

8 Riscos

Este capítulo discute um dos principais problemas que causa o insucesso de projetos de software, o *risco*, cujo estudo, muitas vezes é subestimado. Inicialmente, o capítulo apresenta o *plano de gerência de riscos* (Seção 8.1), que será construído a partir de uma *identificação de riscos* (Seção 8.2), incluindo um *checklist* (Seção 8.3) para sua efetiva identificação. Os riscos identificados passam por um processo de *análise* (Seção 8.4), que vai determinar o grau de importância ou exposição efetiva de cada risco. Em seguida são apresentados os *planos de mitigação de riscos* (Seção 8.5), que podem ser executados para reduzir a probabilidade ou impacto do risco antes que ele se torne um problema, e os *planos de contingência* (Seção 8.6), que devem ser executados quando, apesar de todos os esforços o risco efetivamente se tornar um problema. O capítulo termina com uma discussão sobre o *monitoramento* (Seção 8.7), *controle* (Seção 8.8) e a *comunicação* dos riscos (Seção 8.9).

Todo projeto de desenvolvimento de software apresenta um conjunto de incertezas em diferentes graus que podem causar problemas. Um planejador que não esteja atento aos riscos do projeto não terá planos para tratar situações que podem vir a prejudicar ou até inviabilizar todo um projeto.

Pode-se dizer genericamente que a falta de planejamento em relação aos riscos seja uma das maiores causas de fracasso em projetos na área de software. Afinal, o que faz o projeto falhar é o cronograma que não foi cumprido, os custos que extrapolaram o orçamento, a qualidade que ficou abaixo do esperado... tudo isso são riscos que se tornaram problemas.

Um gerente de projeto sem planos de tratamento de riscos não terá parâmetros para ação. É por falta de planejamento relacionado a riscos que muitos projetos de software fracassam ou não conseguem manter seu cronograma e orçamento dentro do previsto. A necessidade de planejar ações referentes aos riscos decorre do fato de que se isso não for feito, cronogramas e orçamentos dificilmente serão cumpridos.

Na grande maioria dos casos, um planejamento adequado pode fazer então que os riscos nunca se tornem problemas ou, caso isso aconteça, seu prejuízo seja minimizado.

O modelo de gerenciamento de riscos do SEI (Carr, Konda, Monarch, Ulrich, & Walker, 1993) envolve seis atividades conforme listado abaixo:

- a) *Identificação*. Antes que os riscos possam ser tratados, eles precisam ser identificados. A Seção 8.2 apresenta algumas técnicas de identificação de riscos.
- b) *Análise*. A atividade de análise transforma a lista de riscos potenciais em um documento mais útil para o planejador e o gerente de um projeto, pois, a partir da análise, os riscos são priorizados e, assim, o planejador e gerente poderão se concentrar nos riscos mais importantes, sem perder tempo com os insignificantes. A Seção 8.4 trata da análise de riscos.
- c) *Planejamento*. O planejamento frente aos riscos permite ao gerente prevenir problemas, usualmente de três formas: planejando e executando planos para reduzir a probabilidade de o risco ocorrer (Seção 8.5.1), planejando e executando planos para

reduzir o impacto do risco, caso ocorra (Seção 8.5.2), e planejando as atividades de recuperação de projeto, caso o risco efetivamente ocorra (Seção 8.6).

- d) *Rastreamento*. O rastreamento (ou monitoramento) de riscos consiste em avaliar, ao longo do projeto, as propriedades do risco (por exemplo, a probabilidade de ele ocorrer). O rastreamento deve ser baseado em métricas de avaliação de risco (Seção 8.7).
- e) *Controle*. Em função de mudanças no *status* de um risco, planos podem ter que ser executados ou, muitas vezes, é necessário improvisar a resposta ao problema causado pelo risco (Seção 8.8).
- f) *Comunicação*. A comunicação é um processo fundamental ao longo de todo um projeto de software, especialmente em relação à prevenção e tratamento de riscos. Assim, não há uma atividade específica de comunicação em riscos, mas uma prática que permeia todas as outras atividades (Seção 8.9).

8.1 Plano de Gerência de Riscos

O *Plano de Gerência de Riscos* inclui os vários elementos apresentados neste capítulo. Em resumo, ele deve mostrar:

- a) Quais são os riscos identificados (Seção 8.2 e Seção 8.3).
- b) Uma análise qualitativa ou quantitativa de cada risco, por exemplo, indicando a probabilidade de sua ocorrência e seu provável impacto sobre o projeto, caso ocorra (Seção 8.4).
- c) Como a probabilidade do risco ocorrer pode ser reduzida (Plano de Redução de Probabilidade, Seção 8.5.1).
- d) Como o impacto do risco, caso ocorra, pode ser reduzido (Plano de Redução de Impacto, Seção 8.5.2).
- e) O que fazer se o risco ocorrer (Plano de Resposta ao Risco, Seção 8.6).
- f) Como monitorar os riscos (Seção 8.7).

Os planos de redução de probabilidade e impacto também são chamados de planos de *mitigação* de risco, ou seja, são planos executados de forma preventiva para evitar que o risco ocorra e, se ainda assim ocorrer, seu prejuízo seja reduzido.

O planejamento de riscos conta com alguns fatores inibidores, o que faz com que muitas vezes a equipe não esteja preparada para lidar com imprevistos. Entre outras coisas, existe uma cultura da aversão ao risco. Mas, em projetos de software, todo o otimismo do mundo não vai afetar o andamento das coisas, especialmente se problemas ocorrerem ao longo do projeto. É necessário que se esteja preparado. Ninguém espera que ocorra um incêndio, mas mesmo assim os bombeiros estão sempre de prontidão.

Uma característica altamente desejada para um planejador e mesmo um gerente de projeto em relação ao risco é a capacidade de visão antecipada. Um bom planejador e um bom gerente são capazes de visualizar com antecedência possíveis situações anômalas que poderiam impedir o bom andamento do projeto. Essa previsão, longe de ser uma atitude pessimista pode ser uma atitude que poderá salvar o projeto no futuro ou pelo menos mantê-lo dentro do cronograma e orçamento previstos. Além disso, não são todos os riscos que

precisam preocupar o planejador de projeto. Como será visto adiante, apenas os riscos de maior importância merecerão uma atenção mais detalhada.

Outro princípio importante é a antecipação das atividades de maior risco, como preconizado nos modelos Espiral, UP e métodos ágeis. A ideia é que se um risco pode, provavelmente, inviabilizar um projeto, é melhor que isso seja analisado e descoberto o quanto antes, porque quanto mais tempo passar maior terá sido o investimento e, por conseguinte, o custo.

Os riscos podem ser classificados em três grupos em relação ao conhecimento que se tem deles:

- a) *Riscos conhecidos*. São aqueles já identificados e para os quais a equipe possivelmente está preparada.
- b) *Riscos desconhecidos*. São aqueles que, se as medidas de identificação adequadas tivessem sido tomadas poderiam ter sido descobertos, mas não foram.
- c) *Riscos impossíveis de prever*. São aqueles que, mesmo com as melhores técnicas de identificação não teriam sido identificados.

Então, o plano de gerência de riscos vai poder tratar apenas o primeiro tipo de riscos. Para minimizar o segundo grupo, boas atividades de identificação de riscos devem ser desenvolvidas. Já o terceiro grupo vai depender da capacidade do gerente de responder a situações totalmente imprevistas de forma organizada.

8.2 Identificação de Riscos

Um risco, normalmente compõe-se de dois elementos:

- a) Uma *causa*, na forma de uma condição incerta.
- b) Um *problema* que pode ocorrer em função da causa, o qual vai provocar um *efeito* ou impacto em um ou mais objetivos do projeto ou iteração.

Assim, um risco é um problema que pode ocorrer como resultado de uma causa. Se esse problema ocorrer então se estima que um dos objetivos do projeto será impactado, seja no tempo, custo, qualidade, ou outro aspecto. Por exemplo: “Como consequência do uso de um novo hardware (uma exigência definida), erros inesperados de integração do sistema podem ocorrer (um risco incerto), o que levaria a um estouro dos custos do projeto (um efeito sobre o objetivo do orçamento).” (Dinsmore, Cabanis-Brewin, Abdollahyan, Anselmo, Cota, & Cavalieri, 2009).

O PMBOK¹⁰⁵ (PMI, 2004), referência em gerenciamento de projetos, define que o risco é uma condição incerta que pode ter tanto um efeito positivo quanto negativo sobre o projeto. Assim, existiriam também os riscos positivos. Mas certamente os mais importantes a serem identificados são aqueles que podem prejudicar o projeto.

Assim como os requisitos de um projeto, os riscos devem ser identificados e priorizados para que sejam abordados adequadamente.

¹⁰⁵ www.pmi.org

O ideal é que o planejador já tenha a sua disposição um catálogo com riscos que já ocorreram no passado em projetos semelhantes. É importante que esse histórico nunca se perca. Mas caso não exista tal registro, uma identificação de riscos pode ser feita em uma reunião com a equipe e discussão sobre possíveis incertezas relacionadas aos tópicos acima. Sugestões para essa discussão são apresentadas nas subseções seguintes.

