning, knowledge, mitigation, risks.

---

## **Introducción**

El Instituto de Gestión de Proyectos (*Project Management Institute PMI®*) es una organización internacional sin fines de lucro que asocia a profesionales para la gestión de proyectos (Project Management Institute, 2011). Es considerada una de las organizaciones más importantes del mundo en su rubro dado que se encuentra integrada por más de 260.000 miembros alrededor de 171 países (CIBERTEC, 2011). El mismo, basa sus ideas fundamentales en que: un proyecto es un esfuerzo temporal, único y progresivo, emprendido para crear un producto o un servicio también único (García, 2009). El desarrollo de un proyecto debe ser una actividad planificada y controlada, puesto que requiere

participación humana en un tiempo dado para su cumplimiento. La gestión de proyectos es la disciplina de organizar y administrar recursos de manera tal que se pueda culminar el trabajo requerido dentro del alcance, el tiempo, y costo definidos (Expert Soft, 2010), teniendo en cuenta que todo reposa sobre la base de obtener un producto con calidad. Los proyectos de desarrollo de software se diferencian de otros proyectos de ingeniería tradicional en la naturaleza lógica del producto. El software se desarrolla, no se fabrica en un sentido clásico. La gestión del proyecto de software es el primer nivel del proceso de ingeniería de software, porque cubre todo el proceso de desarrollo. Para lograr un proyecto de software fructífero se debe comprender el ámbito del trabajo a realizar, los riesgos en los que se puede incurrir, los recursos requeridos, las tareas a llevar a cabo, el esfuerzo a consumir y el plan a seguir.

Como cualquier actividad humana, el desarrollo de un proyecto incluye la ocurrencia de riesgos, los cuales son eventos o condiciones inciertas, que si se producen, afectan de manera positiva o negativa al menos un objetivo del proyecto (Pressman, 2010). Una acertada gestión de riesgos es la mejor manera de mitigar un riesgo o un conjunto de estos, ya que permite identificar, analizar y responder a los riesgos a lo largo de la vida de un proyecto, con el propósito de aumentar la probabilidad y el impacto de los eventos positivos y disminuir la de los eventos adversos. Este proceso dentro del ciclo de desarrollo de un proyecto de software es generalmente complejo, pues requiere seguimiento constante dependiendo en gran medida de las experiencias acumuladas, el conocimiento adquirido y los estándares definidos.

La automatización de los procesos relacionados con la gestión de riesgos garantiza la obtención de resultados que apoyen la toma de decisiones en aras de enfrentar, mitigar o aprovechar la ocurrencia de uno o varios riesgos. De este modo sería posible que un proyecto se desarrolle con la mayor calidad posible, cumpliendo con los alcances definidos en el menor período de tiempo y con la utilización del mínimo de recursos. Entre las técnicas más distintivas utilizadas actualmente para este fin se encuentran las que permiten predecir e identificar un resultado determinado simulando el razonamiento humano. A la rama de las Ciencias de la Computación dedicada al desarrollo o uso de los ordenadores, con los que se intenta reproducir los procesos de la inteligencia humana se le denomina Inteligencia Artificial (IA) (Gálvez, 1998).

En el Centro Informatización de la Seguridad Ciudadana (ISEC) de la Facultad 2 en la Universidad de las Ciencias Informáticas (UCI), dedicado al desarrollo de aplicaciones informáticas para diversos órganos e instituciones que brindan seguridad a los ciudadanos, la identificación y mitigación de riesgos a lo largo del ciclo de vida del software

es realizado por parte de la dirección de cada proyecto, utilizando diferentes criterios de estimación y metodologías de Gestión de Riesgos (GR). Este es un proceso largo y complejo, que se realiza de forma manual y que incluye el seguimiento y control de muchos de los integrantes del equipo de desarrollo. Actualmente el centro no cuenta con una herramienta que permita, ayude o agilice el proceso de gestión de riesgos en el ciclo de vida de los proyectos de desarrollo de software. Este trabajo provee un sistema inteligente con estos fines y de esta forma permite automatizar este proceso.

La aplicación informática desarrollada, “Sistema Inteligente de Mitigación de Riesgos”, relaciona los sistemas basados en casos con los procesos incluidos en la gestión de riesgos. El objetivo de dicho sistema es aprovechar las ventajas que brindan las técnicas de IA, para facilitar el manejo de algunos de los elementos distintivos en los proyectos de desarrollo de software, que permiten identificar la ocurrencia de un determinado evento, para su aprovechamiento o mitigación. Por esta razón, el sistema brinda la posibilidad de que a partir de un conjunto de características relevantes del proyecto que le son introducidas, se logre obtener de forma automática los riesgos potenciales a incidir a lo largo del ciclo de desarrollo, así como una posible mitigación o aprovechamiento de los mismos. Esto permite tener una visión adelantada, agilizando el proceso de toma de decisiones.

### Sistemas computarizados para la gestión de riesgos

Existen estrategias que se ocupan de agrupar los riesgos según su impacto sobre los objetivos del proyecto como se muestra en la Figura 1.

