

## Gestión de riesgos en proyectos de desarrollo de software en España: estudio de la situación

### Risk management in software development projects in Spain: a state of art

*Luis Fernández Sanz\**, *Pedro Bernad Silva*

Departamento de Ciencias de la Computación, Universidad de Alcalá. CP. 28871. Alcalá de Henares, Madrid, España.

(Recibido el 31 de octubre de 2012. Aceptado el 28 de noviembre de 2013)

#### Resumen

El software es en nuestros días un elemento crítico en todos los sistemas. El desarrollo y la puesta en marcha de programas de ordenador están sujetos a numerosos riesgos que deben ser objeto de una gestión cuidadosa. La gestión de riesgos en proyectos de software es una actividad reglada en múltiples metodologías pero que en la práctica se aplica de forma muy desigual en los distintos equipos de trabajo y organizaciones. Un primer estudio llevado a cabo entre desarrolladores españoles revela un interés por la gestión de riesgos y sus técnicas, pero también serias carencias en su aplicación y algunas actitudes que no favorecen a esta disciplina. Estos datos preliminares constituyen una base para profundizar con más detalle en la práctica de gestión de riesgos en software y para plantear soluciones más eficaces en dicha gestión.

----- *Palabras clave:* Gestión de riesgos, desarrollo de software, estándares de gestión de proyectos, encuesta

#### Abstract

Software is today a critical element in every system. The development and implementation of computer programs is influenced by many varied risks which should be managed properly. Risk management in software projects is an activity addressed in several software methodologies, but the different workgroups and organizations apply it in practice in many different ways. A preliminary study has been conducted among Spanish software professionals and developers. It reveals interest in risk management and its techniques, but it also shows that serious deficiencies as well as attitudes which do not help to this activity. These preliminary data represent a basis for a deeper and more detailed analysis of risk management practices which might lead to more effective and practical solutions.

----- *Keywords:* Risk management, software development, project management standards, survey

---

\* Autor de correspondencia: teléfono: +34+91+8856935, fax: +34+91+885 66 46, correo electrónico: luis.fernandezs@uah.es (L. Fernández)

## Introducción

En la historia reciente de la ingeniería muchos incidentes están relacionados con problemas en el software. Pero estos problemas rara vez ocupan las páginas de los periódicos, y solamente llaman nuestra atención cuando se producen en sistemas críticos, causando pérdida de vidas humanas o arruinando grandes inversiones. Es el caso de la explosión del cohete Ariane V en 1996 debida a una reutilización fallida de software [1], o los fallos de programación en las primeras versiones de los misiles Patriot que, a pesar de su superioridad en el aire, resultaban totalmente ineficaces contra los misiles Scud iraquíes [2], uno de cuyos ataques costó la vida a 28 personas.

Los estudiosos del tema, como Peter G. Neumann, llevan más de 25 años observando y cuantificando los riesgos derivados de un mal desarrollo o un uso indebido de los sistemas informáticos y sus consecuencias. La tabla 1 recoge algunas cifras extraídas del Risk Digest [3]. Debe señalarse que no se incluyen todas las categorías del trabajo original, y que un mismo riesgo puede hallarse bajo más de un epígrafe.

**Tabla 1** Número de riesgos recogidos por Peter G. Neumann

<i>Categoría</i>	<i>Nº riesgos</i>
Sobre proyectos de defensa	104
Sobre proyectos de espacio	121
Sobre proyectos de aviación comercial	370
Sobre proyectos de medicina y salud	221
Problemas de calendario	250
Problemas de seguridad	363
Problemas de privacidad	479
Pérdida de vidas humanas	188
Pérdida de recursos durante el proyecto	1.384
Problemas en el desarrollo del sistema	193
Problemas de evolución, mantenimiento, actualización	433
Problemas de requisitos	178
Defectos en diseño o implementación	2.017
Errores humanos con interfaz hombre máquina	810
Riesgos identificados en total en Risk Digest [3]:	4.442

En este artículo se hablará sobre el riesgo asociado al desarrollo de software, su gestión, y los principales problemas de ésta. Se expondrán definiciones, encuestas realizadas en todas las épocas y, por último, se mostrará los primeros resultados de una encuesta realizada en España entre los profesionales del software. El artículo concluye las conclusiones obtenidas a la vez que describe las líneas de profundización en el estudio de esta disciplina que los autores ya están desarrollando.

## El riesgo

Según el diccionario Webster [4] el riesgo puede definirse como la posibilidad de un daño, desventaja o destrucción. Es un ente que algunos autores [5-9] caracterizan como bidimensional y que representa típicamente una función de la probabilidad de ocurrencia de un fallo y de la gravedad de sus consecuencias. Esta doble dimensión es lo que para algunos autores [9] diferencia el riesgo y el peligro, que solo se caracteriza por una dimensión de severidad contra las personas y los equipos. El riesgo siempre está presente en todas las actividades humanas individuales y de grupo, incluyendo las de ingeniería, y dentro de estas, las de desarrollo de software. Para el PMBOK [10] del Project Management Institute el riesgo es un evento o condición incierta que si ocurre, tendrá un efecto positivo o negativo en los objetivos del proyecto. Como puede verse, en esta definición tendrían también cabida los efectos positivos de un evento. La tabla 2 recoge algunos matices comunes a definiciones de riesgo extraídas de la literatura [4-13]

Los riesgos detectados en un proyecto inciden de dos formas en el mismo. A corto plazo, van a condicionar la decisión sobre cuál va a ser la siguiente acción a tomar, encaminada a evitar, contrarrestar o asumir el riesgo detectado. A medio y largo plazo, los riesgos detectados o experimentados en proyectos pasados pueden determinar también los niveles de calidad y las acciones que se van a exigir a los proyectos futuros.