O Processo Unificado associa diferentes tipos de riscos com as diferentes fases do projeto:

- a) *Concepção*: riscos de requisitos e de negócio.
- b) *Elaboração*: riscos de tecnologia e arquitetura de sistema.
- c) *Construção*: riscos de programação e teste de sistema.
- d) *Transição*: riscos de utilização do sistema no ambiente final.

A literatura apresenta várias técnicas para identificação de riscos na fase de planejamento de um projeto. Entre outros, pode-se citar:

- a) Uso de *checklists* pré-definidos com possíveis riscos. Tais listas podem ser obtidas tanto na literatura ou na internet quanto a partir de projetos anteriores executados pela mesma equipe. Uma excelente base para iniciar uma lista deste tipo é o relatório SEI de taxonomia de riscos (Carr, Konda, Monarch, Ulrich, & Walker, 1993)¹⁰⁶ apresentado na Seção 8.3.
- b) Reuniões e *brainstormings* com gerente e equipe de projeto com experiência em outros projetos.
- c) Análise de cenários e lições aprendidas em projetos anteriores com contexto semelhante.

A identificação de riscos deve considerar diferentes fontes, como, por exemplo:

- a) Tecnologia (hardware e software).
- b) Pessoas (cliente, equipe, mercado, etc.).
- c) Projeto (atrasos, custos exagerados, etc.).

8.2.1 Riscos Tecnológicos

Riscos tecnológicos estão relacionados a todas as incertezas referentes a como a equipe será capaz de lidar com a tecnologia necessária para realizar o projeto. Quanto menos experiência nessas tecnologias, maiores serão os riscos.

Projetos que envolvam diferentes sistemas de software e hardware também frequentemente enfrentarão problemas de compatibilidade. Tornar tais sistemas compatíveis demandará tempo e custo extras ao projeto. Assim, o planejador deve saber se diferentes tecnologias terão de interagir e qual a experiência da equipe com esse tipo de integração.

Outro ponto que pode oferecer risco tecnológico a um projeto é a questão da obsolescência. Quão rápido as tecnologias usadas ou produzidas serão suplantadas por outras mais eficientes?

¹⁰⁶ www.sei.cmu.edu/reports/93tr006.pdf

8.2.2 Riscos Relacionados a Pessoas

Há vários tipos de interessados em um projeto. Cada um dos papéis pode produzir um risco característico, como, por exemplo:

- a) *Riscos de pessoal*. Projetos são executados por pessoas. Então perder uma pessoa da equipe de forma permanente ou temporária pode ter um impacto grande na medida em que essa pessoa for insubstituível.
- b) *Riscos de cliente*. Até que ponto o cliente se manterá interessado no projeto? Mudanças políticas ou administrativas poderão afetar o interesse da empresa em investir no projeto? Mesmo que a empresa mantenha interesse no projeto, o cliente estará disponível para esclarecer requisitos e realizar testes?
- c) *Riscos de negócio*. Muitas vezes a empresa até constrói um bom produto no prazo e custo definidos, mas mesmo assim o projeto fracassa. Entre outras coisas, a empresa poderá não ter a habilidade necessária para vender o produto, ou ainda, ter essa habilidade, mas o produto não ter efetivamente apelo comercial.
- d) *Riscos legais*. Existem problemas ou possibilidade de litígio? Uso de material protegido por direitos autorais? Necessidade de celebração de contrato com terceiros? Normas e leis específicas em outros estados ou países?

8.2.3 Riscos de Projeto

Os riscos de gerenciamento do projeto envolvem a capacidade da equipe de planejar e seguir o plano dentro do cronograma e custo previstos. Esse tipo de risco costuma se manifestar das seguintes formas:

- a) *Riscos de requisitos*. A equipe, por ser inexperiente, pode não ter sido capaz de identificar corretamente os requisitos do projeto, o que causará problemas no decorrer do mesmo. Requisitos poderão ser insuficientes, excessivos ou incorretos. Ou ainda, os requisitos podem ser naturalmente instáveis devido a características do próprio projeto.
- b) *Riscos de processo*. O modelo de processo escolhido é adequado às características do projeto? A equipe tem experiência com o processo? O gerente tem experiência em projetos anteriores?
- c) *Riscos de orçamento*. A verba necessária para o projeto está garantida até que ponto? Os custos foram corretamente previstos em projetos passados?
- d) *Riscos de cronograma*. É possível que prazos sejam alterados? É possível que a ordem em que as funcionalidades são entregues possa mudar? O planejador também deve saber em que grau a equipe mostrou-se capaz de ater-se ao cronograma em projetos passados.

A inexistência de qualquer uma dessas informações caracteriza um risco importante ao projeto na medida da sua incerteza.

8.3 Checklist de Riscos

O método de identificação de riscos do SEI (Carr, Konda, Monarch, Ulrich, & Walker, 1993) é baseado em uma taxonomia que contém termos relacionados ao processo de desenvolvimento de software. Para cada item desta taxonomia, pode ser elaborado um questionário, ou *checklist*, a partir do qual riscos podem ser identificados.

São definidas três grandes classes de risco:

- a) Engenharia do produto.
- b) Ambiente de desenvolvimento
- c) Restrições externas.

Cada uma destas categorias possui seus próprios elementos de risco, e cada elemento apresenta suas propriedades a partir das quais são formuladas perguntas que permitem analisar se o projeto corre ou não algum risco referente àquela propriedade.

Inicialmente apresenta-se o *checklist* referente à *engenharia do produto*, que tem como elementos de risco os requisitos, o *design*, a codificação e teste de unidade, a integração e teste e outros aspectos específicos de engenharia:

- a) Requisitos
 - a. Estabilidade: os requisitos podem mudar durante o desenvolvimento?
 - i. Os requisitos são estáveis?
 - 1. Se não, quais os efeitos disso no sistema? (qualidade / funcionalidade / cronograma / integração / *design* / teste)
 - ii. As interfaces externas do sistema estão mudando ou vão mudar?
 - b. Completeza: estão faltando requisitos ou estão especificados de forma incompleta?
 - i. Existem tópicos a serem esclarecidos nas especificações?
 - ii. Existem requisitos que se sabe que deveriam estar nas especificações mas não estão?
 - 1. Se sim, é possível obter esses requisitos e colocá-los na especificação?
 - iii. O cliente tem expectativas ou requisitos que não estão escritos?
 - 1. Se sim, há forma de capturá-los?
 - 2. As interfaces externas são completamente definidas?
 - c. Clareza: os requisitos estão obscuros ou necessitam interpretação?
 - i. Você é capaz de entender os requisitos da forma como estão escritos?
 - 1. Se não, há ambiguidades sendo resolvidas satisfatoriamente?
 - 2. Se sim, não há ambiguidades ou problemas de interpretação?
 - d. Validade: os requisitos vão levar ao produto que o cliente tem em mente?
 - i. Existem requisitos que podem não especificar exatamente o que o cliente quer?
 - 1. Se sim, como você está resolvendo isso?
 - ii. Você e o cliente compreendem a mesma coisa a partir dos requisitos?
 - 1. Se sim, existe um processo para determinar isso?
 - iii. Como você valida os requisitos junto ao cliente?
 - 1. Prototipação?
 - 2. Análise?
 - 3. Simulação?
 - e. Exequibilidade: os requisitos são exequíveis de um ponto de vista analítico?
 - i. Há requisitos que sejam tecnicamente difíceis de implementar?
 - 1. Se sim, quais são?
 - 2. Se sim, porque eles são difíceis de implementar?
 - 3. Se não, foram feitos estudos de exequibilidade sobre os requisitos?
 - a. Se sim, qual seu grau de confiança em tais estudos?

- f. Precedentes: os requisitos especificam algo que nunca foi feito antes, ou que a empresa nunca fez antes?
 - i. Existe algum requisito do estado da arte?
 - 1. Tecnologias?
 - 2. Métodos?
 - 3. Linguagens?
 - 4. Hardware?
 - 5. Se não, algum destes é novo para a equipe?
 - 6. Se sim, a equipe tem conhecimento suficiente nestas áreas?
 - a. Se não, há um plano para adquirir conhecimento nestas áreas?
 - g. Escala: os requisitos especificam um produto maior, mais complexo ou requerendo mais organização do que a empresa tem experiência?
 - i. O tamanho e complexidade do sistema são uma preocupação?
 - 1. Se não, você já fez algo deste tamanho e complexidade antes?
 - ii. O tamanho requer uma organização maior do que o usual para a empresa?
- b) *Design*.
- a. Funcionalidade: existem problemas potenciais para obter os requisitos funcionais?
 - i. Há algum algoritmo especificado que possa não satisfazer os requisitos?
 - 1. Se não, há algum algoritmo ou *design* que obtenha os requisitos de forma marginal?
 - ii. Como você determina a exequibilidade dos algoritmos e *design*?
 - 1. Prototipação?
 - 2. Modelagem?
 - 3. Análise?
 - 4. Simulação?
 - b. Dificuldade: o *design* ou implementação serão difíceis de serem realizados?
 - i. Alguma parte do *design* depende de hipóteses otimistas ou não realistas?
 - ii. Existe algum requisito que seja difícil de obter um *design*?
 - 1. Se não, você tem soluções para todos os requisitos?
 - 2. Se sim, quais são os requisitos e porque eles são difíceis?
 - c. Interfaces: as interfaces internas de hardware e software são bem definidas e controladas?
 - i. Há interfaces internas bem definidas?
 - 1. Software para software?
 - 2. Software para hardware?
 - ii. Há um processo para definir interfaces internas?
 - 1. Se sim, há um processo de controle de mudança para interfaces internas?
 - iii. Há hardware sendo desenvolvido em paralelo com o software?
 - 1. Se sim, as especificações do hardware estão mudando?
 - 2. Se sim, todas as interfaces com o software já foram definidas?
 - 3. Se sim, há modelos de *design* de engenharia que possam ser usados para testar o software?
 - d. Performance: existem tempos de resposta ou taxas de transferência rigorosos?
 - i. Existem problemas com performance?
 - 1. Taxa de transferência?
 - 2. Escalonamento de eventos de tempo real assíncronos?
 - 3. Respostas em tempo real?
 - 4. Tempo para recuperação de falhas (*recovery timeline*)?