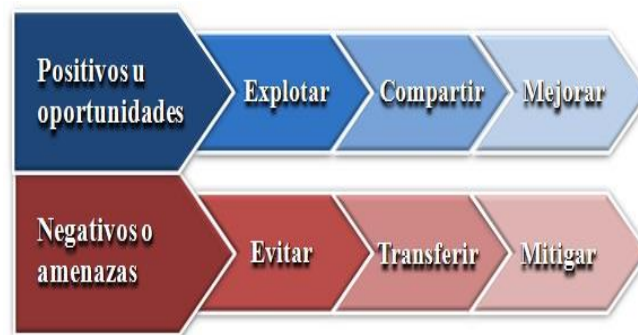


Figura 1. Estrategias para la gestión de riesgos.

Un modelo de GR es una guía que define las tareas necesarias para la identificación de los riesgos y el análisis de estos, para lo cual se tienen en cuenta las estrategias mostradas en la figura 1. También proporcionan una vía para

identificar factores potenciales de riesgos, así como medidas para reducirlos o lograr que impacten según su propia característica. Además proveen métodos y definen tareas para el aseguramiento de los proyectos.

La propuesta tiene sus bases sobre el Modelo de Gestión de Riesgos (MOGERI) definido para la UCI teniendo en cuenta las peculiaridades de su proceso. Recoge las prácticas adecuadas según el entorno donde debe aplicarse. Define procesos que permiten planificar las actividades sobre los riesgos, identificarlos, analizarlos, concebir las respuestas ante ellos, seguir y controlar los riesgos en el contexto del proyecto, así como comunicar la información generada al respecto (Zulueta, 2009).

Algunas soluciones informáticas a la problemática de los riesgos están orientadas a compañías maduras que poseen amplias bases de datos organizacionales que les permiten generar información de categorías propias de riesgos. Tal es el caso de *Risk Trak* y *Welcome Risk* que ayuda en la identificación, definición, estimación y análisis de las incertidumbres y así mejorar la competitividad (Risk Trak International, 2010). Otras, emplean un mecanismo que no se orienta al uso de clasificaciones de riesgos como es el caso de *Active Risk Manager ARM*. Esta última permite la identificación, análisis, tratamiento y seguimiento de los riesgos, oportunidades y problemas tanto cuantitativa como cualitativamente (Connection Acquisition Community, 2010). El Sistema de Mitigación e Identificación de Riesgos Técnicos / *Technical Risk Identification and Mitigation System (TRIMS)*, es una herramienta de software de gestión de riesgos disponible en el mercado que proporciona información sobre la gestión de riesgos técnicos, costos y cronograma de forma semejante a un sistema basado en el conocimiento (TRIMS Software, 2010). Sin embargo solo se enfoca en la categoría de riesgos técnicos.

A partir del análisis de estas herramientas, se puede concluir que son inadaptables a las necesidades del Centro de Informatización de la Seguridad Ciudadana (ISEC), pues son aplicaciones propietarias y los riesgos que gestionan son ajenos a los de ciclo de vida del software o se centran en etapas específicas del análisis de éstos. Por otra parte constituyen ejemplos de sistemas o herramientas software convencional que tratan riesgos específicos, según las características para las cuales se concibieron, pero no incluyen comportamiento o razonamiento inteligente.

La IA ayuda a la toma de decisiones a través de la utilización del conocimiento y la experiencia acumulada, ajustándola a la nueva situación y aprendiendo de cada escenario lo que se considere relevante, retroalimentándose con nuevo conocimiento. Este conocimiento pasa a formar parte del sistema, incorporándolo para su posterior utilización en su mecanismo de razonamiento, comparación y aprendizaje. Por tanto, para una buena gestión de

riesgos, es necesario utilizar este conocimiento para enfrentarse a los posibles riesgos a incidir en el ciclo de vida de desarrollo del software.

Una de las técnicas de IA más utilizadas para la concepción de este tipo de herramientas son los Sistemas Basados en el Conocimiento. Este tipo de técnica utiliza la madurez de las empresas, los estándares definidos, ejemplos anteriores, resultados obtenidos, datos de transacciones de recursos de cualquier tipo, experiencias en el campo donde se quiera implantar, entre otros. Esto permite el empleo de dicho contenido para dar apoyo a la toma de decisiones en la búsqueda de determinados comportamientos, así como resultados a ocurrir.

### **Sistemas basados en el conocimiento**

Los sistemas basados en el conocimiento (SBC), típicos del campo de la IA, no son más que programas para computadoras que simulan las cadenas de razonamiento que realiza un experto para resolver un problema de su dominio (Expósito, 2008). Para conseguirlo, se dota al sistema de un conjunto de principios o reglas que infieren nuevas evidencias a partir de la información previamente conocida.