**Tabla 2** Diferentes matices contemplados en definiciones de riesgo

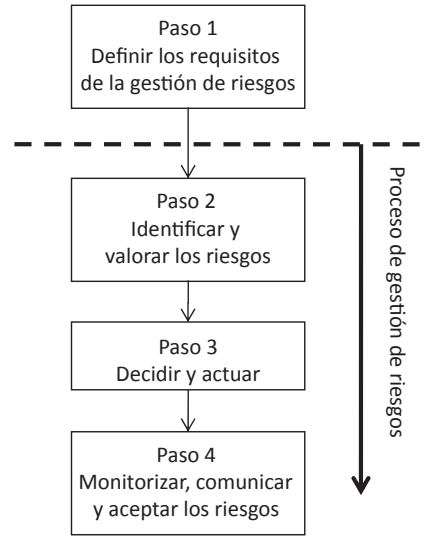
<i>origen</i>	<i>probabilidad detectabilidad</i>	<i>impacto</i>	<i>enfase otros riesgos</i>	<i>certeza</i>	<i>consecuencias positivas</i>	<i>consecuencias negativas</i>
RAE [11]		√	√			√
Webster [4]	√	√				
Rowe [6]	√	√				√
Defence Systems [5]	√	√				
ECCS [12]	√					√
Rosenberg [7]	√	√		√		√
PMBOK [10]	√	√			√	√
US Air Force [13]				√		
Moore [14]		√			√	√
Stamatelatos [9]	√	√				√
Charette [8]	√	√				√
Charette [8] (atributos)	√	√	√	√		√

### Métodos para la gestión de riesgos

La gestión de riesgos puede definirse como el proceso sistemático de identificación, análisis y respuesta a los riesgos que se presentan durante el ciclo de vida de un proyecto [10]. Su objetivo principal es minimizar la probabilidad y las consecuencias de los eventos perjudiciales. Se considera un proceso más dentro de las metodologías de gestión.

La gestión de riesgos se ha regulado e incluso estandarizado en muchas ocasiones. Generalmente las organizaciones que desarrollan proyectos críticos son las más preocupadas por contar con normas claras y comunes para su proceso de gestión de riesgos. Por ejemplo, en el ámbito espacial las normas ECCS de la Agencia Espacial Europea (ESA) [12] y el NPG7120.5 de la NASA [15] incluyen guías (Figura 1) sobre

cómo llevar a cabo esta gestión. La famoso PMBOK [10] dedica también un capítulo entero a esta tarea.



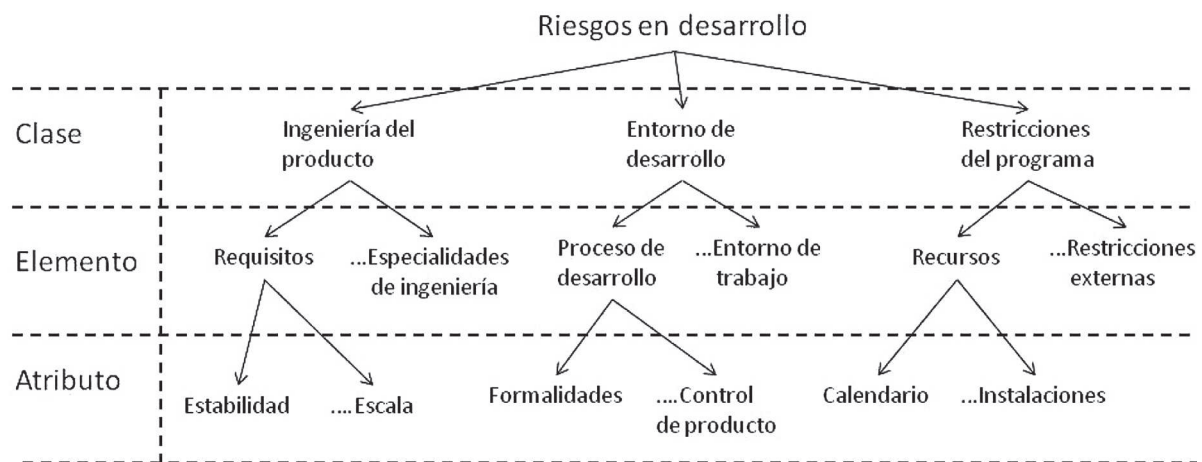
**Figura 1** Ciclo de gestión de riesgos en la ESA. (adaptado de [12])

Las tres fuentes señaladas coinciden al señalar las principales fases en la gestión de riesgos: una fase de planificación que se encuadra dentro de la planificación general del proyecto, y en las que se decide cómo se va a gestionar el riesgo en todo el ciclo de vida; una fase de identificación de los riesgos que pueden afectar al proyecto; un análisis cualitativo, cuantitativo o ambos de los riesgos detectados, para evaluar sus efectos y priorizar las posibles acciones; una planificación de la respuesta a aplicar, que puede ir desde asumir los riesgos íntegramente, esquivarlos, o hasta realizar acciones específicas de mitigación; y por último una etapa de monitorización y control para seguir la evolución de los riesgos tratados y proporcionar información más allá del ámbito del proyecto.