5. Tempo de resposta?
6. Tempo de resposta, acesso e número simultâneo de usuários da base de dados?
- ii. Foi feita uma análise de performance?
 1. Se sim, qual o seu nível de confiança na análise feita?
 2. Se sim, você tem um modelo para rastrear a performance através do *design* e implementação?
- e. Testabilidade: o produto é difícil ou impossível de testar?
 - i. O software vai ser fácil de testar?
 - ii. O *design* inclui características que facilitam o teste?
 - iii. Os testadores se envolveram com a análise dos requisitos?
- f. Restrições de hardware: existem restrições apertadas no hardware alvo?
 - i. Arquitetura?
 - ii. Capacidade de memória?
 - iii. Taxa de transferência?
 - iv. Resposta em tempo real?
 - v. Tempo para recuperação de falhas?
 - vi. Performance da base de dados?
 - vii. Funcionalidade?
 - viii. Confiabilidade?
 - ix. Disponibilidade?
- g. Software não desenvolvido: há problemas com software usado mas não desenvolvido pela equipe?
 - i. Você está reusando ou fazendo reengenharia em software não desenvolvido pela equipe?
 1. Se sim, você prevê algum problema?
 - a. Documentação?
 - b. Performance?
 - c. Funcionalidade?
 - d. Prazo de entrega?
 - e. Personalização?
 - ii. Se estiver usando COTS, há algum problema com este tipo de software?
 1. Documentação insuficiente para determinar interfaces, tamanho ou performance?
 2. Performance fraca?
 3. Requer uma grande parcela de memória ou da base de dados?
 4. É difícil de interfacear com o software?
 5. Não foi testado sistematicamente?
 6. Não está livre de defeitos?
 7. Não foi adequadamente mantido?
 8. O vendedor demora para responder?
 - iii. Você prevê algum problema com a integração de atualizações ou revisões de COTS?
- c) Codificação e teste de unidade.
 - a. Exequibilidade: a implementação do *design* é difícil ou impossível?
 - i. Existem partes do produto não completamente especificadas no *design*?
 - ii. Os algoritmos e *design* são fáceis de implementar?
 - b. Teste: os níveis e tempos especificados para os testes de unidade são adequados?
 - i. Você começa o teste de unidade antes de verificar código com respeito ao *design*?

- ii. Testes de unidade suficientes são especificados?
 - iii. Há tempo suficiente para realizar todo o teste de unidade que você acha necessário?
 - iv. Serão feitos relaxamentos nos testes de unidades se houver problemas de cronograma?
- c. Codificação/Implementação: há problemas com codificação e implementação?
 - i. Os *designs* e especificações estão em um nível adequado para permitir a codificação?
 - ii. O *design* muda à medida que o código é escrito?
 - iii. Há restrições de sistema que tornam o código difícil de ser feito?
 - 1. Tempo?
 - 2. Memória?
 - 3. Armazenamento externo?
 - iv. A linguagem de programação é adequada para este projeto?
 - v. O projeto usa mais de uma linguagem?
 - 1. Se sim, há compatibilidade entre o código produzido pelos diferentes compiladores?
 - vi. O computador de desenvolvimento é o mesmo onde o sistema vai rodar?
 - 1. Se não, há diferenças relativas à compilação entre os dois computadores?
 - vii. Se estiver sendo desenvolvido hardware, suas especificações são suficientes para o desenvolvimento do software?
 - viii. As especificações do hardware mudam a medida que o software é codificado?
- d) Integração e teste
 - a. Ambiente: o ambiente de integração e teste é adequado?
 - i. Haverá hardware suficiente para uma adequada integração e teste?
 - ii. Há problemas para desenvolver cenários realísticos e testar dados para demonstrar algum dos requisitos?
 - 1. Tráfego de dados especificado?
 - 2. Resposta em tempo real?
 - 3. Tratamento de eventos assíncronos?
 - 4. Interação multiusuário?
 - iii. Você é capaz de verificar a performance nas suas instalações?
 - 1. Se sim, isso é suficiente para todos os testes?
 - b. Produção: a definição de interfaces e instalações são inadequadas ou o tempo insuficiente?
 - i. O hardware alvo estará disponível quando necessário?
 - ii. Critérios de aceitação foram acordados com o cliente para todos os requisitos?
 - 1. Se sim, há um acordo formal?
 - iii. As interfaces externas foram definidas, documentadas e transformadas em *baselines*?
 - iv. Há algum requisito que seja difícil de testar?
 - v. Foi especificada integração de produto suficiente?
 - vi. Foi alocado tempo suficiente para integração e teste do produto?
 - vii. Se usar COTS, os dados do vendedor serão aceitos na verificação dos requisitos alocados a COTS?
 - 1. Se sim, o contrato é claro neste ponto?

- c. Sistema: a integração de sistema é descoordenada, a definição de interface é pobre ou as instalações são inadequadas?
 - i. Integração de sistema suficiente foi especificada?
 - ii. Tempo suficiente para integração de sistema e testes foi previsto?
 - iii. O produto será integrado a um sistema existente?
 - 1. Se sim, haverá um período de transição definido?
 - a. Se não, como vai-se garantir que o programa funcionará corretamente depois de integrado?
 - iv. A integração do sistema vai ocorrer nas instalações do cliente?
- e) Aspectos específicos de engenharia.
 - a. Manutenibilidade: a implementação será difícil de entender e manter?
 - i. A arquitetura, design ou código criam dificuldades de manutenção?
 - ii. A equipe de manutenção foi envolvida cedo no processo?
 - iii. A documentação do produto é suficiente para que seja mantido por uma organização externa?
 - b. Confiabilidade: os requisitos de confiabilidade ou disponibilidade são difíceis de obter?
 - i. Existem requisitos de confiabilidade?
 - ii. Existem requisitos de disponibilidade?
 - 1. Se sim, o prazos de recuperação de falhas são um problema?
 - c. Riscos à segurança (*safety*): os requisitos relacionados a riscos à segurança são inexequíveis ou não demonstráveis?
 - i. Existem requisitos relacionados a riscos à segurança?
 - 1. Se sim, você vê alguma dificuldade em obtê-los?
 - ii. Será difícil verificar a satisfação destes requisitos?
 - d. Segurança do sistema (*security*): os requisitos de segurança do sistema são mais rigorosos do que o usual?
 - i. Existem requisitos de segurança no estado da arte ou sem precedentes?
 - ii. O sistema deve seguir regulamentos estritos de segurança estabelecidos por agência governamentais (como, por exemplo, *Orange Book*¹⁰⁷)?
 - iii. Você já implementou este nível de segurança antes?
 - e. Fatores humanos: o sistema será difícil de usar por conta de interface com usuário mal definida?
 - i. Você vê alguma dificuldade em satisfazer os requisitos de fatores humanos?
 - 1. Se não, como você garante que vai satisfazer estes requisitos?
 - ii. Se estiver usando prototipação, é um protótipo do tipo *throw-away*?
 - 1. Se não, está fazendo prototipação evolucionária?
 - a. Se sim, você tem experiência neste tipo de desenvolvimento?
 - b. Se sim, há versões preliminares entregáveis?
 - c. Se sim, isso complica o controle de mudança?
 - f. Especificações: a documentação é adequada para o *design*, implementação e teste do sistema?
 - i. A especificação dos requisitos do software é adequada para o *design* do sistema?
 - ii. A especificação dos requisitos de hardware é adequada para o *design* e implementação do sistema?
 - iii. As interfaces externas necessárias foram bem especificadas?

¹⁰⁷ www.hm-treasury.gov.uk/d/orange_book.pdf

- iv. As especificações de teste são adequadas para testar completamente o sistema?
- v. Se já estiver na fase de implementação ou após ela, as especificações de *design* são adequadas para implementar o sistema?
 - 1. Interfaces internas?