Los SBC tienen ventajas y desventajas cuando se comparan con otras soluciones como el software convencional o expertos humanos. Esto viene aparejado a la adecuada integración con la gestión de riesgos pues poseen el conocimiento explícito y accesible, permitiendo una fácil modificación del mismo. Posibilitan incorporar nuevas experiencias adquiridas sobre la gestión de riesgos en otros escenarios o problemáticas pertenecientes a los riesgos de proyectos de software en este caso. Las respuestas dadas por este tipo de sistema son consistentes, ya que el conocimiento anterior suministrado a la base de conocimientos puede obtenerse de diferentes expertos o de resultados anteriores, de los cuales se pueden considerar los más adaptables.

La disponibilidad de estas aplicaciones propicia su utilización en el mayor espacio de tiempo posible, facilitando así que tras la llegada, creación, o concepción de un nuevo proyecto se puedan analizar, identificar y obtener posibles riesgos a incidir en el mismo, contribuyendo a la toma de decisiones. Unido a esto, coexiste la preservación de la experticia y la capacidad para adquirir nuevo conocimiento. De esta forma se perfecciona el que se tiene, se posibilita el aprendizaje de nuevos datos, así como la incorporación de otros riesgos o la variación de la ocurrencia de estos según diferentes escenarios.

Aunque por sus propias características, la relevancia de la utilización de los SBC es elevada, poseen algunas desventajas pues se parte de una herramienta basada en modelos matemáticos que arroja una solución sobre el dominio para el que fue concebido. Esto hace que el resultado no siempre sea el correcto y limite el conocimiento sólo al dominio de experticia definido, dependiendo del tamaño de la base de conocimientos y del aprendizaje definido para esta.

Este tipo de sistema, es utilizado en diversas empresas y enfocado a una amplia gama de temáticas. Los mismos intervienen y agilizan la toma de decisiones en el campo de las finanzas, el sector público, transacciones aduaneras, análisis de vulnerabilidades informáticas, manejo de datos personales entre otros. En cada una de estas áreas se garantiza la gestión de los diversos riesgos que pueden incidir en los escenarios en los cuales se enmarcan. La complejidad para el desarrollo de una aplicación basada en el conocimiento para la gestión de riesgos, viene dada por la forma de representar dicho conocimiento siendo este su elemento fundamental. Por la variedad de su representación, han surgido diferentes formas de representarlo, las más difundidas son los Sistemas Basados en Reglas (SBR), las Redes Neuronales Artificiales (RNA) y el Razonamiento Basado en Casos (RBC).

La dificultad de la elaboración de un SBR radica en la cantidad de reglas de producción a emplear según el dominio o características del problema a resolver. Aunque poseen gran modularidad, uniformidad en el conocimiento y naturalidad en el lenguaje de representación, estos sistemas no logran reconocer qué reglas aplicar en cada ciclo, perdiéndose así la perspectiva global del conjunto de reglas al tener que ser analizadas una a una. Además las propias características de la gestión de riesgos requieren de la utilización de muchas reglas.

Por otra parte en las RNA la adquisición del conocimiento incluye la selección de los ejemplos, el diseño de su topología y el entrenamiento de la red para hallar el conjunto de pesos. Facilitan el trabajo con información incompleta y brindan algoritmos poderosos de aprendizaje para crear la base de conocimiento; pero requieren de muchos ejemplos y son cajas negras que no explican cómo se alcanza la solución. Éstas generalmente utilizan pesos como forma de representar el conocimiento y el cálculo de niveles de activación de las neuronas como método de solución de problemas.

El RBC significa razonar sobre la base de experiencias o "casos" previos. Constituyendo una alternativa entre otras metodologías para construir sistemas basados en el conocimiento. El mismo denota un método donde la solución de un nuevo problema se realiza a partir de las soluciones conocidas para un conjunto de problemas previamente



resueltos o no, del dominio de aplicación. Este tipo de técnica, se distingue por utilizar directamente la información almacenada en la memoria del sistema a partir de casos ya resueltos. Para ello utiliza una memoria permanente o base de conocimientos en la cual se almacena de forma explícita la información sobre el dominio de aplicación. Los dominios abordados mediante técnicas RBC implican generalmente tareas de análisis, clasificación, interpretación, diagnóstico, diseño, planificación o asesoría.

Dentro de las herramientas inteligentes para la gestión de riesgos se encuentran el Sistema Inteligente de Administración de Riesgos SIAR desarrollado por Cotecna, una de las empresas mundiales líderes en materia de inspección de mercancías, seguridad y facilitación del comercio y certificación comercial, con sede en Suiza. Este sistema está diseñado para ayudar a los funcionarios de las aduanas a determinar el nivel adecuado de intervención para cada transacción comercial, basándose en el tipo de mercancías a inspeccionar (Cotecna, 2010). Otra solución, el Sistema Inteligente de Gestión de Vulnerabilidades Informáticas (SIGVI. Ecuador 2008-2009), facilita la gestión de alertas de vulnerabilidades para que las organizaciones actúen con tiempo suficiente para poder corregirlas antes de que el riesgo potencial de su ocurrencia se convierta en una amenaza real o en un desastre (Funiber, 2010).