Las fases interactúan entre sí y con otros procesos de gestión. Para las fases de identificación y análisis de riesgos aparecen en la literatura dos ayudas diferentes. Por un lado están las taxonomías de riesgos, que buscan identificar los riesgos a partir de una clasificación completa y estática

de los factores que intervienen en un proyecto. Su punto fuerte es la exhaustividad, puesto que intentan abarcar toda la tipología de proyectos y situaciones susceptibles de albergar riesgos. Por el contrario, su punto débil es su tamaño, puesto que el usuario debe analizar un gran número de situaciones para llegar a identificar los riesgos que le pueden afectar. Como ejemplos de taxonomías de riesgos podemos citar las propuestas por [16-21]. La más conocida es tal vez la taxonomía de riesgos del Software Engineering Institute (SEI)

[22], descrita en la figura 2. En esta clasificación los riesgos se agrupan en tres clases que a su vez se dividen en elementos, cada uno de los cuales contempla diversos atributos. El resultado es un marco formado por 79 categorías que puede usarse para identificar riesgos en un proyecto. No todas las taxonomías de riesgos pretenden abarcar todas las facetas de los proyectos. [23] se centra en la práctica del outsourcing, mientras que [24] apunta su estudio en el desarrollo de software científico y de ingeniería.



**Figura 2** Taxonomía de riesgos del SEI (adaptado de [22])

Por otro lado están las listas con los factores de riesgo más comunes. Sin ánimo de ser exhaustivas ordenan los factores de acuerdo con su popularidad entre los profesionales de software. Estas listas tienen a su favor su sencillez, puesto que generalmente se trata de estructuras cortas, y su alto grado de acierto en el ámbito para el que se elaboraron. El principal inconveniente es la posibilidad de que algunos riesgos poco comunes no se estén considerando. A continuación se repasan algunas de las listas de riesgos más conocidas.

El equipo de [25] identificó cinco categorías principales donde se situaban los riesgos en los proyectos software: la tecnología novedosa, el

tamaño de las aplicaciones, la falta de experiencia del equipo de desarrollo, la complejidad en general de la aplicación, y el entorno de la organización.

Para [26], son también cinco los factores principales que generan riesgo, pero en este caso, diferentes: una planificación inherentemente fallida, requisitos cambiantes, los movimientos de personal, una especificación ambigua, y la baja productividad.

El trabajo de [27] es un profundo estudio realizado en la Arizona State University. Su objeto era generar un modelo para la valoración de riesgos en el software a través de sus efectos potenciales

durante el desarrollo. Primeramente se extrajeron 21 factores de riesgo comunes en la literatura, y a continuación, se consultó a los expertos para elaborar un modelo con sus relaciones y sus efectos.

El norteamericano Capers Jones ha cuantificado el coste de los principales riesgos en el desarrollo de software sobre el coste total de desarrollo [28]. En total, se identifican 11 factores de riesgo como los más significativos.

Por otra parte, [29] identifica 28 factores de riesgo a partir de siete trabajos anteriores distintos, como punto de partida para su estudio.

Las conclusiones de estos estudios son muy dependientes de la época en que se realizaron y del ámbito en el que se recogieron los datos, puesto que la naturaleza de los grupos de trabajo, de las tecnologías y de los proyectos varía sustancialmente. Si queremos conocer la situación de los riesgos en desarrollo de software en España en la actualidad resulta necesario preguntar directamente a los profesionales del software. Este es el trabajo que describe la segunda parte de este artículo.

### **Problemas en la gestión de riesgos**

Muchos proyectos de desarrollo de software se ven amenazados e incluso acaban de forma fallida entre otras razones por una mala gestión de sus riesgos. Incluso muchas veces en los ámbitos de desarrollo más críticos como el software espacial o el aeronáutico la gestión de riesgos ni siquiera se lleva a cabo de forma manera sistemática.

Para algunos autores como [30] una razón para esta crisis en la gestión de riesgos puede ser el componente subjetivo que tienen éstos, y la fuerte dependencia del observador. Mientras que es posible cuantificar de forma objetiva costes o esfuerzos cuyos datos provengan de distintas fuentes, resulta difícil combinar, valorar y dar la credibilidad adecuada a las observaciones de riesgos de diferentes personas o en diferentes contextos. La subjetividad de los observadores

puede afectar incluso al propio concepto de éxito y fracaso en proyectos de software. En algunos proyectos este concepto depende exclusivamente de la percepción de las personas que participan en él como el jefe de proyecto, desarrolladores y usuarios [31-33]. La diferencia puede explicarse por la importancia relativa que unos y otros otorgan a aspectos como el cumplimiento de la planificación, de los requisitos de usuario, la satisfacción final de este, y otros [34-37]

Para [30] las dificultades más comunes con las que se encuentran los grupos de trabajo para aplicar los métodos de gestión de riesgos son: la falta de definición de riesgos, la ausencia de una clasificación de riesgos completa, la falta de una definición del proceso de riesgos y la falta también de definición del proceso de comunicación de riesgos.

