

GUÍA DE APRENDIZAJE - TEMA 7

GESTIÓN DE RIESGOS EN PROYECTOS SOFTWARE

Objetivos:

Plantear la importancia de realizar una adecuada gestión de riesgos en los proyectos software. Conocer los riesgos más habituales y algunos métodos para mantener el control.

Índice:

1. Introducción.
2. Planificación de los riesgos.
 - 2.1. Identificación.
 - 2.2. Análisis.
 - 2.3. Priorización.
 - 2.4. Planificación de respuestas.
3. Supervisión y control de riesgos.
 - 3.1. Supervisión.
 - 3.2. Resolución o control.

Bibliografía utilizada:

CON	Connell, S. Desarrollo y Gestión de Proyectos Informáticos. McGraw-Hill Iberoamericana, 1997.	Cap. 5
PMI	Project Management Institute, A Guide to the Project Management Body of Knowledge, USA 2000.	Cap. 11

NOTAS DE USO:

- Se indica la referencia de la bibliografía básica (siglas y número de página). Ejemplo: GIL-9/10 significa páginas 9 a 10 de la referencia GIL (ver tabla de bibliografía básica).
- Las transparencias que corresponden se indican entre corchetes. Ejemplo: [T13].
- Se incluyen Ejercicios para realizar por parte de los alumnos. Se identifican con el símbolo ➤.

Contenido:

Introducción

[T5]

- MAPA DEL TEMA RESPECTO A PMBOK.
- Lugar que ocupan en el marco PMI de Gestión de Proyectos los conceptos, técnicas y herramientas del tema:

Área	Proceso	Grupo	Conceptos, técnicas y herramientas C=conceptos, T=técnicas y herramientas, O=salidas
Riesgos	Identificación	Planificación	T: Listas de Comprobación de Riesgos C: Riesgos en Proyectos Software
	Análisis Cuantitativo	Planificación	T: Tablas de Estimación de Riesgos T: Priorización de Riesgos
	Planificación de Respuestas	Planificación	O: Planes de Contingencia O: Plan
	Supervisión y Control	Control	C. Supervisión de riesgos en proyectos software C: Resolución o control de riesgos en proyectos software

C: conceptos que amplían y extienden lo comentado del proceso en el tema 4, pero ahora particularizando en proyectos informáticos y especialmente proyectos software (PS).

T: técnicas y herramientas útiles en el proceso.

O: salidas (outputs), es decir, resultados del proceso.

Concepto de Riesgo

PMI-127/129, [T6]

Riesgo en un proyecto: evento o condición incierta que, en caso de ocurrir, tiene un efecto positivo o negativo sobre los objetivos de un proyecto.

Un riesgo tiene una causa y, si ocurre (evento de riesgo), una consecuencia (efecto).

Riesgos conocidos: aquellos que han sido identificados y analizados durante la planificación del proyecto.

Habitualmente se gestionan los riesgos con efecto negativo, es decir, aquellos que suponen una amenaza para el éxito del proyecto.

Concepto de Gestión de Riesgos

CON-92, [T7]

La función de la gestión de riesgos del software es identificar, estudiar y eliminar las fuentes de riesgo antes de que empiecen a amenazar la finalización satisfactoria de un proyecto software.

Dependiendo del momento en que se detecta la ocurrencia del riesgo, existen cinco niveles de gestión de riesgos:

1. Control de crisis: intentar controlar los riesgos cuando ya se han convertido en problemas.
2. Arreglar cada error: reaccionar rápidamente cuando ya se ha producido el riesgo.
3. Mitigación de riesgos: planificar con antelación el tiempo que necesitaría para cubrir riesgos en caso de que ocurran, pero sin intentar eliminarlos inicialmente.
4. Prevención: crear y ejecutar, como parte del proyecto software, un plan para identificar riesgos y evitar que se conviertan en problemas.
5. Eliminación de causas principales: identificar y eliminar los factores que pueden provocar la presencia de algún tipo de riesgo.