Por su parte, el Sistema Inteligente de Selección de Información (SISI) es un sistema cubano que está destinado para estudiantes de medicina, enfermería y estomatología, residentes y médicos. Permite la realización de consultas y diagnósticos médicos con la combinación de las técnicas proporcionadas por los sistemas basados en casos y las redes neuronales. Una ventaja muy importante de la misma tiene que ver con su utilidad para la consulta de información (Fernández, 2007), pero no está concebido para la identificación de riesgos de ningún tipo.

### **Arquitectura de los sistemas basados en casos**

Un sistema basado en casos tiene tres componentes principales: una interfaz de usuario, un motor de inferencia y una base de casos. La base de casos contiene las descripciones de los problemas resueltos previamente en forma de rasgos (predictores y objetivos). Cada caso puede describir un episodio particular o una generalización de un conjunto de episodios relacionados. El motor de inferencia es la máquina de razonamiento del sistema, la cual compara el problema insertado con los que están almacenados en la base de casos y como resultado infiere una respuesta con el mayor grado de semejanza a la que se busca. La interfaz de usuario permite la comunicación entre el sistema y el usuario, dando la posibilidad de interactuar con la base de casos, plantear nuevos problemas y consultar los resultados inferidos.



En el estilo de solución de problemas con el uso de esta técnica se recupera un caso semejante al nuevo y la solución del problema recuperado se propone como solución potencial del nuevo problema. Esto se deriva de un proceso de adaptación en el cual se adecua la vieja solución a la nueva situación (Gálvez, 1998). Estos sistemas definen una serie de pasos y componentes que interactúan en un ciclo de razonamiento. A partir de un nuevo problema son recuperados los casos similares al introducido, que posteriormente pasan por un proceso de adaptación lográndose una respuesta acorde a la situación planteada. Luego, de ser necesario y posterior a su revisión, el sistema decide si aprender o no la solución dada. Lo anteriormente expuesto es considerado el ciclo de razonamiento basado en casos como se muestra en la Figura 2:

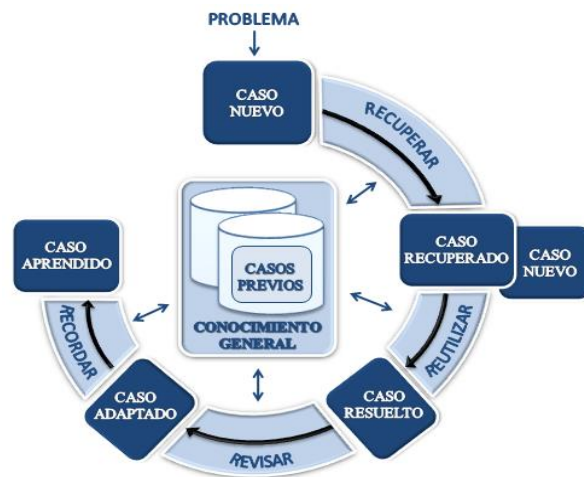


Figura 2. Ciclo de razonamiento basado en casos.

### Descripción de la solución propuesta

El Sistema Inteligente de Mitigación de Riesgos se basa en la información que introduce el usuario referente a la descripción de un proyecto. Utiliza técnicas de Inteligencia Artificial, específicamente de un sistema de RBC para distinguir por rasgos la descripción dada. Luego, los compara con los almacenados en la base de casos con que cuenta y devuelve un grupo de casos semejantes. Adapta la solución y retorna una respuesta obteniéndose las posibles acciones a realizar o estrategias a tomar, permitiendo la retroalimentación y aprendizaje del sistema.

La solución desarrollada combina principalmente un sistema de gestión de información con las técnicas de RBC. Constituye una aplicación web, basada en modelo Cliente-Servidor desarrollada sobre la plataforma Java con el uso

del framework Grails<sup>1</sup>. Posee un diseño sencillo con una interfaz amigable, predominan los colores azul claro y gris. Incluye una ayuda para facilitar el uso y comprensión del sistema por parte de los usuarios. Permite el almacenamiento, inserción, actualización y eliminación de toda la información referente a los proyectos, los usuarios, además de los riesgos y su posible mitigación. La combinación con un sistema de RBC permite que, a partir de la información insertada, y las características de los proyectos, los riesgos y las clasificaciones de éstos con sus respectivas mitigaciones, se conviertan en el conocimiento almacenado para su utilización en el proceso mostrado en la figura 2 para la identificación de riesgos. Además permite mostrar estos resultados de forma organizada y comprensible para el usuario facilitando su posterior análisis.

Como mecanismo de seguridad y control de acceso, el sistema cuenta con formulario de autenticación, y niveles de acceso definidos por roles de usuarios en tres niveles: administrador, consultor y usuario. Este mecanismo se basa en el framework de seguridad Grails Spring Security Core 1.0.1. Permite definir la seguridad por usuarios y roles haciendo uso de las anotaciones a nivel de funcionalidad. Cada rol tendrá el acceso necesario para ejecutar determinadas acciones sobre el sistema y así manejar la información que se requiera según las características de cada uno. Para ello, el sistema permite la gestión de los usuarios, además de facilitar la conexión al servidor LDAP (Protocolo de Acceso Ligerero a Directorios) existente en la Universidad de las Ciencias Informáticas, evitando así que se almacene y dupliquen datos innecesarios.