Los problemas en la aplicación de una gestión de riesgos es un tema poco tratado en la literatura. Por esta razón el estudio recogido en la segunda parte de este artículo dedica algunas de sus preguntas a profundizar sobre esta realidad y sus causas.

### **Un análisis de la situación en España**

Durante la elaboración de este artículo se llevó a cabo un estudio piloto para conocer la situación real de la gestión de riesgos en proyectos de desarrollo de software. Sus conclusiones deberían servir de base para un estudio más profundo que los autores ya están desarrollando. En concreto, eran tres los aspectos a conocer.

La primera cuestión era el grado de conocimiento y de sensibilización sobre los riesgos y su gestión que tenían los profesionales españoles dedicados al desarrollo de programas. En segundo lugar, se quería conocer cuáles eran las principales prácticas relativas a la gestión de riesgos en proyectos y en qué medida aparecían los problemas identificados en la gestión. Por último se consideró del máximo interés saber cuál era para los desarrolladores españoles la lista de los riesgos más comunes.



En la actualidad existen múltiples plataformas para desplegar test en Internet y gestionar encuestas. Entre las gratuitas probablemente las más conocidas sean Lime Survey [38] en el ámbito académico y Google Docs Forms [39] para propósito general. En nuestro trabajo empleamos Lime Survey versión 1.91 para elaborar y gestionar la encuesta.

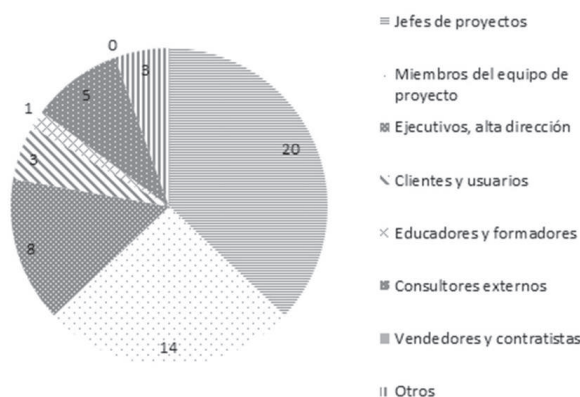
### Primeros resultados

Para llevar a cabo el estudio planteado nos interesaba agrupar a los encuestados en función de varios parámetros profesionales. En concreto se preguntó a los encuestados sobre su experiencia, los sectores de actividad de sus empresas (tomando como sectores de partida los contemplados en la Clasificación Nacional de Actividades Económicas (CNAE) [40]), los tipos de proyectos software donde trabajaban (tomando como referencias la Clasificación Estadística de Productos por Actividades CPA 2008 [41] y la clasificación de proyectos software de [28]). También parecía importante distinguir cómo percibían los riesgos los distintos integrantes de un mismo proyecto de desarrollo software (la clasificación de partida de los roles en un proyecto nos la proporcionaron el Capability Maturity Model Integration (CMMI) [42], el PMBOK [10] y la Clasificación Internacional Uniforme de Ocupaciones (ISCO) [43]).

La tabla 3 recoge algunas de las cifras principales del perfil de los encuestados. En la figura 3 puede verse cómo se distribuyen los roles actuales de los participantes en la muestra. El promedio de experiencia de cada uno en su rol es de 8,1 años.

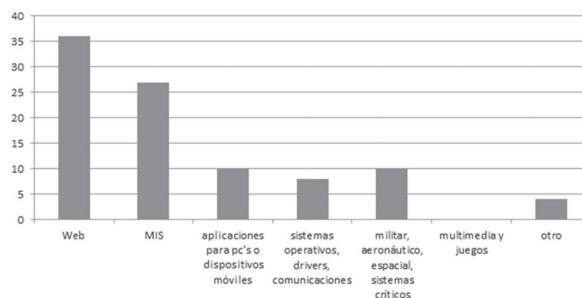
**Tabla 3** Principales cifras de la encuesta

<i>Magnitud</i>	<i>Valor</i>
Encuestas recogidas:	53
Total experiencia:	717 años
Experiencia media en software:	13,5 años
<b>Principales áreas donde han trabajado los encuestados alguna vez:</b>	
Información y comunicaciones	46%
Actividades financieras y seguros	44,4%
Administración pública, defensa y sanidad pública	42,5%



**Figura 3** Roles de los encuestados

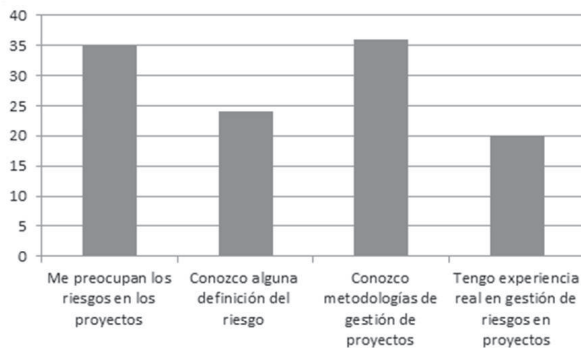
La figura 4 recoge los resultados recibidos sobre los tipos de software con que trabajan los encuestados. Puede destacarse que 10 de los profesionales trabajan con software militar, aeronáutico, espacial o de sistemas críticos en general. Más adelante repasaremos sus resultados con más detalle.



