

## Gestión de riesgos en proyectos de software educativos

Fecha de recepción: 25 de mayo de 2018 • Fecha de aceptación: 20 de junio de 2018 • Fecha de publicación: 6 de septiembre de 2018

**Ing. Maryoribel Reañez**

Unidad Educativa Nuestra Señora del Rosario. Ecuador  
*maryoribel20@gmail.com*

**Mg. Byron Vallejo**

Instituto Tecnológico Superior Francisco de Orellana. Ecuador  
*ronald.vallejo@itsfo.edu.ec*

**PhD. Mercedes Delgado**

Universidad del Zulia. Venezuela  
*merdelgon@hotmail.com*

### Resumen

El propósito de esta investigación fue analizar la gestión de riesgos en proyectos de software educativos. La investigación se ha fundamentado en autores como Fajardo (2014), Sánchez (2013), Zulue-ta et al. (2009), entre otros; y se encuadra en el paradigma cualitativo, con un enfoque postpositivista, siendo este estudio de tipo documental, analítico. Asimismo, se ha realizado a través de la técnica de la observación directa sobre la literatura referencial recopilada, y usando el análisis de contenido para la interpretación de los referentes conceptuales desarrollados. De esa revisión documental concreta, se evidenciaron fallas en el diseño de proyectos de software educativos, ya que no se atiende al sentido del riesgo, a la incertidumbre e inseguridad en su creación, a su arquitectura, a los procedimientos técnicos, a los factores externos que impiden su funcionalidad; por lo cual se recomienda producir una concienciación sobre su confección, diseño y operatividad.

Palabras clave: gestión de riesgos, proyectos, softwares educativos, diseño, planificación, control.

## Abstract

The purpose of this research was to analyze risk management in educational software projects. This research has been based on authors such as Fajardo (2014), Sánchez (2013), Zulueta et al. (2009), among others; and it fits into the qualitative paradigm, with a post-positivist approach, this study being documentary, analytical. Likewise, it has been carried out through the technique of direct observation on the collected reference literature, and using content analysis for the interpretation of the conceptual referents developed. From this concrete documentary review, flaws in the design of educational software projects were evident, since the sense of risk, uncertainty and insecurity in its creation, its architecture, technical procedures, external factors that impede its functionality are not addressed; therefore, it is recommended to produce an awareness of its preparation, design and operation.

Keywords: Risk management, projects, educational software, design, planning, control.

# Introducción

En un entorno organizacional, la gestión de proyectos busca implementar la modernidad en sus acciones, lo novedoso de los procedimientos a ejecutar, la creatividad en el diseño de estrategias. Pero todo proyecto implica riesgos específicos que deben ser canalizados de manera que se impidan peligros, inseguridades o cualquier alarma que reprima o imposibilite los planes y programas a ser desarrollados.

De allí que, habiéndose evidenciado en la praxis organizacional educativa mundial problemas de riesgos en la planeación de los proyectos de software, en los análisis de requerimientos, en el diseño de los mismos, en su desarrollo, en la puesta en práctica de las pruebas para examinar su funcionalidad y en su implementación, se hace necesario la búsqueda de fórmulas de eliminación o minimización de tales riesgos. Este estudio se genere precisamente con la finalidad de analizar la gestión de riesgos en proyectos de software educativos, buscando asumir las tareas dirigidas a la creación efectiva de tales programas educacionales.

## Gestión de riesgos

La gestión de riesgos se define como el conjunto de actividades que se realizan para reducir las incertidumbres asociadas con ciertas tareas, o eventos. En el contexto de los proyectos, esa misma gestión del riesgo reduce los impactos en ellos de los eventos no deseados, por lo cual requiere la realización de actividades del proceso de toma de decisiones.

Así lo expresa Fajardo (2014), quien a la vez considera que según la fuente SEI Administrative Agent (2009) acerca de la metodología para la gestión de riesgos en proyectos de software Spanish Technical Report CMMI V 1.3, la gestión de riesgos se puede dividir en las siguientes partes: en primer término, definir una estrategia de gestión de riesgos; en segundo término, identificar y analizar los mismos, y en tercer término, tratar los riesgos identificados, incluyendo la implementación de planes de mitigación, según sea necesario.