### **Base de casos definida y el proceso de identificación de riesgos**

Un caso es una pieza contextualizada de conocimiento que representa una experiencia (Perez, 2002) fundamental para alcanzar los objetivos del razonador. Estos están compuestos por un conjunto de rasgos predictores, que definen las características distintivas de los proyectos de desarrollo de software que permiten identificar la ocurrencia de riesgos; así como un conjunto de rasgos objetivos definidos por los posibles riesgos a incidir.

La base de casos propuesta cuenta con 23 rasgos predictores, definidos según los elementos fundamentales del modelo de gestión de riesgos MOGERI (Zulueta, 2009). Estos constituyen características de los proyectos que permiten predecir, a partir de las experiencias acumuladas, la ocurrencia de situaciones tanto adversas como favorables para un determinado proyecto de desarrollo de software. Además, se adaptaron los elementos obtenidos a

---

<sup>1</sup> Marco de Trabajo de desarrollo de aplicaciones web implementado sobre la plataforma java

partir del estudio de otras metodologías y modelos de gestión de riesgos, formando un híbrido que se correspondiera de manera correcta con las condiciones existentes en el centro ISEC. Los rasgos que componen los casos almacenados en la base de casos, están formados por: el nombre, el valor, la relevancia (peso) y la incertidumbre o grado de certeza.

La siguiente Tabla muestra los rasgos definidos en la base de casos exponiendo también los valores de dominio y el tipo de las variables correspondientes. El rasgo objetivo corresponderá a los posibles riesgos a incidir, así como una posible mitigación asociada al mismo que le fue dada en el pasado.

Tabla 1. Conjunto de rasgos definidos, valores de dominio y tipo de variables respectivos.

<b>Nombre del Rasgo</b>	<b>Valor de Dominio</b>	<b>Tipo</b>
Dominio del negocio	Muy Alta, Alta, Media, Baja,	Ordinal
Experiencia en la plataforma	Ninguna	
Nivel de conocimiento del lenguaje		
Nivel de conocimiento de las herramientas		
Nivel de conocimiento de los desarrolladores		
Nivel de Planificación		
Nivel de dificultad del lenguaje		
Experiencia de trabajo en equipo	Mucha, Media, Baja, Ninguna	
Motivación		
Experiencia en el desarrollo de aplicaciones similares		
Cohesión del equipo		
Estabilidad de los requisitos	Muy Estables, Estables, Poco estables, Inestables	
Complejidad de la aplicación	Alta, Media, Baja	
Relación con el cliente	Buena, Regular Mala	
Cronograma	Muy Variable, Variable, Poco variable	
Información que se maneja	Confidencial, Pública, Privada	Literal
Cantidad de especialistas a tiempo parcial	[1...n]	Numérico

Cantidad de especialistas a tiempo completo	[1...n]	
Cantidad de Computadoras	[1...n]	
Cantidad de funcionalidades simples	[1...n]	
Cantidad de funcionalidades medias	[1...n]	
Cantidad de funcionalidades complejas	[1...n]	

Como se aprecia en la Tabla 1, los rasgos de la base de casos están definidos sobre tres tipos de variables: ordinales, literales y numéricos. Las funciones de comparación que permiten maximizar la semejanza entre los rasgos se definen en base al tipo de variable que lo describe. El peso asociado a cada una fue obtenido a través del criterio de expertos, analizando para esto el nivel de importancia que cada uno le otorgaba a las características definidas.

Para las variables literales en el marco de este trabajo se utiliza la siguiente función:

$$\delta_j (X_j(O_1), X_j(O_2)) = \begin{cases} 1 & X_j(O_1) = X_j(O_2) \\ 0 & \text{e. o. c.} \end{cases} \quad (1)$$

donde  $\delta(X_j(O_1), X_j(O_2))$ : define la función de comparación entre los casos  $O_1$  y  $O_2$ ,

$X_j(O_1)$ : valor del rasgo  $X_j$  del caso  $O_1$ ,

$X_j(O_2)$ : valor del rasgo  $X_j$  del caso  $O_2$ ,

e.o.c: en otro caso.

Teniendo como base el principio de obtener la semejanza entre un proyecto existente en la base de casos y uno nuevo a analizar, los rasgos cuyo valor de dominio lo definen variables numéricas son comparados utilizando una variación de la función Manhattan ajustada. Esta permite transformar el cálculo de la distancia entre dos valores en un intervalo definido, o bien en un conjunto del cual se conozcan el mínimo y el máximo valor que puedan tomar sus elementos a un resultado que se define como semejanza. Siguiendo la misma notación definida en (1) la función es la siguiente:

$$\delta_i (X_i(O_1), X_i(O_2)) = 1 - \frac{|X_i(O_1) - X_i(O_2)|}{r_{\text{máx}} + r_{\text{mín}}} \quad (2)$$

donde  $\delta(X_i(O_1), X_i(O_2))$ : define la función de comparación entre los casos  $O_1$  y  $O_2$ ,

$r_{\text{máx}}$  y  $r_{\text{mín}}$ : los valores de rango máximo y mínimo del conjunto de valores existentes respectivamente. Estarán dados por los volares mínimo y máximo del rasgo  $X_j$  en la Base de Casos y

$X_j(O_1)$ : valor del rasgo  $X_j$  del caso  $O_1$  y

$X_j(O_2)$ : valor del rasgo  $X_j$  del caso  $O_2$ .