**Figura 4** Tipo de software con que trabajan los encuestados

### Importancia de la gestión de riesgos

Una parte importante del estudio se refirió a la importancia que conceden los encuestados a la gestión de riesgos en general y a la manera en que materializan ese interés en los proyectos. La figura 5 recoge la percepción general de los encuestados.



**Figura 5** Percepción de los riesgos

El área donde más preocupa el riesgo fue, en primer lugar, la gestión de tiempos y en segundo lugar, por igual, la gestión de costes y la gestión del proyecto y su integración. En el último puesto estaba tanto la gestión de la comunicación como la de suministros.

Cuando se preguntó a los usuarios en qué parte del proyecto los riesgos eran más frecuentes, sin duda el más nombrado fue la integración de las partes del sistema y la integración con partes desarrolladas por terceros, quedando la gestión de requisitos en segundo lugar. Sin embargo, cuando preguntamos por los riesgos más graves, los usuarios apuntaron, en primera posición, a los procesos de gestión del proyecto (como alcance, tiempos, costes, calidad, riesgos, suministros) y, en segundo lugar, a la integración de las partes del sistema y con partes de terceros. Curiosamente los requisitos quedan relegados a una quinta posición, como si la disponibilidad de tiempo por delante en el proyecto pudiera mitigar cualquier problema de requerimientos.

En cuanto a los aspectos del proyecto que se consideraron como más importantes, los primeros resultados fueron los tiempos, los costes, y el plan de proyecto e integración. En el extremo opuesto se situaron la comunicación y los suministros. Sobre los aspectos dónde es más frecuente que haya problemas, los encuestados destacaron la integración de las partes, y los requisitos y restricciones de calendario.

Profundizando en la experiencia del usuario con la gestión de riesgos, una pregunta obligada era qué hábitos o qué acciones concretas llevaba a cabo como individuo o como organización relativas a la gestión de riesgos. La tabla 4 recoge algunos resultados.

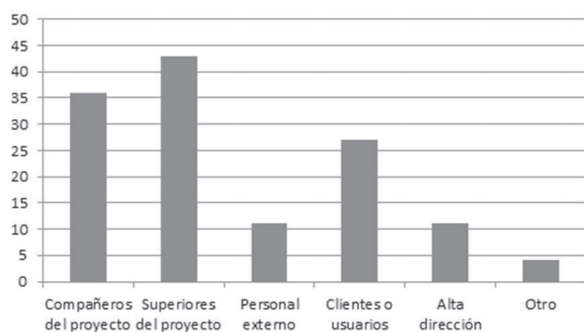
**Tabla 4** Prácticas de gestión de riesgo

<i>Ítem</i>	<i>De acuerdo</i>
En su grupo de trabajo se utiliza alguna herramienta para la gestión de riesgos (p. ej.: específicas, hojas de cálculo, bases de datos, etc.)	55,56%
Se usa alguna herramienta de estimación para dimensionar el proyecto (tiempos, recursos, costes).	59,26%
Su proyecto tiene disponible información de riesgos de proyectos pasados de la organización, cómo listas, buenas prácticas, estadísticas, etc.	18,52%
Al acabar un proyecto, se lleva a cabo algún informe del desarrollo donde puedan indicarse problemas aparecidos y su impacto, o cómo se combatieron.	20,37%

### Problemas en la gestión de riesgos

Sobre las razones para no llevar a cabo una gestión de riesgos, solo uno de los encuestados la consideró innecesaria con los sistemas de calidad. Un 42% lo consideró inviable por razones de carga de trabajo y tiempos, y un 50% se atrevió a afirmar que no hay una cultura de buscar problemas por adelantado.

La figura 6 recoge las respuestas de los encuestados a la pregunta sobre con qué personas les parecía positivo hablar de riesgos en un proyecto. Como puede verse, existe una fuerte reticencia a hablar de posibles riesgos con la alta dirección, así como con personal externo como asesores o colaboradores de otras empresas.



**Figura 6** Con qué personas le parece positivo hablar de riesgos en un proyecto

En cuanto a la posible influencia de las prácticas de gestión de riesgos en un proyecto las respuestas de los encuestados se recogen en la tabla 5.

**Tabla 5** Influencia de las prácticas de gestión de riesgo en el proyecto

<i>Ítem</i>	<i>De acuerdo</i>
Tienen una influencia objetiva en el éxito del proyecto	62,96%
Tiene una influencia objetiva en la percepción del éxito del proyecto por sus miembros	37,04%
Identificar los riesgos es positivo, medir los riesgos y abordarlos sistemáticamente no tiene demasiados efectos	16,67%
Es mejor para el éxito del proyecto una gestión sistemática realizada por externos que una gestión 'relajada' por los miembros del proyecto	27,78%
Es mejor para el éxito del proyecto que los miembros del proyecto hablen de riesgos y se autogestionen, aunque sus prácticas no sean del todo ortodoxas.	35,19%
Otro	3,70%

Como puede verse no existe consenso sobre si la gestión la deben llevar a cabo externos profesionales o por el contrario lo debe hacer personal interno. Tampoco coincide con los resultados de Bakker [44], para los que el efecto positivo reside más en identificar los riesgos y hablar de ellos que en medirlos o abordarlos de una manera sistemática.