Asimismo, las metas y prácticas especificadas por la metodología CMMI que se deben cumplir son: Tabla 1: Área de proceso Gestión de Riesgos, Meta 1: Preparar la gestión de riesgos, debiendo ejecutar las siguientes prácticas: Práctica 1.1. Determinar las fuentes y las categorías de riesgos, Práctica 1.2. Definir los parámetros de riesgos, Práctica 1.3. Establecer una estrategia de gestión de riesgos.

De igual modo, en cuanto a la Meta 2 se deben Identificar y analizar los riesgos, debiendo ejecutarse las siguientes prácticas: Práctica 2.1. Identificar los riesgos, Práctica 2.2. Evaluar, clasificar y priorizar los riesgos. Por su parte, la Meta 3: Mitigar los riesgos, siendo la Práctica 3.1. Desarrollar los planes de mitigación de riesgos y la Práctica 3.2. Implementar los planes de mitigación de riesgos.

En este sentido, se indica que los avances en las técnicas de gestión del riesgo han permitido a algunas organizaciones de software implementarlas y fortalecer su gestión de riesgo; así, técnicas

de modelado y métodos de pensamiento como lluvia de ideas son las más usadas para dicha gestión. Estas técnicas se utilizan para comprender el riesgo e incluyen técnicas cualitativas, semicuantitativas y cuantitativas; además, estos métodos se aplican con frecuencia en el lugar de trabajo por sí mismos o en conjunto con otras técnicas.

Se destaca que el área de proceso Gestión de Riesgos describe una evolución de estas prácticas específicas para planificar, prevenir y mitigar los riesgos sistemáticamente a fin de minimizar proactivamente su impacto sobre el proyecto. Aunque el énfasis principal del área de proceso Gestión de Riesgos se realiza sobre el proyecto, estos conceptos también pueden aplicarse para gestionar los riesgos de la organización.

De la misma manera, se considera que los riesgos pueden ser agrupados en tres (3) categorías:

Riesgos del Negocio, que comprenden todo lo que puede afectar el cumplimiento de los objetivos del proyecto, los cuales deben ser gestionados por la empresa y no se pueden transferir, tales como riesgos de mercado, de estrategia, de ventas, de gestión y/o de presupuesto.

Riesgos Técnicos, los cuales se pueden identificar de acuerdo al ciclo de vida de desarrollo de software (SDLC) y deben ser gestionados por las personas responsables de cada etapa del SDLC.

Riesgos Externos, que corresponden a todos los riesgos que no se pueden controlar desde el proyecto, como lo es la legislación a la cual deban sujetarse los mismos.

Esa misma gestión de riesgos se define en forma estructurada, operacional y organizacional, según Maniasi, Britos y Diez (2006), como una serie de actividades para darle tratamiento a los riesgos de los proyectos a lo largo de todas las fases de su ciclo de vida de desarrollo de software. En la mayor parte de los casos, esto se traduce en la creación de planes tendientes a impedir que los riesgos se transformen en problemas o a minimizar su probabilidad de ocurrencia o impacto.

### **Gestión de riesgos en software educativos**

Los proyectos de software al igual que cualquier otro tipo de proyectos tienen riesgos que pueden ser materializados durante su ejecución, según lo expresa Fajardo (2014), quien señala que el desarrollo de software es una actividad muy compleja e impredecible. En este sentido hace referencia a Standish Group International (2013), siendo conocida por sus informes sobre proyectos de implementación de sistemas de información en el sector público y privado, especialmente centrada en los fallos y posibles mejoras en los proyectos de TI.

Dicha organización, firma independiente de investigación de tecnologías de información (TI) internacional, indicó para ese año 2013 que el 43% de los proyectos de software no se entregan a tiempo, dentro del presupuesto, y con las funciones requeridas, por efectos de algunos riesgos evidenciados; mientras que el 18% de los proyectos de software fueron cancelados.

Sobre el particular, expresa la mencionada autora, Fajardo (2014), que las empresas invierten recursos sustanciales y esfuerzo en el desarrollo de software; en consecuencia, el control de los

riesgos asociado a los proyectos es vital para garantizar resultados exitosos. Vale decir que la comprensión de la naturaleza de los distintos riesgos y su relación con el rendimiento del proyecto es cada vez más importante.