Outro conjunto de questionamentos diz respeito ao *ambiente de desenvolvimento*, que tem como elementos de risco o processo de desenvolvimento, o sistema de desenvolvimento, o processo de gerência, os métodos de gerência e o ambiente de trabalho. As questões relacionadas a testes elementos de risco são apresentadas abaixo:

- a) Processo de desenvolvimento.
 - a. Formalidade: a implementação será difícil de entender e manter?
 - i. Há mais de um modelo de desenvolvimento sendo usado?
 - 1. Espiral?
 - 2. Cascata?
 - 3. Incremental?
 - a. Se sim, a coordenação entre eles é um problema?
 - ii. Há planos formais e controlados para todas as atividades de desenvolvimento?
 - 1. Análise de requisitos?
 - 2. *Design*?
 - 3. Codificação?
 - 4. Integração e teste?
 - 5. Instalação?
 - 6. Garantia de qualidade?
 - 7. Gerenciamento de configuração?
 - a. Se sim, os planos especificam bem o processo?
 - b. Se sim, os desenvolvedores são familiarizados com os planos?
 - b. Adequação: o processo é adequado para o modelo de desenvolvimento?
 - i. O processo de desenvolvimento é adequado para este produto? (ver início do Capítulo 3)
 - ii. O processo de desenvolvimento é suportado por um conjunto de procedimentos, métodos e ferramentas compatíveis?
 - c. Controle de processo: o processo de desenvolvimento é executado, monitorado e controlado com métricas? Os locais de desenvolvimento distribuídos são coordenados?
 - i. Todos seguem o processo de desenvolvimento?
 - 1. Se sim, como isso é garantido?
 - ii. Você consegue mensurar se o processo de desenvolvimento está atingindo suas metas de qualidade e produtividade?
 - iii. Se há locais de desenvolvimento distribuídos, há coordenação adequada entre eles?
 - d. Familiaridade: os membros do projeto têm experiência no uso do processo? O processo é compreendido por toda a equipe?
 - i. As pessoas estão confortáveis com o processo de desenvolvimento?
 - e. Controle de produto: há mecanismos para controlar a mudança no produto?
 - i. Existe um mecanismo de controle de rastreabilidade de requisitos, que rastreia requisitos desde sua especificação até os casos de teste?

- ii. O mecanismo de rastreabilidade é usado na avaliação de impacto de mudanças de requisitos?
 - iii. Existe um mecanismo formal de controle de mudança?
 - 1. Se sim, ele cobre todas as mudanças nas *baselines* de requisitos, design, código e documentação?
 - iv. As mudanças em qualquer nível são mapeadas para cima até o nível de sistema e para baixo até o nível de teste?
 - v. Há análise adequada quando novos requisitos são adicionados ao sistema?
 - vi. Você tem algum meio de rastrear interfaces?
 - vii. Os planos e procedimentos de teste são atualizados como parte do processo de mudança?
- b) Sistema de desenvolvimento.
- a. Capacidade: existe poder de processamento, estações de trabalho, memória e espaço de armazenamento suficientes?
 - i. Há estações de trabalho e poder de processamento para toda a equipe?
 - ii. Há capacidade suficiente para fases que se entrelaçam como codificação, integração e teste?
 - b. Adequação: o sistema de desenvolvimento dá suporte a todas as fases, atividades e funções?
 - i. O sistema de desenvolvimento suporta todos os aspectos do projeto?
 - 1. Análise de requisitos?
 - 2. Análise de performance?
 - 3. *Design*?
 - 4. Codificação?
 - 5. Teste?
 - 6. Documentação?
 - 7. Gerenciamento de configuração?
 - 8. Requisitos e gerenciamento de rastreabilidade?
 - c. Usabilidade: qual a facilidade de uso do sistema de desenvolvimento?
 - i. As pessoas avaliam que o sistema de desenvolvimento é fácil de usar?
 - ii. Existe boa documentação sobre o sistema de desenvolvimento?
 - d. Familiaridade: existe pouca experiência anterior da empresa ou membros da equipe com o sistema de desenvolvimento?
 - i. As pessoas já usaram estas ferramentas e métodos antes?
 - e. Confiabilidade: o sistema de desenvolvimento sofre de erros, paralização, e capacidade de *backup* nativa insuficiente?
 - i. O sistema de desenvolvimento é considerado confiável?
 - 1. Compilador?
 - 2. Ferramentas de desenvolvimento?
 - 3. Hardware?
 - f. Suporte ao sistema: existe suporte ágil de vendedor ou especialista no sistema de desenvolvimento?
 - i. A equipe foi treinada no uso das ferramentas?
 - ii. Você tem acesso a especialistas no seu uso?
 - iii. Os vendedores respondem rapidamente aos problemas?
 - g. Capacidade de entrega (*deliverability*): os requisitos de definição e aceitação para a entrega do sistema de desenvolvimento para o cliente não foram incluídos no orçamento?
 - i. Você vai entregar o sistema e desenvolvimento ao cliente?

1. Se sim, recursos, cronograma e orçamento foram alocados para esta entrega?
- c) Processo de gerência.
- a. Planejamento: o planejamento é oportuno, líderes técnicos foram incluídos e planos de contingência realizados?
 - i. O desenvolvimento é gerenciado de acordo com o plano?
 1. Se sim, as pessoas apagam incêndios com frequência?
 - ii. É feito replanejamento quando ocorrem imprevistos no plano?
 - iii. As pessoas de todos os níveis estão incluídas no planejamento de seu próprio trabalho?
 - iv. Existem planos de contingência para riscos conhecidos?
 1. Se sim, como você determina a hora de ativar estes planos?
 - v. Questões de longo prazo foram adequadamente tratadas?
 - b. Organização de projeto: os papéis e relacionamentos estão claros?
 - i. A organização do desenvolvimento é efetiva?
 - ii. As pessoas compreendem os seus papéis e os papéis dos outros?
 - iii. As pessoas sabem quem tem autoridade para quê?
 - c. Experiência em gerência: os gerentes têm experiência em desenvolvimento de software, gerenciamento de projeto de software, no domínio de aplicação, no processo de desenvolvimento ou em grandes projetos?
 - i. O projeto tem gerentes experientes?
 1. Gerenciamento de projeto de software?
 2. Com desenvolvimento de software?
 3. Com o processo de desenvolvimento em uso?
 4. Com o domínio da aplicação?
 5. Com projetos deste tamanho ou complexidade?
 - d. Interfaces da equipe de desenvolvimento: existem interfaces fracas com o cliente, outros contratados ou gerentes de nível superior?
 - i. A gerência comunica problemas para cima e para baixo na linha hierárquica?
 - ii. Os conflitos com o cliente são documentados e resolvidos no tempo apropriado?
 - iii. A gerência envolve as pessoas apropriadas nos encontros com o cliente?
 1. Líderes técnicos?
 2. Desenvolvedores?
 3. Analistas?
 - iv. A gerência trabalha de forma que todas as facções do lado do cliente estejam representadas nas decisões sobre funcionalidade e operação?
 - v. É uma boa política apresentar uma visão otimista ao cliente ou gerentes de nível superior?
- d) Métodos de gerência.
- a. Monitoramento: as métricas de gerência são definidas e o progresso do desenvolvimento rastreado?
 - i. Existem relatórios de *status* estruturados periódicos?
 1. Se sim, as pessoas obtêm respostas aos seus relatórios de *status*?
 - ii. A informação apropriada é reportada aos níveis organizacionais corretos?
 - iii. O progresso é comparado com o plano?
 1. Se sim, a gerência tem uma visão clara sobre o que está acontecendo?
 - b. Gerenciamento de pessoal: o pessoal do projeto é treinado e usado adequadamente?
 - i. O pessoal é treinado nas habilidades requeridas pelo projeto?

1. Se sim, isso é parte do plano do projeto?
- ii. Há pessoas alocadas ao projeto que não possuem o perfil necessário?
- iii. É fácil para os membros do projeto conseguir ação da gerência?
- iv. Os membros em todos os níveis estão conscientes de seu status *versus* plano?
- v. As pessoas sentem que é importante manter o plano?
- vi. A gerência consulta as pessoas antes de tomar decisões que afetam o trabalho delas?
- c. Garantia de qualidade: existem procedimentos e recursos adequados para garantir a qualidade do produto?
 - i. A função de garantia de qualidade tem pessoal adequado para o projeto?
 - ii. Existem mecanismos adequados para garantir a qualidade?
 1. Se sim, todas as áreas e fases possuem procedimentos de qualidade?
 2. Se sim, as pessoas estão acostumadas a trabalhar com estes procedimentos?
- d. Gerenciamento de configuração: os procedimentos de gerenciamento de configuração, mudança e instalação são adequados?
 - i. Há um sistema gerenciador de configuração adequado?
 - ii. A função de gerenciamento de configuração tem pessoal adequado?
 - iii. É necessário coordenação com algum sistema instalado?
 1. Se sim, há gerenciamento de configuração adequado para o sistema instalado?
 2. Se sim, o sistema de gerenciamento de configuração sincroniza o seu trabalho com as mudanças no local?
 - iv. O sistema vai ser instalado em múltiplos locais?
 1. Se sim, o sistema de gerenciamento de configuração permite múltiplos locais?
- e) Ambiente de trabalho
 - a. Atitude em relação a qualidade: há falta de orientação relacionada ao trabalho com qualidade?
 - i. Todos os níveis de pessoal estão orientados para procedimentos de qualidade?
 - ii. O cronograma costuma ficar no caminho da qualidade?
 - b. Cooperação: falta espírito de equipe? Os conflitos resultantes requerem intervenção da gerência?
 - i. As pessoas trabalham cooperativamente através das fronteiras funcionais?
 - ii. As pessoas trabalham cooperativamente na direção de metas comuns?
 - iii. A intervenção da gerência às vezes é necessária para fazer as pessoas trabalharem juntas?
 - c. Comunicação: há pouca consciência a respeito de metas ou missão e comunicação pobre de informação técnica entre membros da equipe e gerentes?
 - i. Há boa comunicação entre os membros do projeto?
 1. Gerentes?
 2. Líderes técnicos?
 3. Desenvolvedores?
 4. Testadores?
 5. Gerente de configuração?
 6. Equipe de qualidade?
 - ii. Os gerentes são receptivos às comunicações da equipe?
 1. Se sim, você se sente a vontade para pedir ajuda ao seu gerente?