Los rasgos que sus valores de dominio lo definen variables de tipo ordinal discreto, expresan medidas o cualidades que no pueden ser calculadas de manera objetiva. En estos casos no son esenciales los valores en sí, sino su orden relativo. Por este motivo, son normalizados y luego comparados utilizando (2).

Para normalizar los valores según (Rui, 2009) se prosigue de la siguiente manera:

Se define  $M_i$  como la cantidad de estados ordenados que puede tomar el rasgo  $X_j$ . Se hace corresponder a cada valor del dominio, según su orden un número entero entre 1 y  $M_i$ . Por ejemplo, para el caso de la Complejidad de la aplicación,  $M_i=3$ ; Baja=1, Media=2, Alta=3. Luego cada valor del dominio se hace corresponder con un valor real entre 0 y 1, siguiendo la siguiente función (Rui, 2009):

$$X_j(O) = \frac{v - 1}{M_i - 1} \quad (3)$$

donde  $v$ : representa el valor del número entero asignado al valor del rasgo  $X_j$ .

### **Tratamiento de la incertidumbre**

De forma empírica, la incertidumbre es una medida inseparable de casi cualquier media resultante de una combinación de inevitables errores de medición y límites de resolución de los instrumentos de medición utilizados. De forma cognitiva, esta emerge de la vaguedad y la ambigüedad inherente al lenguaje natural (Gutiérrez, 2002). Muchos de los sistemas de RBC pueden aplicarse solo a situaciones deterministas. Sin embargo, hay casos prácticos que implican incertidumbre. En un sistema de RBC para la gestión de riesgos de software hay posibilidades de imprecisión dada la naturaleza de los riesgos y los elementos que provocan su surgimiento e impacto. Los rasgos que se tienen en cuenta no afectan con un mismo grado de certeza la obtención de un resultado determinado, por lo que es útil extender la lógica clásica para incorporar incertidumbre.

Los rasgos que componen los casos almacenados en la base de casos, están formados por tres elementos: el nombre, el valor y la relevancia; pero es necesario también representar una información adicional que indique el grado de certeza del valor de un rasgo, lo que da lugar a la inserción de un nuevo elemento. Cada valor de incertidumbre representará un valor continuo perteneciente a un conjunto difuso definido en el intervalo de  $[0;1]$  donde el 0 representa el desconocimiento total que se tiene del rasgo mientras que 1 el conocimiento pleno del mismo. Los

valores ubicados dentro de este intervalo se traducen en el grado de incertidumbre que se tiene sobre el rasgo, por lo que, los valores que tienden a 0 representan un mayor grado de incertidumbre mientras los que tienden a 1 un mayor grado de certeza.

Las incertidumbres de los rasgos predictores pueden ser especificadas en caso de que el especialista la conozca. Previendo que no siempre son expertos los que realizarán esta labor, el sistema propone un valor de incertidumbre haciendo uso de la ingeniería del conocimiento, partiendo de las incertidumbres previas o registradas en la base de casos para cada uno de los rasgos y utilizando un enfoque probabilístico. El resultado describirá el comportamiento más probable que puedan tener las incertidumbres desconocidas para cada rasgo respectivamente, haciendo uso de la media de ocurrencia de ese elemento:

$$I(X_j) = \frac{\sum_{j=1}^n I(X_j(O_n))}{n} \quad (4)$$

donde  $I(X_j)$ : define la incertidumbre no especificada del rasgo  $X_j$ ,

$I(X_j(O_n))$ : la incertidumbre del rasgo  $j$  para el caso  $n$  y

$n$ : la cantidad de casos de la base de casos.

La identificación de riesgos es la funcionalidad de mayor relevancia en el sistema. A través de los datos del proyecto de un usuario, este último ejecuta la acción Identificar Riesgos. De esta forma, se comparan los datos entrados con los casos almacenados en la base de conocimientos a partir de las funciones de comparación correspondientes para luego evaluar el caso. Cada dato, según su valor de dominio tendrá una función de semejanza, la cual compara dicho valor con su rasgo correspondiente en cada caso de la base de conocimientos. Al terminar de analizar y comparar cada rasgo predictor (columna de la Base de Casos o campo independiente), se procede a evaluarlo (rasgo objetivo o fila de la Base de casos). La función de evaluación es la siguiente:

$$\beta(O_1, O_2) = \frac{\sum_{i=1}^n \omega \cdot \delta(X_j(O_1), X_j(O_2)) \cdot [1 - (I(X_j(O_1)) - I(X_j(O_2)))]}{\sum_{i=1}^n \omega} \quad (5)$$

donde  $\beta(O_1, O_2)$ : define la semejanza entre los casos  $O_1$  y  $O_2$

$\omega$ : define el peso de los rasgos,

$\delta(X_j(O_1), X_j(O_2))$ : define la función de comparación entre los casos  $O_1$  y  $O_2$  y

$I(X_j(O_1), I(X_j(O_2)))$  : las incertidumbres respectivas del rasgo  $X_j$ .