Cuando preguntamos a los encuestados sobre las definiciones del éxito en un proyecto, son mayoría los que vinculan este concepto al proyecto y al producto frente a los que vinculan el éxito a resultados para la entidad o el grupo de trabajo, como la continuidad o el beneficio económico. La tabla 6 recoge estos resultados.

**Tabla 6** Concepto de éxito de un proyecto

<i>Ítem</i>	<i>De acuerdo</i>
Satisfacer los requisitos fijados en el tiempo y coste acordado	74,07%
Que al cliente o al usuario le guste el resultado final	61,11%
Que posibilite una continuidad en la labor o con el cliente	29,63%
Que proporcione beneficios económicos o de otro tipo	35,19%
Otro	3,70%

### Factores de riesgo

Por último, en este estudio preliminar nos parecía útil contar con una lista de factores de riesgo más significativos entre los desarrolladores españoles, para poder compararla con las obtenidas en otras encuestas. Para ello se tomaron como partida 24 factores de riesgo presentes en otras trabajos y se pidió a los encuestados que los valoraran en las dimensiones de gravedad de sus consecuencias, probabilidad de ocurrencia, y una tercera indicando cuales eran los más cuidados o vigilados en sus proyectos.

Los resultados preliminares más destacados pueden verse en la tabla 7:

**Tabla 7** factores de riesgo elegidos por los encuestados

<i>Los más graves</i>
Requisitos de usuario cambiantes, o que son mal entendidos o interpretados
Las planificaciones o los presupuestos no son realistas
El cliente o el usuario no se implican adecuadamente
Se abandona la planificación con la presión



**Los más frecuentes**

- Las planificaciones o los presupuestos no son realistas
- Requisitos de usuario cambiantes, o que son mal entendidos o interpretados
- Se abandona la planificación con la presión
- Falta de personal en general

**Los que más se cuidan**

- El código fuente no se controla, o es muy malo
- Las planificaciones o los presupuestos no son realistas
- Requisitos de usuario cambiantes, o que son mal entendidos o interpretados
- Problemas en el rendimiento en tiempo real, o tiempos de respuesta deseados de la aplicación

**Con más votos**

- Mal desarrollo del interfaz de usuario
- Requisitos de usuario cambiantes, o que son mal entendidos o interpretados
- El cliente o el usuario no se implican adecuadamente
- El código fuente no se controla, o es muy malo

**Sistemas críticos versus no críticos**

De entre las encuestas analizadas hay 10 que corresponden a profesionales que trabajan en el desarrollo de software crítico como el espacial y el militar. El análisis preliminar de los resultados ha revelado algunas diferencias en su gestión del riesgo y los factores de riesgo más significativos que vale la pena estudiar más a fondo. Un 90% de los profesionales de proyectos de software críticos considera que las prácticas de gestión de riesgos tienen influencia objetiva frente a un 58% del resto de los grupos. También parece que les preocupa más que a otros grupos la integración con partes de terceros, las restricciones de calendario, costes y equipos, y la gestión de contrato. Entre las prácticas de gestión de riesgos también puede destacarse que es mucho mayor el porcentaje de profesionales que al acabar un proyecto elabora un informe reflejando los problemas encontrados, y posibilitando así una experiencia compartida.

**Conclusiones**

Este trabajo recoge describe el estado del arte y los principales problemas de la gestión de

riesgos en el desarrollo de software. Existe una abundante normativa sobre cómo identificar y tratar los riesgos. Además se dispone facilidades como son las taxonomías y las listas de factores de riesgos más conocidos.

El trabajo presenta también los primeros resultados de una encuesta piloto realizada en España sobre el tema. Los primeros datos parecen ser optimistas sobre el conocimiento de la gestión de riesgos por parte de los encuestados, la experiencia real de muchos de ellos, y la percepción positiva de su aplicación a los proyectos. Sin embargo llama la atención el contraste entre buen nivel de conocimiento y uso de las herramientas disponibles para gestionar los riesgos y las dificultades que parecen existir para acceder a información de proyectos anteriores así como la aparente falta de disciplina a la hora de documentar los problemas aparecidos durante el desarrollo de un proyecto. Más allá de contar con los medios aun es preciso crear actitudes favorables a gestionar los riesgos de una forma más sistemática.

La gestión de la comunicación es una de las que menos preocupa a los encuestados. A la hora de hablar sobre posibles riesgos sí parecen existir tabúes hacia el personal externo, y hacia la alta dirección. Esta falta de diálogo con la alta dirección de los proyectos podría tener consecuencias a medio plazo impidiendo que los procesos de gestión de riesgos se extiendan por las organizaciones.