Señala igualmente Islam (2009) que tales proyectos de desarrollo de software a menudo se enfrentan a problemas imprevistos que plantean riesgos potenciales dentro del entorno. El control de estos riesgos surge tanto de los componentes técnicos como de los no técnicos del desarrollo, y tener control de ellos desde las primeras etapas es crucial para lograr un proyecto exitoso. Por lo tanto, la gestión del riesgo de desarrollo de software está siendo reconocida como la mejor práctica en la industria del software para reducir tales peligros e incertidumbres antes de que ocurran.

Ahora bien, sobre la gestión de riesgos de software educativos ha de señalarse que Zulueta, Despaigne y Hernández (2009) sostienen que la misma debe integrar de forma sistémica los procesos que se encargan tanto de planificar, identificar y analizar como de responder al riesgo; y además de seguir, controlar y comunicar las actividades planificadas al respecto. Es así como, de manera ejemplar, destaca la Universidad de las Ciencias Informáticas (UCI) de Cuba, donde se propuso alcanzar este propósito con la ejecución de ambiciosos programas curriculares y de producción, con la aplicación de las más modernas tecnologías en la docencia.

En la UCI se renueva y afianza el modelo de formación-producción-investigación, donde el estudiante puede ejercitar y comprobar, en condiciones de trabajo reales, las habilidades adquiridas de forma curricular, logrando así completar el ciclo de enseñanza-aprendizaje y aplicar las mejores prácticas de producción de software basadas en modelos y estándares internacionales.

Sin embargo, señalan los mencionados teóricos que educar a los estudiantes en la gestión de riesgos no es fácil, porque asimilar las habilidades necesarias generalmente toma varios años. Con base en la experiencia, analizando que la aparición de los riesgos no está limitada por los tipos de actividades del ser humano y tampoco sujeta a delimitaciones temporales, es ineludible la necesidad de la formación temprana en el tratamiento de los riesgos.

Asimismo, se indica que la práctica profesional propicia las condiciones pedagógicas para iniciar este proceso de enseñanza aprendizaje. En ella los estudiantes deben planificar el trabajo para la realización de proyectos de curso sencillos, lo cual debe completarse con la identificación de los riesgos que pueden afectar su desarrollo exitoso. Al concluir, resulta provechoso valorar los riesgos que resultaron problema, y analizar la efectividad de la identificación inicial.

Este modelo de gestión de riesgos comenzó su implantación en el año 2006, lo cual permitió develar sus bondades, debilidades y las acciones necesarias para generalizarlo en el proceso productivo UCI. El primer ciclo de aplicación incluyó proyectos con diferentes características y en diferentes fases del ciclo de vida: Proyecto “Atención Primaria de Salud”, encargado de la informatización de la gestión de la información relacionada con este proceso (APS); Proyectos del Programa Nacional de Informatización del Conocimiento Geológico en Cuba (PNICG); Proyecto “A Jugar”: Software Educativo para la enseñanza preescolar (AJ); Proyecto “Sistema de información geográfica de la Universidad de las Ciencias Informáticas (SIG-UCI)”, entre otros.



En igual sentido de lo expresado, Sánchez (2013) señala que en la gestión de riesgos existen algunos problemas, como identificarlos, diseñar planes para compensar el riesgo, prevenirlos proporcionalmente, vale decir, evitar al gestor paranoico, obsesivo en las tareas, así como monitorizar su estado. En tal sentido, el objetivo será identificar, analizar y cuantificar posibles riesgos que puedan aparecer durante el desarrollo de un proyecto software, desarrollar respuestas adecuadas para los posibles riesgos y monitorizar el transcurso de un proyecto para evaluar el estado de los riesgos y actuar en consecuencia.

De allí que proponga un esquema general de la gestión de riesgo, centrado en establecer un Plan General de Riesgos, identificar los mismos, hacer un análisis cualitativo, un análisis cuantitativo y un plan de respuesta a riesgos, así como el control, haciendo énfasis en las técnicas de identificación de riesgos, las cuales deben centrarse en la revisión de la documentación existente, la revisión de planificación y estimaciones, la tormenta de ideas, el juicio experto sobre el cual enfatiza la Técnica Delphi, además de realizar la Taxonomía de riesgos, el Análisis SWOT y el Diagrama de Ishikawa.