2. Os membros são capazes de comunicar riscos mesmo sem ter uma solução pronta?
- d. Moral: há uma atmosfera não produtiva ou não criativa? As pessoas sentem que não há reconhecimento ou recompensa por trabalho feito acima das expectativas?
 - i. Como está a moral no projeto?
 1. Se não estiver bem, qual é o principal motivo para a baixa moral?
 - ii. Há problemas para manter as pessoas que são necessárias?

Finalmente, são apresentados questionamentos referentes às *restrições externas* ao projeto, os quais incluem os seguintes elementos: recursos, contrato e interfaces de comunicação externas. Os questionamentos são, então, os seguintes:

- a) Recursos.
 - a. Cronograma: o cronograma é inadequado ou instável?
 - i. O cronograma tem sido estável?
 - ii. O cronograma é realístico?
 1. Se sim, o método de estimação é baseado em dados históricos?
 2. Se sim, o método de estimação funcionou bem no passado?
 - iii. Existe alguma coisa para a qual um cronograma realístico não foi preparado?
 1. Análises e estudos?
 2. Garantia de qualidade?
 3. Treinamento?
 4. Cursos e treinamento em manutenção?
 5. Equipamentos?
 6. Sistema de desenvolvimento que vai ser entregue?
 - iv. Há dependências externas que poderiam impactar o cronograma?
 - b. Pessoal: a equipe é inexperiente, sem conhecimento de domínio, sem habilidades ou em número insuficiente?
 - i. Há alguma área em que esteja faltando pessoal qualificado?
 1. Engenharia de software e métodos de análise de requisitos?
 2. Especialistas em algoritmos?
 3. *Design* e métodos de *design*?
 4. Linguagens de programação?
 5. Métodos de integração e teste?
 6. Confiabilidade?
 7. Manutenibilidade?
 8. Disponibilidade?
 9. Fatores humanos?
 10. Gerenciamento de configuração?
 11. Garantia de qualidade?
 12. Ambiente alvo?
 13. Nível de segurança?
 14. COTS?
 15. Reuso de software?
 16. Sistema operacional?
 17. Banco de dados?
 18. Domínio de aplicação?
 19. Análise de performance?
 20. Aplicações de tempo crítico?
 - ii. Há pessoal adequado para formar a equipe de projeto?

- iii. A equipe é estável?
 - iv. Você tem acesso às pessoas certas quando precisa delas?
 - v. Os membros da equipe já implementaram sistemas deste tipo?
 - vi. O projeto depende de algumas poucas pessoas?
 - vii. Há problemas para obter pessoal?
 - c. Orçamento: o orçamento é insuficiente ou instável?
 - i. O orçamento é estável?
 - ii. O orçamento é baseado em uma estimativa realística?
 - 1. Se sim, o método de estimativa é baseado em dados históricos?
 - 2. Se sim, o método funcionou bem no passado?
 - iii. Funcionalidades ou características do sistema já foram removidas para reduzir custo?
 - iv. Há alguma coisa para a qual um orçamento adequado não foi alocado?
 - 1. Análises e estudos?
 - 2. Garantia de qualidade?
 - 3. Treinamento?
 - 4. Cursos de manutenção?
 - 5. Equipamento?
 - 6. Sistema de desenvolvimento a ser entregue?
 - v. Mudanças no orçamento acompanham mudanças nos requisitos?
 - 1. Se sim, esta é uma parte padrão do processo de controle de mudança?
 - d. Instalações: as instalações são adequadas para construir e entregar o produto?
 - i. As instalações de desenvolvimento são adequadas?
 - ii. O ambiente de integração é adequado?
- b) Contrato.
- a. Tipo de contrato: o tipo de contrato é uma fonte de risco para o projeto?
 - i. Que tipo de contrato foi feito? (custo mais prêmio, preço fixo, ...)
 - 1. Isto apresenta algum problema?
 - ii. O contrato representa uma carga para algum aspecto do projeto?
 - 1. Declarações de trabalho?
 - 2. Especificações?
 - 3. Descrições de itens de dados?
 - 4. Partes?
 - 5. Envolvimento excessivo de cliente?
 - iii. A documentação requerida é uma carga?
 - 1. Quantidade excessiva?
 - 2. Cliente detalhista e exigente?
 - 3. Longo ciclo de aprovações?
 - b. Restrições: o contrato causa alguma restrição?
 - i. Há problemas com direitos de dados?
 - c. Dependências: o projeto tem dependência de algum produto ou serviço externo que podem afetar o produto?
 - i. Contratados associados?
 - ii. Contratado principal? (se você for um subcontratado)
 - iii. Subcontratados?
 - iv. Vendedores ou fornecedores?
 - v. Equipamento ou software desenvolvido pelo cliente?
- c) Interfaces de Comunicação Externas

- a. Cliente: há problemas com o cliente como ciclo longo de aprovação de documentos, comunicação fraca ou conhecimento inadequado do domínio?
 - i. O ciclo de aprovação do cliente é adequadamente rápido?
 - 1. Documentação?
 - 2. Revisões de projeto?
 - 3. Revisões formais?
 - ii. Você sempre segue em frente antes de receber aprovação do cliente?
 - iii. O cliente entende os aspectos técnicos do sistema?
 - iv. O cliente entende software?
 - v. O cliente interfere com processos ou pessoas?
 - vi. O gerente trabalha com o cliente para alcançar entendimentos e decisões mútuas em tempo aceitável?
 - 1. Entendimento sobre requisitos?
 - 2. Critérios de teste?
 - 3. Ajustes de cronograma?
 - 4. Interfaces?
 - vii. Quão efetivos são seus mecanismos para chegar a um acordo com o cliente?
 - 1. Grupos de trabalho (contratuais)?
 - 2. Encontros de intercâmbio técnico (contratuais)?
 - viii. Todas as facções do cliente são envolvidas para se chegar a acordos?
 - 1. Se sim, há um processo formalmente definido?
 - ix. A gerência apresenta um quadro realista ou otimista ao cliente?
- b. Contratados associados (se houver): há problemas com outras empresas que desenvolvem de forma associada para o mesmo cliente como interfaces fracas ou inadequadamente definidas, comunicação pobre ou falta de colaboração?
 - i. As interfaces externas mudam sem notificação adequada, coordenação ou procedimentos formais de mudança?
 - ii. Há um plano de transição adequado?
 - 1. Se sim, ele é suportado por todos os contratados e pessoal no local?
 - iii. Existem problemas em conseguir cronogramas ou dados de interface de contratados associados?
 - 1. Se não, eles são acurados?
- c. Subcontratados (se houver): o projeto depende de subcontratados em alguma área crítica?
 - i. Há alguma ambiguidade nas definições de tarefas de subcontratados?
 - ii. Os procedimentos de monitoramento e relatório dos subcontratados são diferentes dos usados no projeto?
 - iii. A administração dos subcontratados e seu gerenciamento técnico é feito por uma organização diferente?
 - iv. Você é altamente dependente de conhecimento especializado de subcontratados em alguma área?
 - v. O conhecimento dos subcontratados está sendo transferido para a empresa?
 - vi. Há problemas para obter cronogramas ou dados de interface com subcontratados?
- d. Contratado principal (se o projeto é um subcontrato): o projeto enfrenta dificuldades com o contratador principal?
 - i. As suas definições de tarefas do contratado principal são ambíguas?
 - ii. Você interfaceia com duas organizações diferentes do contratado principal em relação a administração e gerenciamento técnico?

- iii. Você é altamente dependente de conhecimento especializado do contratado principal?
 - iv. Há problemas em obter cronograma ou dados de interface do contratado principal?
- e. Gerenciamento corporativo: existe falta de suporte ou microgerenciamento por parte da gerência superior?
 - i. O gerente de projeto comunica problemas à gerência superior?
 - 1. Se sim, isso parece ser efetivo?
 - ii. A gerência corporativa dá apoio oportuno à resolução de problemas?
 - iii. A gerência corporativa tem tendência a microgerenciar?
 - iv. A gerência apresenta uma visão realista ou otimista para a gerência superior?
- f. Vendedores: os vendedores atendem rapidamente as necessidades dos projetos?
 - i. Você confia nos vendedores para entregas de componentes críticos?
 - 1. Compiladores
 - 2. Hardware
 - 3. COTS
- g. Políticas: as políticas causam problemas ao projeto?
 - i. As políticas afetam o projeto?
 - 1. Empresa?
 - 2. Cliente?
 - 3. Contratados associados?
 - 4. Subcontratados?
 - ii. As políticas afetam decisões técnicas?