En lo relativo a los factores de riesgo más populares, la importancia de los requisitos parece ser una constante en todas las categorías, pero sin embargo puede verse diferencias en los demás. Con los datos disponibles podría decirse que los factores de riesgo como más graves se concentran en el lado del cliente, los más frecuentes en el proyecto, y los más cuidados en el producto. Esta diversidad merece un estudio más detallado, pero podría ser un buen punto de partida para mejorar el esfuerzo de los desarrolladores de software en la gestión de sus riesgos.

Como conclusión final puede decirse que aunque los datos recogidos muestran ya algunos avances

sobre la información existente en la literatura, es preciso seguir investigando para establecer un modelo completo de la gestión de los riesgos en proyectos. Por esta razón el trabajo continúa realizándose afinando los cuestionarios en los aspectos más discutidos, aumentando el número y la variedad de los encuestados, y profundizando en el análisis de los datos recogidos. Uno de los ámbitos donde se más va a trabajar será el de los sistemas críticos, con el fin de conocer mejor las diferencias con otros tipos de proyectos.

### Referencias

1. J. Lions, Chairman. *Ariane 501*. Ed. Inquiry Board Report. ESA y CNES. Paris, France. 1996. pp.1-60.
2. Information Management and Technology Division. *GAO/IMTEC-92-26 Patriot Missile Software Problem*. General Accounting office. Washington DC., USA. 1992. pp.1-16. Available on: <http://www.gao.gov/products/IMTEC-92-26>. Accessed: August 6, 2012.
3. P. Neumann. *Forum on Risks to the Public In Computers and Related Systems*. Committee on Computers and Public Policy of the Association for Computing Machinery. 1985. Available on: <http://catless.ncl.ac.uk/Risks>. Accessed: August 4, 2012.
4. *Webster's On-line dictionary*. 2005. Available on: <http://www.websters-online-dictionary.org>. Accessed: July 2, 2012.
5. Defense Systems Management College. *Risk Assesment Techniques*. 1<sup>st</sup> ed. Ed. Fort Velvoir. Virginia, USA. 1983. pp. 1-18.
6. W. Rowe. *An Anatomy of Risk*. 1<sup>st</sup> ed. Ed. Krieger Publishing Co. Florida, USA. 1988. pp. 1-488.
7. L. Rosenberg, A. Hammer, T. Gallo. *Continuous Risk Management at NASA*. Applied Software Measurement / Software Management Conference. San Jose, USA. 1999. pp. 1-34.
8. R. Charette. *Software Engineering Risk Analysis and Management*. 1<sup>st</sup> ed. Ed. McGraw-Hill. New York, USA. 1989. pp. 1-325.
9. M. Stamatelatos, H. Dezfuli. *Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners*. 2<sup>nd</sup> ed. Available on: <http://www.hq.nasa.gov/office/codeq/risk/index.htm>. Accessed: November 28, 2013.
10. Project Management Institute. *PMBOK: A Guide to the Project Management Body of Knowledge*. 2000. Ed. Newton Square. Pennsylvania, USA. 2000. pp. 1-211.
11. Real Academia Española. *Diccionario de la lengua española*. 22<sup>nd</sup> ed. Ed. Espasa libros, S. L.U. Madrid, España. 2001. pp. 1975-1976.
12. ECSS Secretariat. ESA-ESTEC Requirements & Standards. *ECSS-M-00-03a. Space Project Management. Risk management*. 1<sup>a</sup> ed. Ed. ESA-ESTEC. Noordwijk, The Netherlands. 2000. pp. 1-40.
13. Corporate Air Force Systems Command. Software Risk Abatement. Boehm, Barry W. (editor) *Software Risk Management*. 1<sup>a</sup> ed. Ed. IEEE Press. Piscataway, USA. 1989. pp. 148-171.
14. A. Moore, A. Fearon, M. Alcock. *Implementation of Opportunity and Risk Management in BAE SYSTEMS Astute Class Limited - A Case Study*. Proceedings of the Fourth European Project Management Conference, PMI Europe2001. London, UK. 2001. pp. 1-7.
15. J. Ansell, F. Wharton. *Risk Analysis Assessment and Management*. 1<sup>a</sup> ed. Ed. John Willey & Sons LTD. Chichester, England. 1992. pp. 1-220.
16. J. Calvo, L. Maté, T. San Feliú. *A Risk Management Approach*. T.NATO AC/243 Panel 11 Research Study Group 3. Madrid, España. 1993. pp.1-22.
17. J. Ropponen, L. Kalle. "Components of software development risk; how to address them? A project manager survey". *IEEE Transactions on Software Engineering*. Vol. 26. 2000. pp. 98-112.
18. J. Verner, S. Overmyer, K. McCain. "In The 25 Years Since The Mythical Man-Month What Have We Learned About Project Management?". *Information and Software Technology*. Vol. 41. 1999. pp. 1021-1026.
19. J. Verner, W. Evanco. *The state of the practice of software effort estimation in business organizations*. Proceedings of ESCOM-SCOPE Conference. Munich, Germany. 2000. pp. 1-5.
20. J. Procaccino, J. Verner. *Early Risk factors for Software Development*. Proceedings of the 12<sup>th</sup> European Software Control and Metrics Conference. London, England. 2001. pp. 107-116.
21. C. Robbie, T. Nakatsu. "A comparative study of important risk factors involved in offshore and domestic outsourcing of software development projects: A two-panel Delphi study". *Information & Management*. Vol. 46. 2009. pp. 57-68.