En este tema estudiaremos la gestión de riesgos en los niveles 4 y 5; si se pretende controlar los riesgos en los tres primeros niveles, está perdida la planificación del proyecto.

PMI-127, [T8]

Los seis procesos que forman la Gestión de Riesgos en el modelo PMI, son los siguientes:

a) De planificación:

- Planificación de la Gestión de Riesgos: decidir cómo abordar y planificar las actividades de gestión de riesgos en el proyecto.
- Identificación de Riesgos: determinar cuales son los riesgos que pueden afectar al proyecto y documentar sus características.
- Análisis Cualitativo de Riesgos: realizar un análisis cualitativo de los riesgos y sus condiciones para priorizar sus efectos sobre los objetivos del proyecto.
- Análisis Cuantitativo de Riesgos: estimar la probabilidad y consecuencias de los riesgos y sus implicaciones sobre los objetivos del proyecto.
- Planificación de Respuestas a los Riesgos: elaborar procedimientos y técnicas para oportunidades de mejora y reducir las amenazas a los objetivos del proyecto.

b) De Control:

- Supervisión y Control de Riesgos: supervisar los riesgos residuales, identificar nuevos riesgos, ejecutar planes de reducción de riesgos, y evaluar su efectividad a lo largo del ciclo de vida del proyecto.

[T9]

Nosotros los vamos a estudiar agrupados de la siguiente manera (grupos de procesos, procesos, actividades):

- De Planificación
 - Identificación
 - Estimación (en PMBOK son los dos procesos de análisis)
 - Análisis
 - Priorización
 - Planificación de Respuestas
- De Control
 - Supervisión y Control de Riesgos
 - Supervisión (monitorización)
 - Resolución o control

No siempre se mantiene esta estructura de procesos y tareas. Por ejemplo, en el libro de Connell la Planificación de respuestas (llamado ahora planificación de la gestión) se incluye en el Control en lugar de la planificación).

Planificación

Identificación

CON-94/95, [T10]

Identificar los riesgos potenciales que pueden ocurrir en el proyecto.

Los más habituales (aunque no de obligada aparición) en proyectos software son los mostrados en la tabla siguiente:

- Cambio de requisitos
- Meticulosidad en requerimientos o de los desarrolladores
- Escatimar en la calidad
- Planificaciones demasiado optimistas
- Diseño inadecuado
- Síndrome de la panacea (“esta herramienta ahorrará la mitad del trabajo”)
- Desarrollo orientado a la investigación (proyectos muy novedosos, más propios de investigación que de desarrollo)
- Personal mediocre

- Errores en la contratación
- Diferencias con los clientes

Listas de Comprobación (Checklists)

PMI-133, [T11]

Se construyen a partir de información histórica.

Ventaja: permiten una identificación de riesgos rápida y relativamente sencilla.

Desventaja: es prácticamente imposible tener una lista que incluye todos los posibles riesgos en un proyecto software. Se resuelve utilizando la lista como punto de partida, pero permitiendo después incluir nuevos riesgos específicos del proyecto.

CON-95/101, [T12]

Nosotros vamos a emplear una lista muy completa, que agrupa los riesgos de proyectos software en las siguientes categorías:

- A. Elaboración de la Planificación
- B. Organización y Gestión
- C. Ambiente/Infraestructura de Desarrollo
- D. Usuarios finales
- E. Cliente
- F. Personal Contratado
- G. Requisitos
- H. Producto
- I. Fuerzas Mayores
- J. Personal
- K. Diseño e Implementación
- L. Proceso

Esta lista será utilizada en el trabajo T4. De ahí los códigos incluidos. 111 riesgos.