Un principio importante del RBC es recuperar solo aquellos casos cuya semejanza tenga un valor significativo para darle solución al nuevo problema. Por esta razón, todo caso para ser considerado como posible solución requiere que el resultado de su comparación con el nuevo esté por encima de un valor umbral que define a partir de qué valor una solución es considerada significativa. Este valor umbral es calculado a partir de la construcción de un matriz de semejanza MS entre los casos y a partir de esta se aplica la siguiente función (Martínez, 2009):

$$\beta_{0} = \frac{2}{m(m-1)} \sum_{i=1}^{m-1} \sum_{j=j+1}^m \beta(o_1, o_2) \quad (6)$$

donde  $\beta_0$ : define el umbral de semejanza,

$\beta(O_1, O_2)$ : la semejanza entre los casos  $O_1$  y  $O_2$ ,

m: número de casos,

i y j recorren las filas y columnas respectivamente.

Después de realizar la búsqueda y selección de las soluciones candidatas resultantes de la recuperación (como primer elemento de entrada al ciclo de razonamiento), se procede a la adaptación de la solución, paso siguiente del ciclo de razonamiento basado en casos mostrado en la figura 2. En el marco de este trabajo se hará uso de una adaptación por reinstanciación basada en el criterio de la utilidad esperada del caso.

### Utilidad esperada del caso

La utilidad esperada del caso, expresa un criterio entre la semejanza del caso devuelta por la función de evaluación y la incertidumbre del caso similar (Gutiérrez, 2002), definida por la siguiente función:

$$\mu(O_u) = \alpha \beta(O_n, O_r) + (1 - \alpha) * \theta(O_r) \quad (7)$$

donde  $\mu(O_u)$ : define la utilidad esperada del caso,

$\beta(O_n, O_r)$ : la semejanza el caso nuevo  $O_n$  y el caso recuperado  $O_r$ ,

$\theta(O_r)$ : el valor de incertidumbre del caso recuperado. Es obtenido mediante el cálculo de la media aritmética de los valores de las incertidumbres de los rasgos que componen al caso  $O_r$ .

$\alpha$  se considera un coeficiente destinado a establecer la relación de la utilidad con respecto a la semejanza o la incertidumbre.

El valor de  $\alpha$  es un parámetro que a medida que tienda a 1 se le dará más importancia a la semejanza que a la certeza de la solución y viceversa. Es común que este valor sea dado por un experto que sea capaz de identificar para su



problema qué es lo que se necesita: un valor más cierto o bien uno más semejante. Si el experto sabe decidir qué es lo desea obtener, podrá especificar el valor de  $\alpha$  con un número ubicado en el intervalo [0; 1].

En caso de no ser un experto o que se desconozca el posible valor que pueda tomar  $\alpha$ , se hará la propuesta de un valor a partir de la siguiente función:

$$\alpha = |\beta(O_1, O_2) - \vartheta(O_r)| \quad (8)$$

Se mantiene la notación definida en (7).

Luego de obtenido el valor de la utilidad el caso seleccionado para realizar la adaptación y con ello dar respuesta al problema planteado en la descripción del nuevo caso, será el que mayor utilidad presente.

De esta forma, cada caso nuevo es comparado con los existentes teniendo en cuenta la incertidumbre. Este proceso se realiza de forma iterativa para todos los rasgos de cada caso obteniendo así el conjunto de casos que expresen mayor semejanza a la descripción entrada. Luego, este grupo de casos es adaptado para finalmente mostrarle al usuario en una página algunos datos del proyecto y los posibles riesgos a incidir con la una posible mitigación.

La necesidad de automatizar el proceso de identificación de riesgos, planteó la tentativa de utilizar la aplicación para la obtención de resultados inmediatos en el centro. De ocho grandes proyectos existentes en el mismo, la aplicación fue utilizada en cinco, de los cuales en cuatro ofreció buenos resultados a corto y mediano plazo, mientras que en solo uno ofreció resultados a largo plazo, dependiendo del análisis de los especialistas del proyecto. Lo anterior facilitó aminorar el tiempo en las actividades referentes a la gestión de riesgos y de esta forma disminuir la utilización de un grupo de recursos, facilitando la culminación de diferentes fases de los proyectos según el alcance definido en el tiempo requerido.