Conforme comentado, esta lista é apenas uma base ou referência para a criação de uma *checklist* específica para uma empresa ou projeto. A partir dela, pode-se incorporar novos riscos identificados e, assim, construir um patrimônio de riscos identificados de forma que projetos futuros tenham desde o início uma base de conhecimento para que possam ser melhor planejados.

8.4 Análise de Riscos

Uma vez identificados riscos potenciais a análise dos riscos vai determinar quais são verdadeiramente relevantes para que se deva gastar tempo e dinheiro com sua prevenção.

Via de regra a análise de riscos vai tentar determinar a probabilidade de ocorrência e o impacto de cada risco potencial.

Algumas propriedades de riscos, entre outras, podem ser assim identificadas:

- a) *Probabilidade*. É a chance de que o risco realmente se torne um problema. Caso se disponha de séries históricas, a probabilidade pode ser medida numericamente. Por exemplo, se 35% dos projetos anteriores sofreram atraso no cronograma então se pode esperar que esse risco tenha uma probabilidade de 35% em projetos futuros. Porém, na maioria das vezes poderá ser mais factível considerar apenas valores discretos como probabilidade alta (quase certo que vá ocorrer), média (tem alguma chance de ocorrer) e baixa (pouco provável que ocorra), para um risco.
- b) *Impacto*. O impacto é a medida do prejuízo que um risco pode trazer ao projeto. O impacto é independente da probabilidade, pode-se ter riscos de alto impacto com alta ou baixa probabilidade e riscos de baixo impacto com alta ou baixa probabilidade.

Algumas vezes o impacto pode ser mensurado, por exemplo, quando se sabe o valor da multa por dia de atraso na entrega do produto, mas normalmente será mais prático avaliar o impacto como alto (pode inviabilizar o projeto), médio (pode aumentar significativamente o custo do projeto), ou baixo (apenas provoca algum contratempo, mas sem inviabilizar os objetivos do projeto).

- c) *Proximidade*. Alguns riscos podem ser de alta probabilidade, mas baixa proximidade, ou seja, podem ocorrer só um futuro distante. Por exemplo, a obsolescência tecnológica é um risco de alto impacto e alta probabilidade, mas possivelmente de baixa ou média proximidade, dependendo da tecnologia. Quando a proximidade é considerada como propriedade dos riscos convém também usá-la para o cálculo da importância do risco.
- d) *Acoplagem*. A acoplagem define o quanto um risco pode afetar outros riscos. Por exemplo, se os requisitos estiverem mal estabelecidos, outros riscos de projeto podem ter sua probabilidade ou impacto aumentados.

O produto da probabilidade pelo impacto consiste na *importância* do risco (ou *exposição*). Assim, riscos de maior importância (alta probabilidade e alto impacto) precisarão ter uma abordagem detalhada no projeto para que sejam tratados. Já os riscos de baixa importância não necessitam tanto investimento. Eventualmente, será suficiente apenas manter o gerente ciente de sua existência para tomar providências, caso a importância do risco se altere.

Deve-se lembrar que apesar dessa identificação das propriedades de riscos no momento do planejamento, o gerente de projeto deve manter em mente durante a execução do mesmo que todas as propriedades podem mudar com o tempo. Assim, o monitoramento dos riscos também é uma atividade que deve ser realizada com frequência pelo gerente de projeto.

Quando a probabilidade e impacto do risco são avaliados qualitativamente, pode-se usar uma tabela como a Tabela 8-1, onde a partir da combinação da probabilidade pelo impacto chega-se a um valor para a importância do risco.

Tabela 8-1: Forma de cálculo para a importância de um risco.

		Probabilidade		
		Alta	Média	Baixa
Impacto	Alto	Alta importância	Alta importância	Média importância
	Médio	Alta importância	Média importância	Baixa importância
	Baixo	Média importância	Baixa importância	Baixa importância

Normalmente os atributos de riscos são definidos em função da experiência e conhecimento de especialistas. Mas se houver um registro de projetos anteriores pode-se efetivamente ter uma expectativa mais concreta sobre a ocorrência de alguns riscos. Assim, muitas vezes o planejador de projeto poderá também adicionar à sua análise de riscos um conjunto de dados ou observações que justifiquem sua escolha para os valores de determinadas propriedades dos riscos.

8.5 Planos de Mitigação de Riscos

Planos de mitigação de riscos são executados antes que o risco ocorra. Para os riscos de alta importância, os planos de mitigação são definidos ainda na fase de planejamento do projeto. Para riscos de média importância, os planos são definidos e guardados para serem aplicados caso a importância do risco aumente ao longo do projeto.

Há dois tipos de planos de mitigação: plano de redução de probabilidade e plano de redução de impacto.

Para exemplificar a apresentação de planos de mitigação será usado um conjunto de riscos identificados e analisados, conforme a Tabela 8-2. Os riscos cujo código é prefixado com “t” são riscos tecnológicos. Os riscos prefixados com “pe” são riscos de pessoal. Os riscos prefixados com “pr” são riscos de projeto e os riscos prefixados com “l” são riscos legais.

O projeto trata da produção de uma ferramenta CASE em tecnologia *touch-screen* com comandos de voz e interface em realidade virtual. O projeto seria desenvolvido como projeto de pesquisa por alunos bolsistas de uma universidade. Os riscos foram identificados e avaliados por uma turma de alunos da disciplina de Análise e Projeto de Sistemas II, na UFSC, em 2011. Os riscos foram avaliados em função de sua probabilidade (P) e impacto (I), e a exposição (E), ou importância, resultante é calculada de acordo com a Tabela 8-1, com os valores: baixo (B), médio (M) e alto (A). Os riscos aparecem na tabela ordenados de acordo com sua importância.

Tabela 8-2: Identificação e análise de riscos de um projeto fictício.

Id	Causa	Risco	Efeito	P	I	E
pr3	Requisitos ainda muito instáveis	Pode haver mudanças importantes nos requisitos ao longo do desenvolvimento	Perda de tempo desenvolvendo partes que depois não serão usadas e atrasos no cronograma.	A	A	A
pr2	O tempo de desenvolvimento pode ser alto	Pode haver concorrentes que lancem produtos antes	Chegar ao mercado depois da janela de oportunidade	A	A	A
t8	Necessidade de muitos comandos baseados em gestos	Gestos muito parecidos podem significar comandos diferentes	O sistema pode interpretar erroneamente os comandos (desenhos, formas). Usuário pode ter que decorar muitos comandos diferentes.	A	M	A
pe1	Ainda não se sabe se será possível contratar equipe com experiência nas tecnologias	Necessidade de treinamento	Atrasos de cronograma e custos com treinamento	A	M	A
t4	O processo implementado pela ferramenta pode não atender aos desejos do usuário	O usuário não vai escolher a ferramenta porque usa um processo de desenvolvimento diferente	Problemas relacionados à venda. Pode haver necessidade de implementar vários processos, o que vai contra a filosofia inicial da ferramenta. Grandes empresas já têm processo estabelecido e teriam que mudar	M	A	A
t6	A tecnologia de comando de voz ainda não é bem desenvolvida.	Comandos de voz podem não ser corretamente entendidos.	Usuários frustrados	A	B	M
t9	Não existem ferramentas CASE com gráficos 3d ou em níveis de profundidade	Não existe um padrão ou referência para tais interfaces nem estudos de usabilidade	Necessidade de pesquisar padrões de usabilidade para interfaces 3d em ferramentas CASE	A	B	M
t3	Não é conhecido um padrão de usabilidade para CASE em <i>touchscreen</i>	Poderá ser desenvolvida uma ferramenta com usabilidade falha	Problemas com usuário final (desinteresse)	M	M	M
pe1	O projeto será desenvolvido com bolsistas	Bolsistas não vêem o projeto como carreira	Pode-se perder desenvolvedores ao longo do projeto, necessitando substituição	M	M	M

l1	Uso de tecnologia de terceiros	Pagamento de direitos autorais	Aumento de custo	M	M	M
t1	Superfície de toque é tecnologia nova	Podem ocorrer mudanças nos padrões. Qual o melhor sistema operacional?	Produto obsoleto ou necessidade de desenvolver para vários sistemas operacionais.	B	M	B
t2	Tecnologia nova	Podem não existir bibliotecas suficientemente adequadas para o desenvolvimento	Necessidade de desenvolver novas bibliotecas básicas	B	B	B
t5	O acesso aos recursos avançados será secundário na interface, que prioriza as ações mais elementares	Pode gerar problemas de usabilidade para usuários mais avançados	Gerar desinteresse por usuários avançados. É necessária uma boa análise de caso de uso e usabilidade na ferramenta durante seu desenvolvimento.	B	B	B
t7	O código gerado pela ferramenta, por <i>default</i> não será modificável	O código pode não ser o mais eficiente possível.	Sistemas gerados pela ferramenta podem ser ineficientes.	B	B	B

Os riscos de importância (exposição) alta tiveram planos de mitigação elaborados de forma sumária e executados. Os riscos de importância média deveriam ter seus planos apenas elaborados e guardados caso a importância do risco aumentasse. Os riscos de baixa importância deveriam apenas continuar a ser monitorados e, caso sua importância aumentasse para média, deveriam ter seus planos elaborados (e executados, caso a importância passasse imediatamente para alta).