A. Elaboración de la Planificación

- A.1. Las definiciones de la planificación, de los recursos y del producto han sido impuestas por el cliente o un directivo superior, y no están equilibradas.
- A.2. Planificación optimista, «mejor caso» (en lugar de realista, «caso esperado»).
- A.3. La planificación no incluye tareas necesarias.
- A.4. La planificación se ha basado en la utilización de personas específicas de un equipo, pero estas personas no están disponibles.
- A.5. No se puede construir un producto de tal envergadura en el tiempo asignado.
- A.6. El producto es más grande que el estimado (en líneas de código, en el número de puntos función, o en relación con el tamaño del proyecto anterior).
- A.7. El esfuerzo es mayor que el estimado (por líneas de código, número de puntos función, módulos, etc.).
- A.8. La reestimación debida a un retraso en la planificación es demasiado optimista o ignora la historia del proyecto.
- A.9. La presión excesiva en la planificación reduce la productividad.
- A.10. La fecha final ha cambiado sin ajustarse al ámbito del producto o a los recursos disponibles.
- A.11. Un retraso en una tarea produce retrasos en cascada en las tareas dependientes.
- A.12. Las áreas desconocidas del producto llevan más tiempo del esperado en el diseño y en la implementación.

B. Organización y Gestión

- B.1. El proyecto carece de un promotor efectivo en los superiores.
- B.2. El proyecto languidece demasiado en el inicio difuso.
- B.3. Los despidos y las reducciones de la plantilla reducen la capacidad del equipo.
- B.4. Dirección o marketing insisten en tomar decisiones técnicas que alargan la planificación.

- B.5. La estructura inadecuada de un equipo reduce la productividad.
- B.6. El ciclo de revisión/decisión de la directiva es más lento de lo esperado.
- B.7. El presupuesto varía el plan del proyecto.
- B.8. La dirección toma decisiones que reducen la motivación del equipo de desarrollo.
- B.9. Las tareas no técnicas encargadas a terceros necesitan más tiempo del esperado (aprobación del presupuesto, aprobación de la adquisición de material, revisiones legales, seguridad, etc.).
- B.10. La planificación es demasiado mala para ajustarse a la velocidad de desarrollo deseada.
- B.11. Los planes del proyecto se abandonan por la presión, llevando al caos y a un desarrollo ineficiente.
- B.12. La dirección pone más énfasis en las heroicidades que en informarse exactamente del estado, lo que reduce su habilidad para detectar y corregir problemas.

C. Ambiente/Infraestructura de Desarrollo

- C.1. Los espacios no están disponibles en el momento necesario.
- C.2. Los espacios están disponibles pero no son adecuados (por ejemplo, falta de teléfonos, cableado de la red, mobiliario, material de oficina, etc.).
- C.3. Los espacios están sobreutilizados, son ruidosos o distraen.
- C.4. Las herramientas de desarrollo no están disponibles en el momento deseado.
- C.5. Las herramientas de desarrollo no funcionan como se esperaba; el personal de desarrollo necesita tiempo para resolverlo o adaptarse a las nuevas herramientas.
- C.6. Las herramientas de desarrollo no se han elegido en función de sus características técnicas, y no proporcionan las prestaciones previstas.
- C.7. La curva de aprendizaje para la nueva herramienta de desarrollo es más larga de lo esperado.

D. Usuarios Finales

- D.1. Los usuarios finales insisten en nuevos requisitos.
- D.2. En el último momento, a los usuarios finales no les gusta el producto, por lo que hay que volver a diseñarlo y a construirlo.
- D.3. Los usuarios no han realizado la compra del material necesario para el proyecto y, por tanto, no tienen la infraestructura necesaria.
- D.4. No se ha solicitado información al usuario, por lo que el producto al final no se ajusta a las necesidades del usuario, y hay que volver a crear el producto.