Todo lo planteado, en consideración del investigador, conlleva en principio la posibilidad de diseñar softwares educativos con el máximo de eficacia, en el entendido de que su confección, implantación y operatividad, estaría basada en estrategias y herramientas precisas, de planificación, de organización de tareas, de medidas de prevención, de control de fallas, que minimicen, reduzcan o eliminen los riesgos posibles en su generación.

### **Riesgos posibles y la necesidad de su resolución**

El software educativo es una de las grandes herramientas que proporciona beneficios tanto para los estudiantes como para los docentes. Entre las ventajas se puede mencionar en primer lugar, como señalan Ibarra y Quiroz (2013), el cambio del rol tradicional del profesor, siendo que ya este no solo es fuente de conocimientos, sino un mentor o animador del aprendizaje.

Asimismo, tenemos que ayudan a los estudiantes a trabajar en diferentes niveles y contenidos según su grado de desarrollo y sus necesidades. Por otra parte, abren nuevas posibilidades para la enseñanza diferenciada, por lo que permiten atender mejor el aprendizaje y desarrollar las potencialidades individuales de cada uno de los alumnos; a la par, ofrecen nuevas posibilidades para evaluar el aprendizaje de los alumnos.

Por esto el desarrollo de un proyecto de software educativo permite mejorar los procesos pedagógicos utilizados por los docentes de una institución educativa, y a los estudiantes les permitirá cambiar la manera de aprender de una forma más interactiva y dinámica, lo cual inducirá a los alumnos a desarrollar mayores competencias en el área que corresponda.

No obstante, no siempre es posible minimizar los problemas o riesgos que derivan de la creación y diseño de un proyecto de software educativo, de su funcionalidad; por lo cual, en la actualidad, algunas organizaciones han intentado definir procedimientos para la gestión de riesgos, los cuales se han complementado a lo largo del proceso de certificación que las compañías han llevado.

Un caso preciso es el de la empresa Informática & Tecnología Stefanini, la cual, a través de su procedimiento PC-PP-03 Gestión de Riesgos y Problemas, clasifica los riesgos y algunas posibles soluciones a los mismos, los cuales, basados en los planteamientos de Fajardo (2015), se desagregan como un mecanismo ejemplarizante para cualquier organización, en este caso educativa.

En materia de planeación, se habla del riesgo de un posible manejo ineficiente de recursos, ante lo cual se exige que la obtención de los recursos siempre debe ser acorde al desarrollo del proyecto. Por otro lado, si ha habido estimaciones imprecisas acerca de costos y materiales, estas deben ser en gran medida a juicio de expertos, pero en algunos casos la realidad de los proyectos ha demostrado que las desviaciones entre lo estimado y lo ejecutado son considerables. Si se considera que la planeación es inadecuada, se debe prever que los gerentes de proyecto establezcan cada uno de los planes requeridos: plan de comunicación, plan de gestión de la configuración, entre otros.

Por otra parte, si el personal no está preparado adecuadamente para las actividades de algunas de las fases concebidas, se debe hacer depender de la disponibilidad de los recursos; en algunos casos los recursos no son propios de un único proyecto. También cuando hay mala definición de entregables, estos deben ser definidos en conjunto con el usuario al inicio del proyecto.

De la misma forma, si la gestión contractual no es apropiada, debe atenderse a que la misma recaee en personas distintas a los ejecutores del proyecto. También pueden existir deficiencias en la definición e involucramiento de los interesados del proyecto, lo cual resulta común y el riesgo principalmente se presenta cuando el cliente no asigna a las personas indicadas. Si existen discrepancias entre los líderes del proyecto, por el desconocimiento de metodologías y mejores prácticas en la ejecución de los proyectos, se debe seguir la metodología del cliente que ha implicado un mayor impacto en tiempo y costo.