## Conclusiones

La solución desarrollada constituye una herramienta capaz de ayudar a la toma de decisiones, pues a partir de ciertos datos de entrada correspondientes a características de los proyectos se pueden predecir posibles riesgos a incidir así como algún tipo de tratamiento que se le puede dar a estos. Para ello se definieron los procesos fundamentales relacionados con la gestión de riesgos de software y tras el análisis detallado de los métodos que abordan el tema, se seleccionó MOGERI como la metodología adecuada de acuerdo con las características del centro de desarrollo de software ISEC.

La formalización de un modelo computacional donde se aplican técnicas de inteligencia artificial, a consideración de la mitigación de riesgos como entidad principal dentro de los procesos de desarrollo de software, permitió poner en práctica esta solución en proyectos reales del centro. Fueron enunciados los algoritmos propuestos para considerar la semejanza entre cada par de proyectos. Además se describen las características generales de una herramienta automatizada que implementa el RBC de forma que se pueda acumular la experiencia en la detección de riesgos en los proyectos de software y emplearla en la ejecución de nuevas revisiones. Del total de proyectos analizados, el 80% arrojó buenos resultados, mostrando un ahorro de recursos y tiempo y cumpliéndose con los objetivos propuestos en el alcance definido.

## Referencias

- CIBERTEC. Gestión de Proyectos de Software bajo el enfoque PMI. 2011. [Consultado el: 20 de agosto de 2011]. Disponible en: [[http://www.cibertec.edu.pe/2/modulos/JER/JER\\_Interna.aspx?ARE=2&PFL=2&JER=3338](http://www.cibertec.edu.pe/2/modulos/JER/JER_Interna.aspx?ARE=2&PFL=2&JER=3338)].
- CONNECTION, ACQUISITION COMMUNITY. 2010. [Consultado el: 23 de julio 2010]. Disponible en: [<https://acc.dau.mil/CommunityBrowser.aspx?id=19156>].
- COTECNA. 2010. [Consultado el: 30 de agosto de 2010]. Disponible en: [[http://www.cotecna.ch/COM/ES/risk\\_management.aspx](http://www.cotecna.ch/COM/ES/risk_management.aspx)].
- EXPERT, SOFT. [Consultado el: 10 de diciembre de 2010]. Disponible en: [<http://www.softexpert.es/norma-pmbok.php>].
- EXPÓSITO GALLARDO MC.; ÁVILA ÁVILA R. Aplicaciones de la Inteligencia Artificial en la medicina: Perspectivas y problemas. ACIMED, 17(5): p. 2-9, 2008.
- FUNIBER. Sistema Inteligente de Gestión de Vulnerabilidades Informáticas (SIGVI) - (Ecuador 2008-2009). [Consultado el: 9 de mayo de 2010]. Disponible en: [<http://www.funiber.org/proyectos/idi/sistema-inteligente-de-gestion-de-vulnerabilidades-informaticas-sigvi-ecuador-2008-2009>].
- GÁLVEZ, DANIEL. Curso de Sistemas Basados en el Conocimiento. Villa Clara, Cuba. Universidad Central “Martha Abreu” de Las Villas, T83-T825, 1998.
- GARCÍA, J. L. Expertos TIC. Red Social de profesionales y empresas. Gestión de Proyectos. [En línea] 2009. [Consultado el: 25 de agosto de 2011]. Disponible en: [<http://www.expertostic.com/articulos/gestion-proyectos.html>].

- GUTIÉRREZ, I; BELLO, R y TELLERÍA, A. Un Sistema Basado en Casos para la toma de decisiones en condiciones de incertidumbre. Revista de Investigación Operacional, 23(2). p. 103-121, 2002.
- MARTÍNEZ SÁNCHEZ, N.; GARCÍA LORENZO MM. y GARCÍA VALDIVIA ZZ. Modelo para diseñar sistemas de enseñanza-aprendizaje inteligentes utilizando el razonamiento basado en casos. Revista Avances en Sistemas e Informática, 6(3): p. 72, 2009.
- PÉREZ-SOLTERO, A. Memoria Organizacional Basada en Casos. Revista de Ciencia e Tecnología Política e Gestao para a Periferia (RECITEC), 6(1), p. 22-39, 2002.
- PRESSMAN, R. Ingeniería del Software. Un enfoque Práctico 7ma Edición., McGraw-Hill Company, New York EE.UU, Cap. 28, p. 744-752, 2010.
- PROJECT MANAGEMENT INSTITUTE. [www.pmi.org](http://www.pmi.org). [Consultado el: 20 de agosto de 2011]. Disponible en: [<http://www.pmi.org/About-Us.aspx>].
- RISKTRAK INTERNACIONAL. [Consultado el: 15 de agosto de 2010]. Disponible en: [<http://risktrak.com>].
- RUI X.; Wunsch II DC. Clustering. IEEE, p. 27-29, 2009.
- TRIMS Software LLC. Products. [Consultado el: 10 de diciembre de 2010]. Disponible en: [<http://www.trims.com/aboutus.htm>].
- ZULUETA, Y; DESPAIGNE, E. La gestión de riesgos en la producción de software y la formación de profesionales de la informática: experiencias de una universidad cubana. Revista Española de Innovación, Calidad e Ingeniería de Software, 5 (3): p. 6-20, 2009.