E. Cliente

- E.1. El cliente insiste en nuevos requisitos.
- E.2. Los ciclos de revisión/decisión del cliente para los planes, prototipos y especificaciones son más lentos de lo esperado.
- E.3. El cliente no participa en los ciclos de revisión de los planes, prototipos y especificaciones, o es incapaz de hacerlo, resultando unos requisitos inestables y la necesidad de realizar unos cambios que consumen tiempo.
- E.4. El tiempo de comunicación del cliente (por ejemplo, tiempo para responder a las preguntas para aclarar los requisitos) es más lento del esperado.
- E.5. El cliente insiste en las decisiones técnicas que alargan la planificación.
- E.6. El cliente intenta controlar el proceso de desarrollo, con lo que el progreso es más lento de lo esperado.
- E.7. Los componentes suministrados por el cliente no son adecuados para el producto que se está desarrollando, por lo que se tiene que hacer un trabajo extra de diseño e integración.
- E.8. Los componentes suministrados por el cliente tienen poca calidad, por lo que tienen que hacerse trabajos extra de comprobación, diseño e integración.
- E.9. Las herramientas de soporte y entornos impuestos por el cliente son incompatibles, tienen un bajo rendimiento o no funcionan de forma adecuada, con lo que se reduce la productividad.
- E.10. El cliente no acepta el software entregado, incluso aunque cumpla todas sus especificaciones.
- E.11. El cliente piensa en una velocidad de desarrollo que el personal de desarrollo no puede alcanzar.

F. Personal Contratado

- F.1. El personal contratado no suministra los componentes en el período establecido.
- F.2. El personal contratado proporciona material de una calidad inaceptable, por lo que hay que añadir un tiempo extra para mejorar la calidad.
- F.3. Los proveedores no se integran en el proyecto, con lo que no se alcanza el nivel de rendimiento que se necesita.

G. Requisitos

- G.1. Los requisitos se han adaptado, pero continúan cambiando.
- G.2. Los requisitos no se han definido correctamente. y su redefinición aumenta el ámbito del proyecto.
- G.3. Se añaden requisitos extra.
- G.4. Las partes del proyecto que se no se han especificado claramente consumen más tiempo del esperado.

H. Producto

- H.1. Los módulos propensos a tener errores necesitan más trabajo de comprobación, diseño e implementación.
- H.2. Una calidad no aceptable requiere de un trabajo de comprobación, diseño e implementación superior al esperado.
- H.3. Utilizar lo último en informática alarga la planificación de forma impredecible.
- H.4. El desarrollo de funciones software erróneas requiere volver a diseñarlas y a implementarlas.
- H.5. El desarrollo de una interfaz de usuario inadecuada requiere volver a diseñarla y a implementarla.
- H.6. El desarrollo de funciones software innecesarias alarga la planificación.
- H.7. Alcanzar el ámbito del producto o las restricciones de velocidad requiere más tiempo del esperado, incluyendo el tiempo para volver a diseñar e implementar.
- H.8. Unos requisitos rígidos de compatibilidad con el sistema existente necesitan un trabajo extra de comprobación, diseño e implementación.
- H.9. Los requisitos para crear interfaces con otros sistemas, otros sistemas complejos, u otros sistemas que no están bajo el control del equipo de desarrollo suponen un diseño, implementación y prueba no previstos.
- H.10. El requisito de trabajar con varios sistemas operativos necesita más tiempo del esperado.
- H.11. El trabajo con un entorno software desconocido causa problemas no previstos.
- H.12. El trabajo con un entorno hardware desconocido causa problemas imprevistos.
- H.13. El desarrollo de un tipo de componente nuevo para la organización consume más tiempo del esperado.
- H.14. Depender de una tecnología que aún está en fase de desarrollo alarga la planificación.

I. Fuerzas mayores

- I.1. El producto depende de las normativas del gobierno, que pueden cambiar de forma inesperada.
- I.2. El producto depende de estándares técnicos provisionales, que pueden cambiar de forma inesperada.