Por otro lado, en algunos casos se observa el uso de herramientas inadecuadas, porque se corre el riesgo de que estas no sean reconocidas en el mercado. A veces el personal no está preparado adecuadamente para las actividades de alguna fase, por lo cual debe evaluarse su desempeño y resultados de acuerdo al rol que cumple. Asimismo, si el tamaño del equipo no es el adecuado, esto debe hacerse depender de la disponibilidad de recursos, y el usuario debe contar con un alto grado de rotación de personal. Si hay falta de conocimiento de procesos de uso, debe atenderse a que tales usuarios pueden ser personas que no llevan mucho tiempo en la entidad, educativa en este caso, o no poseen un perfil que tiene claro lo que necesita el interesado.

De igual modo, si se corre el riesgo de que los usuarios que participan no sean los indicados, desde el inicio del proyecto se debe explicar la importancia de que los usuarios que participen en el proyecto sean los indicados. Si existe falta de claridad en la definición de requerimientos por parte de estos, como parte del proceso se debe establecer que un líder del área debe ser la persona que realice la aprobación del documento de requerimientos con el fin de corroborar la información entregada por los usuarios.

También, si no hay especificación de requerimientos no funcionales, como parte de la plantilla del documento de requerimientos debe ser incluida la sección de Requerimientos No funcionales. Si la información suministrada por los usuarios no es real, como parte del proceso se debe establecer que un líder del área debe ser la persona que realice la aprobación del documento de requerimientos con el fin de corroborar la información entregada por los usuarios.

Por otra parte, en cuanto al Diseño de proyectos, en ocasiones se corre el riesgo de una arquitectura inapropiada, pero puede suceder que el usuario ya tenga establecida esta arquitectura; por tal razón una de las actividades iniciales es hacer una evaluación de la misma. Asimismo, se debe prever que las herramientas que actualmente se manejan sean reconocidas en el mercado. Si el diseño lógico está pobremente definido, debe ser sometido a revisión de pares y aprobación del interesado. Si es el caso de un diseño físico pobremente definido, debe ser sometido a revisión de pares y aprobación del solicitante. Tal es el caso del diseño de interfaces deficiente. Si hay falta de visión, es porque el líder del proyecto asignado no ha sido una persona con experiencia comprobada en su rol, ya sea en la misma entidad solicitante o en otras.

Asimismo, si existe el riesgo de un lenguaje de desarrollo inapropiado, deben utilizarse herramientas reconocidas en el mercado. Asimismo, si existen estándares de documentación inapropiados, deben ser cambiados y establecer lineamientos de codificación, evitando con ello cualquier error o deficiencia. Si hay falta de conocimiento del conjunto de herramientas a usar, se deben promover capacitaciones continuas para mitigar este riesgo. Si existe dependencia de una herramienta en particular, se debe contar con diversas líneas de negocio que garanticen el conocimiento y experticia en distintas herramientas.

Por otra parte, se debe prever el manejo de liberación de versiones inadecuado, evitando errores en el decurso de su uso o mala implantación; sobre esto, actualmente se está implementando el uso de nuevas herramientas para el control de versiones que mejoren esta gestión. Si la Gestión de configuración es inadecuada, como parte de la metodología debe estar definido el proceso de gestión de la configuración y al inicio de cada proyecto hacer acuerdos con los usuarios e incluirlos en el plan de proyecto. Si existen malas especificaciones de construcción, se deben programar inspecciones de código para que el líder o un par del equipo impida incurrir en el riesgo de no validar el cumplimiento de los lineamientos definidos.

De igual manera, con relación a las Pruebas, existe el riesgo de unos criterios de aceptación inadecuados, entonces como parte de la metodología, deben crearse criterios de aceptación definidos en conjunto con los usuarios. Si hay riesgo de herramientas de pruebas inadecuadas, estas deben buscarse de licenciamiento libre. Si la Metodología de pruebas es inadecuada, debe establecerse un plan de pruebas donde se defina todo el proceso y los tipos, el registro de incidentes, entre otras. Si no se involucra al usuario en las pruebas de aceptación, deben ser realizadas por éste, y el equipo del proyecto debe realizar una labor de acompañamiento.