8.5.1 Plano de Redução de Probabilidade de Risco

O plano de redução de probabilidade consiste nas ações identificadas como necessárias para diminuir a probabilidade de que um risco ocorra. Este tipo de plano deve agir nas *causas* do risco, ou seja, na segunda coluna da Tabela 8-2. Por exemplo, se é bem provável que a equipe tenha dificuldade com uma nova tecnologia a ser usada, pode-se realizar cursos de treinamento para diminuir a probabilidade de isso se tornar um estorvo ao andamento do projeto.

Riscos classificados com alta importância devem ter planos de redução de mitigação e contingência com atividades bem definidas, prazos e responsáveis pela sua execução. É importante que tais atividades sejam incluídas no plano do projeto ou iteração e que suas ações sejam contabilizadas no tempo e no custo do projeto.

É interessante lembrar que alguns ciclos de vida como Espiral, Cascata com Redução de Risco, métodos ágeis e o UP já prevêm que atividades de estudo e mitigação de riscos sejam previstas e incorporadas ao projeto.

No caso de riscos de média importância, pode ser recomendado apenas elaborar um plano de ação genérico para ser aplicado caso necessário.

Riscos de baixa importância podem apenas ser monitorados para verificar se a probabilidade ou o impacto aumentam ao longo do projeto e, neste caso, o plano de redução de probabilidade pode ser criado como resposta a essa mudança de estado.

Considerando o exemplo da Tabela 8-2, os planos de redução de probabilidade poderiam ser elaborados como mostrado na Tabela 8-3.

Tabela 8-3: Planos de redução de probabilidade.

Id	Causa do Risco	Risco	Plano de redução de probabilidade
pr3	Requisitos ainda muito instáveis	Pode haver mudanças importantes nos requisitos ao longo do desenvolvimento	Realizar reuniões de eliciação de requisitos. Inspeccionar requisitos. Procurar produtos semelhantes na internet e analisá-los. Planejar desenvolvimento de protótipos.
pr2	O tempo de desenvolvimento pode ser alto	Pode haver concorrentes que lancem produtos antes	Planejar desenvolvimento orientado a cronograma com entregas de versões parciais usáveis em intervalos de 6 meses.
t8	Necessidade de muitos comandos baseados em gestos	Gestos muito parecidos podem significar comandos diferentes	Pesquisar padrões existentes para comandos baseados em gestos e cataloga-los. Definir hierarquia de comandos e comandos baseados em contextos para reduzir a quantidade de comandos necessários.
pe1	Ainda nao se sabe se será possível contratar equipe com experiencia nas tecnologias	Necessidade de treinamento	Publicar anúncio solicitando currículos para pessoas com as habilidades desejadas.
t4	O processo implementado pela ferramenta pode nao atender aos desejos do usuário	O usuário não vai escolher a ferramenta porque usa um processo de desenvolvimento diferente	Pesquisar qual o processo mais usado no mercado alvo. Adaptar a ferramenta para uso com o(s) processo(s) dominantes.
t6	A tecnologia de comando de voz ainda não é bem desenvolvida.	Comandos de voz podem nao ser corretamente entendidos.	Pesquisar aplicativos operados por tecnologia de voz e testá-los. Verificar existência de módulos reusáveis de comando por voz.
t9	Não existem ferramentas CASE com graficos 3d ou em níveis de profundidade	Nao existe um padrão ou referencia para tais interfaces nem estudos de usabilidade	Catalogar e estudar ferramentas semelhantes com interfaces 3d ou em níveis.
t3	Não é conhecido um padrão de usabilidade para CASE em <i>touchscreen</i>	Poderá ser desenvolvida uma ferramenta com usabilidade falha	Catalogar e estudar ferramentas semelhantes já desenvolvidas para superfícies de toque. Estudar normas de usabilidade em geral e normas específicas para ferramentas CASE e para sistemas baseados em superfície de toque.
pe1	O projeto será desenvolvido com bolsistas	Bolsistas nao vêem o projeto como carreira	Verificar valor de salário de mercado. Verificar possibilidade de oferecer salários mais atraentes. Verificar possibilidade de subcontratar desenvolvimento.
l1	Uso de tecnologia de terceiros	Pagamento de direitos autorais	Verificar valores e condições de uso de potenciais tecnologias. Verificar existência de soluções livres.

8.5.2 Plano de Redução de Impacto de Risco

O plano de redução de impacto do risco é definido e aplicado de forma semelhante ao plano de redução de probabilidade, exceto pelo fato de que neste caso, as ações devem procurar diminuir o impacto do risco. Assim, estes planos vão procurar diminuir os *efeitos* do risco, e não suas causas.

Por exemplo, se um determinado membro da equipe é o único a dominar a tecnologia a ser usada no projeto, sua saída pode causar grande impacto ao andamento do projeto. Assim, um plano de redução de impacto poderia consistir em treinar outro funcionário nessa tecnologia ou colocá-lo a trabalhar em conjunto com o funcionário crítico para que possa aprender o máximo possível sobre a tecnologia para no caso da falta do outro o prejuízo ao projeto não seja tão grande.

Novamente, os riscos de alta importância determinam a existência de um plano detalhado de redução de impacto com atividades previstas, prazos e responsáveis. Riscos de média importância poderão receber apenas uma lista de ações a serem efetuadas em caso de necessidade e os riscos de baixa importância serão apenas monitorados.

Considerando o exemplo da Tabela 8-2, os planos de redução de probabilidade poderiam ser elaborados como mostrado na Tabela 8-4.

Tabela 8-4: Planos de redução de impacto.

Id	Risco	Efeito	Plano de redução de impacto
pr3	Pode haver mudanças importantes nos requisitos ao longo do desenvolvimento	Perda de tempo desenvolvendo partes que depois não serão usadas e atrasos no cronograma.	Enfatizar desenvolvimento modular com baixo acoplamento entre módulos. Estabilizar arquitetura base o quanto antes. Implementar um sistema eficiente de gerenciamento de versões.
pr2	Pode haver concorrentes que lancem produtos antes	Chegar ao mercado depois da janela de oportunidade	Manter estudo constante de mercado para garantir que o produto tenha características inovadoras.
t8	Gestos muito parecidos podem significar comandos diferentes	O sistema pode interpretar erroneamente os comandos (desenhos, formas). Usuário pode ter que decorar muitos comandos diferentes.	Elaborar <i>design</i> de interface alternativo, considerando gestos e alguma forma de eliminar possíveis ambiguidades em gestos. Implementar sistema de ajuda <i>online</i> para gestos.
pe1	Necessidade de treinamento	Atrasos de cronograma e custos com treinamento	Pesquisar e encomendar bibliografia para treinamento de equipe nas tecnologias necessárias. Prever orçamento para treinamento.
t4	O usuário não vai escolher a ferramenta porque usa um processo de desenvolvimento diferente	Problemas relacionados à venda. Pode haver necessidade de implementar vários processos, o que vai contra a filosofia inicial da ferramenta. Grandes empresas já têm processo estabelecido e teriam que mudar	Verificar se existe possibilidade de definir um meta-processo adaptável para a ferramenta.
t6	Comandos de voz podem não ser corretamente entendidos.	Usuários frustrados	Projetar interfaces alternativas a comandos de voz
t9	Não existe um padrão ou referência para tais interfaces nem estudos de usabilidade	Necessidade de pesquisar padrões de usabilidade para interfaces 3d em ferramentas CASE	Prever realização de testes de usabilidade para a ferramenta. Prever ciclos de prototipação de interface.
t3	Poderá ser desenvolvida uma ferramenta com usabilidade falha	Problemas com usuário final (desinteresse)	Idem ao risco t9.
pe1	Bolsistas não vêem o projeto como carreira	Pode-se perder desenvolvedores ao longo do projeto, necessitando substituição	Usar programação em pares e padrões de codificação. Planejar integrações frequentes e posse coletiva de código.
l1	Pagamento de direitos autorais	Aumento de custo	Prever custos com direitos autorais no orçamento. Verificar existência de tecnologias livres.

Os planos de redução de probabilidade e impacto mostrados aqui são apenas sugestões elaboradas a partir de uma rápida reflexão sobre os riscos. Outros planos mais detalhados ou mesmo em direções diferentes poderiam ter sido elaborados.

Por vezes, pode haver também certa ambiguidade entre o plano ser de redução de impacto ou probabilidade. Isto é, pode haver determinadas ações que reduzam tanto a probabilidade quanto o impacto de um risco. Se o planejador de projeto começar a ter problemas em classificar um plano como um ou outro tipo, será mais interessante que crie apenas um conjunto de planos de mitigação de risco, pois a classificação dele como redução de probabilidade ou redução de impacto não é tão importante quando a necessidade de que seja efetivamente planejado e executado, se for o caso.

8.6 Plano de Contingência

O *plano de contingência* ou *plano de resposta ao risco* consiste em um conjunto de ações a serem efetuadas caso o risco efetivamente ocorra¹⁰⁸. Estes planos também devem ser incorporados ao plano do projeto, embora de forma opcional, pois sua execução não será sempre necessária.