J. Personal

- J.1. La contratación tarda más de lo esperado.
- J.2. Las tareas preliminares (por ejemplo, formación, finalización de otros proyectos, adquisición de licencias) no se han completado a tiempo.
- J.3. La falta de relaciones entre la dirección y el equipo de desarrollo ralentiza la toma de decisiones.
- J.4. Los miembros del equipo no se implican en el proyecto, y por lo tanto no alcanzan el nivel de rendimiento deseado.
- J.5. La falta de motivación y de moral reduce la productividad.
- J.6. La falta de la especialización necesaria aumenta los defectos y la necesidad de repetir el trabajo.
- J.7. El personal necesita un tiempo extra para acostumbrarse a trabajar con herramientas o entornos nuevos.
- J.8. El personal necesita un tiempo extra para acostumbrarse a trabajar con hardware nuevo.
- J.9. El personal necesita un tiempo extra para aprender un lenguaje de programación nuevo.
- J.10. El personal contratado abandona el proyecto antes de su finalización.
- J.11. Alguien de la plantilla abandona el proyecto antes de su finalización.
- J.12. La incorporación de nuevo personal de desarrollo al proyecto ya avanzado, y el aprendizaje y comunicaciones extra imprevistas reducen la eficiencia de los miembros del equipo existentes.
- J.13. Los miembros del equipo no trabajan bien juntos.
- J.14. Los conflictos entre los miembros del equipo conducen a problemas en la comunicación y en el diseño, errores en la interfaz y tener que repetir algunos trabajos.
- J.15. Los miembros problemáticos de un equipo no son apartados, influyendo negativamente en la motivación del resto del equipo.
- J.16. Las personas más apropiadas para trabajar en el proyecto no están disponibles.

- J.17. Las personas más apropiadas para trabajar en el proyecto están disponibles, pero no se pueden incorporar por razones políticas o de otro tipo.
- J.18. Se necesitan personas para el proyecto con habilidades muy específicas y no se encuentran.
- J.19. Las personas clave sólo están disponibles una parte del tiempo.
- J.20. No hay suficiente personal disponible para el proyecto.
- J.21. Las tareas asignadas al personal no se ajustan a sus posibilidades.
- J.22. El personal trabaja más lento de lo esperado.
- J.23. El sabotaje por parte de la dirección del proyecto deriva en una planificación ineficiente e inefectiva.
- J.24. El sabotaje por parte del personal técnico deriva en una pérdida de trabajo o en un trabajo de poca calidad, por lo que hay que repetir algunos trabajos.

K. Diseño e Implementación

- K.1. Un diseño demasiado sencillo no cubre las cuestiones principales, con lo que hay que volver a diseñar e implementar.
- K.2. Un diseño demasiado complejo exige tener en cuenta complicaciones innecesarias e improductivas en la implementación.
- K.3. Un mal diseño implica volver a diseñar e implementar.
- K.4. La utilización de metodologías desconocidas deriva en un periodo extra de formación y tener que volver atrás para corregir los errores iniciales cometidos en la metodología.
- K.5. El producto está implementado en un lenguaje de bajo nivel (por ejemplo, ensamblador) y la productividad es menor de la esperada.
- K.6. No se puede implementar la funcionalidad deseada con el lenguaje o bibliotecas utilizados: el personal de desarrollo tiene que utilizar otras bibliotecas, o crearlas él mismo para conseguir la funcionalidad deseada.
- K.7. Las bibliotecas de código o clases tienen poca calidad, y generan una comprobación extra, corrección de errores y la repetición de algunos trabajos.
- K.8. Se ha sobreestimado el ahorro en la planificación derivado del uso de herramientas para mejorar la productividad.
- K.9. Los componentes desarrollados por separado no se pueden integrar de forma sencilla, teniendo que volver a diseñar y repetir algunos trabajos.