Por otro lado, cuando el riesgo es la falta de soporte interno, los ambientes deben ser compartidos por varios proyectos; aunque esto a la vez genera inconvenientes. Pueden no realizarse pruebas de stress, debiendo ser conscientes de realizarlas. En ocasiones también hay peligro de dependencia de analistas de pruebas por su experiencia, por lo cual debe buscarse que el



equipo de trabajo sea el apropiado para el proyecto, considerando que cada proyecto es único y la dependencia de un único recurso no es lo más conveniente.

Asimismo, en cuanto a la Implementación, se corre el riesgo de problemas de comunicación, por lo cual deben hacerse seguimientos continuos para garantizar una interacción constante entre los involucrados en el proyecto. Si no hay una definición de la estrategia de implementación, se debe incluir un plan de gestión del cambio, que esté en proceso de definición para determinar su alcance. Si existe falta de capacitación a los usuarios, siempre se debe incluir un plan de capacitaciones: técnica y funcional. Si, por otro lado, existe falta de tiempo para la implementación, antes de iniciar la misma se reevalúa el tiempo establecido, y si no es suficiente se acuerdan con el usuario los ajustes necesarios.

Por otro lado, si hay riesgo de falta de manuales de usuario o técnicos deficientes, se deben realizar formatos con el contenido de los mismos y estos deben ser validados por pares del proyecto. Finalmente, si el soporte posterior a la implementación es pobre, en conjunto con el usuario se establecen acuerdos de niveles de servicio para estas actividades de soporte.

Con lo planteado el investigador evidencia que en el desarrollo de softwares educativos se requiere de una planeación importante con la finalidad de medir adecuadamente las necesidades existentes, identificando a la par las debilidades presentes, las carencias, así como los posibles riesgos, fallas, errores, peligros o situaciones adversas en general que pudieran afectar una correcta implementación, en desmedro del proceso de enseñanza-aprendizaje de los estudiantes.

## Reflexiones finales

Una vez realizada la revisión documental en este estudio, y compiladas una serie de referencias bibliográficas al efecto, derivada de textos escritos, digitales, de algunas referencias electrónicas precisas y pertinentes, se procesa esa información a través de la técnica de la observación documental, pudiendo hacer la interpretación de los textos y documentos digitales recolectados a través del análisis de contenido, para examinarlos, analizarlos, logrando dar respuesta al objetivo de investigación formulado y dilucidar la temática seleccionada a través de la lógica, el saber y entender del investigador.

En este sentido, se evidencia de lo recolectado, y de acuerdo a la lógica del investigador, en primer término, que en la realidad actual de aula, la intromisión del software educativo se ha convertido en una herramienta fundamental para los actores intervinientes, dadas sus características esenciales, las cuales convergen en la elaboración o diseño de materiales didácticos, elaborados con el soporte de la tecnología para simplificar los procesos y procedimientos educacionales, sobre todo en aquellos casos en los cuales resulta necesario simular procesos complejos, crear presentaciones animadas desde donde los sujetos puedan interactuar, pudiendo desarrollar habilidades cognitivas, comunicativas, tecnológicas, tan necesarias en el mundo de hoy.

De la misma manera, el software educativo permite el desarrollo de habilidades motoras a través de la ejercitación, a través de la facilitación de juegos, simuladores, semejando la realidad de manera entretenida, de acuerdo al área en la cual se haya concebido el proyecto, generando en tal



sentido mayor conocimiento en los usuarios, dada la versatilidad en su diseño.

Cabe destacar que lo ideal para las instituciones educativas es que el diseño curricular o las disposiciones normativas internas de cada institución en materia de pedagogía o didáctica, contemple la generación de softwares educativos en cada área del conocimiento, para evitar el uso indiscriminado de los mismos, propiedad de terceros, sin contar con la licencia respectiva, o los insumos presupuestarios y financieros necesarios para su adquisición, lo que les hace incurrir en el riesgo de perder la reputación institucional.

Ahora bien, el diseño de los softwares educativos comporta unos riesgos que deben ser previstos por los actores educativos, si lo que se busca es sacarles ventaja. Para ello se han diseñado unos estándares, los cuales forman parte del proceso de la gestión de riesgos, que pueden ayudar en el momento de determinar cómo prevenirlos o mitigarlos, cuando estos se determinan de manera específica. En general, tales riesgos pueden ser gestionados o mitigados proactivamente mediante la revisión de buenas prácticas, la priorización y las lecciones aprendidas considerando la prevención como una de las principales herramientas en nuevos proyectos y el control ante lo imprevisto.