22. B. Gallagher, P. Case, R. Creel, S. Kushner, R. Williams. *A Taxonomy of Operational Risks (CMU/SEI-2005-TN-36)*. 1<sup>st</sup> ed. Ed. Carnegie Mellon Software Engineering Institute. Pittsburgh, USA. 2005. pp.1-29.
23. G. Manrique. "Método para la construcción de una taxonomía: estructura base para riesgos en outsourcing de software". *Revista Facultad de Ingeniería de la Universidad de Antioquía*. N° 60. 2011. pp. 92-101.
24. R. Kendall, D. Post, J. Carver, D. Henderson, D. Fisher. *A Proposed Taxonomy for Software Development Risks for High-Performance Computing (HPC) Scientific/Engineering Applications*. Ed. Software Engineering Institute. Pittsburgh, USA. 2007. pp. 1-27.
25. H. Barki, S. Rivard, J. Talbot. "Toward an Assessment of Software Development Risk". *Journal of Management Information Systems*. Vol. 10. 1993. pp. 203-225.
26. T. Demarco, T. Lister. *Waltzing With Bears: Managing Risks on Software Projects*. 1<sup>a</sup> ed. Ed. Dorset House Publishing Company. New York, USA. 2003. pp.1-208.
27. D. Houston, J. Collofello. "Finding the Influential Factors in Software Process Simulation Models". *Journal of Systems and Software*. Vol. 59. 2001. pp. 259-270.
28. T. Capers Jones. *Estimating Software Costs*. 2<sup>a</sup> ed. Ed. McGraw-Hill Professional. New York, USA. 2007. pp. 1-644.
29. R. Schmidt, K. Lyytinen, M. Keil, P. Cule. "Identifying Software Project Risks: An International Delphi Study". *Journal of Management Information Systems*. Vol. 17. 2001. pp. 5-36.
30. T. San Feliú. *MANRIS. Método de Análisis de Riesgos de Sistemas de Información*. Tesis doctoral de la Universidad de Castilla-La Mancha. Toledo, España. 2000. pp. 1-219.
31. J. Pereira, N. Cerpa, R. N. *Risk factors in software development projects: Exploratory analysis of the Chilean software industry*. Proceedings of the First Experimental Software Engineering Latin American Workshop. Brazilia, Brazil. 2004. pp.51-56.
32. E. Oz, J. Sosik. "Why information systems projects are abandoned: a leadership and communication theory and exploratory study". *Journal of Computer Information Systems*. Vol 41. 2000. pp. 66-78.
33. K. Walsh, H. Schneider. "The role of motivation and risk behavior in software development success". *Information Research*. Vol. 7. 2002. Available on: <http://InformationR.net/ir/7-3/paper129.html>. Accessed: July 1, 2012.
34. J. Pinto, D. Slevin. "Project success: definitions and measurement techniques". *Project Management Journal*. Vol. 1. 1988. pp. 67-73.
35. D. Baccarini. "The Logical Framework Method for Defining Project Success". *Project Management Journal*. 1999. Vol. 30. pp. 25-32.
36. C. Wohlin, A. Mayrhauser. "Assessing Project Success using Subjective Evaluation factors". *Software Quality Journal*. Vol. 9. 2001. pp. 43-70.
37. R. Pressman. *Software Engineering: A Practitioners Approach*. 7<sup>a</sup> ed. Ed. McGraw Hill. New York, USA. 2009. pp.1-895.
38. C. Schmitz. *Lime Survey: the free & open source survey software tool!*. Available on: <http://www.limesurvey.org/>. Accessed: July 1, 2012.
39. Google Team. *Google Docs Help: Create, send, share, and edit a form*. Available on: <http://support.google.com/docs/bin/answer.py?hl=en&answer=87809>. Accessed: July 1, 2012.
40. Ministerio de Economía y Hacienda. *Real Decreto 475/2007, de 13 de abril, por el que se aprueba la Clasificación Nacional de Actividades Económicas 2009 (CNAE-2009)*. Boletín Oficial del Estado. N° 102. Madrid, España. 2007. pp. 18572-18593.
41. Unión Europea. *Reglamento (ce) no 451/2008 del Parlamento Europeo y del Consejo*. Diario Oficial de la Unión Europea. N°. 145. 2008. pp. 145/65- 145/226.
42. M. Philips. *CMMI Today*. Software Engineering Institute. Pittsburgh, USA. 2004. Available on: <http://resources.sei.cmu.edu/library/asset-view.cfm?assetID=21074>. Accessed: July 2, 2012.
43. Unión Europea. *Recomendación de la Comisión de 29 de Octubre de 2009 relativa al uso de la Clasificación Internacional Uniforme de Ocupaciones (CIUO-08)*. Diario Oficial de la Unión Europea. N°. 292. 2009. pp. 292/31-292/47.
44. K. De Bakker. "Communicative Project Risk Management in IT Projects". *CEPIS Upgrade*. Vol. 12. 2012. pp. 59-66.