L. Proceso

- L.1. La burocracia produce un progreso más lento del esperado.
- L.2. La falta de un seguimiento exacto del progreso hace que se desconozca que el proyecto esté retrasado hasta que está muy avanzado.
- L.3. Las actividades iniciales de control de calidad son recortadas, haciendo que se tenga que repetir el trabajo.
- L.4. Un control de calidad inadecuado hace que los problemas de calidad que afectan a la planificación se conozcan tarde.
- L.5. La falta de rigor (ignorar los fundamentos y estándares del desarrollo de software) conduce a fallos de comunicación, problemas de calidad y repetición del trabajo. Un consumo de tiempo innecesario.
- L.6. El exceso de rigor (aferramiento burocrático a las políticas y estándares de software) lleva a gastar más tiempo en gestión del necesario.
- L.7. La creación de informes de estado a nivel de directiva lleva más tiempo al desarrollador de lo esperado.
- L.8. La falta de entusiasmo en la gestión de riesgos impide detectar los riesgos más importantes del proyecto.
- L.9. La gestión de riesgos del proyecto software consume más tiempo del esperado.

Fuentes:

- Principles of Software Engineering Management (Gilb, 1998).
- Software Risk Management (Boehm, 1989).
- A Manager's Guide to Software Engineering (Pressman, 1993).
- Third Wave Project Management (Thomsett, 1993).
- Assessment and Control of Software Risks (Jones, 1994).

Análisis

CON-101/104, [T13]

Una vez identificados los riesgos, el siguiente paso es analizar cada riesgo para determinar su impacto.

Lo primero que medimos es la “*exposición a riesgos*”, definida como la probabilidad de ocurrencia del riesgo multiplicada por la magnitud de pérdida del riesgo (impacto). Por ejemplo: si existe un 25% de probabilidad de que ocurra un riesgo que retrasaría el proyecto en 4 semanas, entonces la exposición a este riesgo es de $0,25 \cdot 4 = 1$ semana.

Tanto la estimación de la probabilidad de ocurrencia del riesgo como de la magnitud de pérdida por la ocurrencia del riesgo se realizan de forma subjetiva. Sin embargo, existen varias técnicas que pueden ayudar a hacer acotar la subjetividad tanto como sea posible:

- Solicitar la estimación de la persona más familiarizada con el sistema, o con más experiencia en proyectos parecidos.
- Utilizar el método Delphi: varios miembros de un grupo identifican riesgos y les asignan una probabilidad de ocurrencia y una magnitud de pérdida. Estas estimaciones son discutidas por el grupo y refinadas.
- Usar adjetivos (*muy improbable, bastante improbable, improbable, probable, bastante probable, muy probable*) y asignar un valor numérico a cada uno.

PMI-134/135, [T14]

- Probabilidad: porcentaje de probabilidad de que ocurra un riesgo.
- Impacto: Efecto sobre los objetivos del proyecto en caso de ocurrir. Puede computarse en unidades de tiempo (retraso), unidades monetarias (incremento de costes), alcance (tamaño del producto en líneas de código, puntos función, etc.), ...

Una manera más completa de estimar el impacto de un riesgo es trabajar con matrices bidimensionales que indican el impacto del riesgo en cada dimensión de los objetivos de un proyecto (coste, tiempo, alcance, calidad, ...). Ver figura en [T14].

Priorización

CON-104/107, [T15]

Una vez identificados los riesgos, y realizada la estimación de la “*exposición a riesgos*”, el siguiente paso es priorizar los riesgos, de manera que se sepa dónde centrar el esfuerzo de la gestión de riesgos, como se muestra en la tabla de ejemplo siguiente.

No es preciso realizar una ordenación estricta de la lista de riesgos, ya que tal vez prefiramos priorizar algunos riesgos que producirían alguna pérdida muy grande, independientemente del lugar que ocupen en la tabla.