En segundo término, de los hallazgos logrados se evidencia que, en materia de diseño de software educativos, se requiere atender a su planeación, diseño, desarrollo e implementación, si lo que se pretende es producir un resultado certero en el proceso de enseñanza-aprendizaje de los estudiantes, si lo que se busca es la efectividad del mismo de acuerdo al área en el cual se haya concebido el proyecto, pudiendo en tal sentido atender a la variedad de riesgos que fueron categorizados en párrafos anteriores.

En tercer término, es posible concluir que, una vez identificados los riesgos existentes, estos deben ser priorizados y se debe canalizar su resolución con base en estándares o guías y manuales existentes, o de acuerdo a lo que la misma praxis organizacional educativa requiera, pudiendo con ello minimizar el peligro de la indebida utilización de un software en aula.

De acuerdo a todo lo planteado, es posible sugerir que es importante crear una cultura de concientización del riesgo, debiendo las instituciones educativas planear adecuadamente y adaptarse, de acuerdo a sus propósitos y a las características propias de cada proyecto, a las carencias, debilidades y fallas de los estudiantes, con lo que pueden generar softwares educativos adecuados a las necesidades reales existentes.

Con la previsión de todos los procesos es posible impedir la obstaculización de los fines para los cuales se generan los proyectos, planes y programas educativos, debiendo adherirse a esa cultura del riesgo todos los actores intervinientes, en aras de propiciar que las metas para las cuales se crean los mismos sean cumplidas satisfactoriamente, en atención a los parámetros enunciados con anterioridad.

## Referencias bibliográficas

- Fajardo, R. A. (2014). *Metodologías para la gestión de riesgos en proyectos de software*. Universidad Militar Nueva Granada, Colombia. Recuperado de <http://repository.unimilitar.edu.co/bitstream/10654/12579/1/metodolog%c3%8das%20para%20la%20gesti%c3%93n%20de%20riesgos%20en%20proyectos%20de%20software.pdf>
- Ibarra, M. y Quiroz, P. (2013). *Proyecto de Software Educativo*. Universidad Laica “Eloy Alfaro” de Manabí, Facultad de Ingeniería En Sistemas. Manta, Ecuador. Recuperado de <https://es.slideshare.net/leonelibarra/software-educativo-26061634>
- Islam, S. (2009). Software development risk management model: a goal driven approach. Paper presented at the Proceedings of the doctoral symposium for ESEC/FSE on Doctoral symposium.
- Maniasi, S., Britos, M. I. P., & Diez, M. I. E. (2006). *Identificación de Riesgos de Proyectos de Software en Base a Taxonomías* (tesis de maestría). Escuela de Postgrado. ITBA, Buenos Aires, Argentina.
- Sánchez, P. (2013). *Ingeniería del Software II. Gestión de riesgos en proyectos software*. Recuperado de <http://ocw.unican.es/enseanzas-tecnicas/ingenieria-del-software-ii/materiales/tema7-gestionRiesgos.pdf>
- SEI Administrative Agent (2009). *Spanish Technical Report CMMI V 1.3*. Recuperado de <http://repository.unimilitar.edu.co/bitstream/10654/12579/1/metodolog%c3%8das%20para%20la%20gesti%c3%93n%20de%20riesgos%20en%20proyectos%20de%20software.pdf>
- Standish Group International, Inc. (2013). *Third Quarter Research Report*. Recuperado de <http://repository.unimilitar.edu.co/bitstream/10654/12579/1/metodolog%c3%8das%20para%20la%20gesti%c3%93n%20de%20riesgos%20en%20proyectos%20de%20software.pdf>
- Zulueta, Y., Despaigne, E. y Hernández, A. (2009). *La gestión de riesgos en la producción de software y la formación de profesionales de la informática: experiencias de una universidad cubana*. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software [en línea]. Recuperado de <http://www.redalyc.org/articulo.oa?id=92217181003>> ISSN