CON-105, [T16]

Ejemplo de tabla de estimación y priorización de riesgos

Riesgo	Probabilidad de pérdida	Magnitud de la pérdida (semanas)	Exposición a riesgo (semanas)
Añadir nuevas características desde marketing (sin conocer las características específicas)	35%	8	2.8
Planificación demasiado optimista	50%	5	2.5
Diseño inadecuado (hay que volver a diseñar)	15%	15	2.2

Riesgo	Probabilidad de pérdida	Magnitud de la pérdida (semanas)	Exposición a riesgo (semanas)
Las nuevas herramientas de programación no producen el ahorro prometido	30%	5	1.5
Añadir un requisito para la actualización automática desde el servidor	5%	20	1.0
Interfaz del subsistema de formato de gráficos inestable	25%	4	1.0
La aprobación del proyecto tarda más de lo esperado	25%	4	1.0
El personal contratado se retrasa en la entrega del subsistema encargado de formatear los gráficos	20%	4	0.8
Los recursos no están disponibles en su momento	10%	2	0.2
Los informes de estado a nivel de directiva necesitan más tiempo del previsto	10%	1	0.1

Planificación de Respuestas

PMI-143, [T17]

En esta actividad se construye un plan de contingencia para cada uno de los riesgos identificados en las actividades anteriores, que se han elegido como significativos para gestionarlos.

La colección de todos los planes de contingencia se suele agrupar en el llamado Plan de Respuestas a Riesgos. Este plan suele incluir:

- Riesgos identificados, sus descripciones, aspectos del proyecto afectados (elementos del alcance), sus causas, efectos en los objetivos del proyecto.
- Responsabilidades asignadas en cuanto a cada riesgo.
- Resultados del análisis de riesgos (probabilidad, impacto, exposición), incluyendo su priorización.
- Planes de contingencia - Respuestas previstas para aquellos riesgos que se han considerado prioritarios y que serán planificados.
- Nivel de riesgo residual esperado después de que aplique la respuesta prevista.
- Acciones específicas para implementar la estrategia de respuesta a cambios.
- Presupuesto y tiempos para las respuestas.

Supervisión y Control de Riesgos

Identificados, analizados y priorizados los riesgos del proyecto, estamos ya en disposición de someterlos a control.

El control de riesgos cubre dos actividades: resolución de riesgos y supervisión (monitorización).

Supervisión

PMI-144, [T18]

Consiste en realizar un seguimiento de los riesgos durante la ejecución del proyecto. Un buen método es realizar seguimientos periódicos (semanales, por ejemplo) de los riesgos más significativos.

Su finalidad es determinar si:

- Las respuestas son implementadas según se planificó.
- Las acciones de respuesta son tan efectivas como se esperaba o si nuevas respuestas deberían ser elaboradas.
- Las asunciones del proyecto son todavía válidas.
- La exposición a los riesgos ha cambiado y, en caso afirmativo, cual es la tendencia futura.
- Un disparador (evento) de riesgo ha ocurrido.
- Se han seguido las políticas y procedimientos adecuados.
- Han ocurrido riesgos nuevos que no estaban previamente identificados.

Resolución o control de riesgos

CON-107/108, PMI-142/143, [T19]

Qué hacer en este punto depende mucho del tipo de riesgo. Sin embargo, las acciones o estrategias que se pueden seguir son las siguientes:

- Evitar el riesgo: es decir, no realizar actividades arriesgadas cambiando el plan del proyecto.
- Trasladar el riesgo a un tercero (habitual en riesgos económicos) o a otra parte menos crítica del sistema.
- Conseguir información acerca del riesgo cuando éste no es muy conocido.
- Eliminar el origen del riesgo.
- Mitigar el riesgo. No se elimina el origen pero se cambia el plan para que su exposición sea menor.
- Comunicar el riesgo al resto del equipo, al cliente y a la dirección, para que estén prevenidos.
- Controlar/Aceptar el riesgo. Aceptar que puede ocurrir y hacer un plan de contingencias para minimizar su exposición.
- Recordar el riesgo para planes futuros.