A resposta a um risco, depois que ele já se tornou um problema, poderá ocorrer de diferentes formas, como por exemplo:

- a) *Eliminação*. Procura-se eliminar o problema e o risco alterando, por exemplo, o escopo do projeto, renegociando contratos, reestruturando a equipe, repensando tecnologias, etc.
- b) *Transferência*. Procura-se transferir o problema e o risco a outra parte, por exemplo, subcontratando outra empresa para desenvolver a parte do sistema que apresenta o risco. Isso pode ser feito tanto como ação de resposta ao risco como também como ação de mitigação, ou seja, prevenção.
- c) *Aceitação*. Simplesmente aceita-se as perdas ocasionadas pelo problema e segue-se em frente, se possível. O impacto do risco é que vai determinar a gravidade das consequências, que poderão ir desde um leve inconveniente até o cancelamento do projeto.

Todas essas possibilidades implicam em um custo para a empresa desenvolvedora. Assim, sempre será necessário avaliar qual a estratégia de melhor custo/benefício antes de implementar qualquer decisão.

8.7 Monitoramento de Riscos

O processo de gerência de um projeto envolve a conscientização do gerente de que embora existam planos de mitigação e contingência de riscos, os riscos são tão volúveis e imprevisíveis quanto os requisitos. Portanto, novos riscos podem surgir ou serem descobertos com o passar do tempo, ou sua importância pode mudar com o tempo. Assim, o gerente de projeto deverá estar sempre reavaliando o conjunto de riscos e sua importância.

Para monitorar riscos adequadamente é necessário que estes estejam documentados, seja em um sistema eletrônico de controle de riscos, seja em documentos eletrônicos, ou mesmo em papel. A primeira opção é recomendada quase sempre porque o monitoramento do risco implica em várias pessoas poderem analisar e modificar o *status* de um risco ao longo do tempo. Então, um sistema de controle de risco automático, que inclusive gere alarmes para responsáveis pela gerência ou execução de planos, é uma boa escolha.

De qualquer forma, um sistema gerenciador de riscos ou um documento em papel deverá ter algumas informações que são fundamentais. Wiegers (2007, p. 84) apresenta um exemplo de documento de monitoramento de risco. A lista abaixo contém estes elementos e mais o *status* que não consta na lista original:

¹⁰⁸ A este respeito, os fãs de Scott Adams poderão apreciar este filme: www.youtube.com/watch?v=K-qY_b8lo-k

- a) *ID*: o identificador do risco. Um número sequencial ou um mnemônico mais significativo. Esse ID é importante porque a descrição do risco pode mudar com o tempo e o ID não. Isso permite que um risco seja monitorado por um longo período mesmo que ele mude significativamente ao longo desse tempo.
- b) *Descrição*: a descrição do risco. Possivelmente a melhor forma de descrevê-lo seja através de uma causa e uma consequência (se/então). A descrição do risco normalmente não varia com o tempo, mas nada impede que isso aconteça, especialmente se novas informações sobre o risco forem obtidas.
- c) *Probabilidade*: qual a chance de o risco se tornar um problema. Pode ser usada a escala numérica (0-100%) ou a escala (alta/média/baixa). Eventualmente a probabilidade poderá ser apresentada como uma função do tempo, por exemplo, “pequena em seis meses, mas grande após este prazo”. Espera-se que a probabilidade dos riscos varie com o passar do tempo e essa medida é um dos principais itens que um gerente de projeto deve ter em mente ao monitorar riscos. Inicialmente espera-se que os planos de mitigação reduzam a probabilidade do risco se tornar um problema, mas qualquer risco pode ter sua probabilidade aumentada em função de causas imprevistas.
- d) *Impacto*: qual a perda ou dano ocasionadas, caso o risco se torne um problema. Aqui também pode ser usada a escala numérica ou alta/média/baixa. Da mesma forma, os planos de mitigação poderão alterar o impacto do risco, caso este ocorra. O impacto previsto também pode variar com o tempo, a medida que novos conhecimentos sobre o risco vêm a tona, embora talvez não varie tanto quanto a probabilidade.
- e) *Importância*: também chamada de *exposição*, é o produto da probabilidade pelo impacto. Conforme visto na Seção 8.4, os riscos de importância alta (ou numericamente acima de um limite predeterminado) devem ter seus planos de mitigação executados para que a probabilidade ou impacto sejam baixados, desta forma baixando também a importância do risco.
- f) *Primeiro indicador*: é uma descrição da situação inicial a partir da qual se pode concluir que o risco está se tornando um problema. O gerente de projeto deverá estar sempre atento a essa condição, e a equipe também deverá ser alertada para observar essa condição e reportá-la o quanto antes.
- g) *Planos de mitigação e contingência*: são os planos de redução de probabilidade e impacto, além do plano de resposta ao desastre, que devem ser elaborados, caso o risco seja de importância alta ou média. Esses planos são descritos como uma lista de atividades a serem executadas (um subprojeto) e, no caso dos riscos de importância alta, devem ter responsáveis e prazos atribuídos.
- h) *Responsável*: é a pessoa que deve responder pelos planos de mitigação e contingência, caso o risco seja de importância alta. Planos diferentes poderão ter diferentes responsáveis.
- i) *Prazo*: é a data na qual os planos de mitigação deverão ter sido devidamente executados, caso o risco seja de alta importância. No caso de abordagens iterativas como UP ou métodos ágeis, o prazo poderá ser descrito como o final de um dos ciclos iterativos.
- j) *Status*: é o estado do risco. Mesmo riscos de baixa probabilidade poderão, por azar, se tornarem problemas. Então o *status* de um risco poderá ser atribuído

independentemente de uma importância, probabilidade ou impacto. Um risco poderá ser somente um problema *potencial* (por *default*), ou um problema *atual*, caso a condição já tenha acontecido e se esteja sofrendo as consequências, ou ainda um problema *em tratamento*, caso os planos de contingência estejam sendo executados, ou, finalmente, um problema *resolvido*. Depois de resolvido, o risco poderá voltar a ser um problema potencial, caso possa ocorrer novamente (Figura 8-1). Caso o risco não seja resolvido com os planos de contingência, presume-se que ele esteja fora de controle, e neste caso, provavelmente o projeto não irá durar mais muito tempo, a não ser que um novo tratamento seja iniciado.

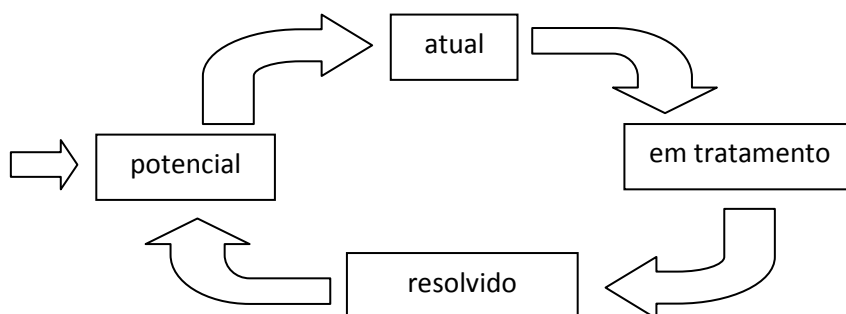


Figura 8-1: Um possível ciclo de vida para o status de um risco.

Essa lista ou arquivo de riscos deve ser ordenada de forma que os riscos com maior importância estejam no topo (mais visíveis) e os de menor importância no final da lista. Sugere-se que os 10 maiores riscos estejam sempre visíveis para a equipe. Se novos riscos surgirem, o menos importante deve ser tirado da vitrine. Isso é sugerido porque se a lista de riscos for grande demais talvez não se acabe prestando atenção naqueles que realmente são importantes.

Uma coisa que também não deve ser esquecida é que *novos* riscos podem surgir ao longo do projeto, não apenas aqueles que já estavam na lista.

É interessante colocar a descrição do risco na forma causa/consequência porque isso mostra mais claramente a estrutura do risco. Por vezes os analistas apresentam o risco de forma incompleta. Por exemplo, ao invés de apresentar apenas a causa como “não haver acordo sobre os requisitos entre os clientes”, ou apenas a consequência como “vamos poder atender apenas o cliente principal”, deve-se colocar as duas partes, formando uma sentença como “se os clientes não chegarem a um acordo sobre os requisitos *então* vamos poder atender apenas ao cliente principal”.

Poderá ser interessante que exista uma pessoa exclusivamente no papel de gerente de riscos de um projeto, aliviando, dessa forma, o trabalho do gerente de projeto. Isso é interessante porque talvez seja demais pedir a uma única pessoa que faça um projeto dar certo ao mesmo tempo em que ela precisa se preocupar com tudo o que pode (e vai) dar errado. O gerente de

risco seria uma pessoa com um pessimismo saudável (sem os exageros de Hardy¹⁰⁹), que vai estar sempre analisando as abordagens e situações e avaliando seus riscos.

No caso dos métodos ágeis, as reuniões diárias sugeridas por *Scrum* e XP poderão ser um excelente momento para a equipe avaliar e acompanhar os maiores riscos do projeto.

8.8 Controle de Risco

O *controle de risco* é o processo de observação e gerência que visa acompanhar o estado dos riscos de forma a evitar que se tornem problemas ou que seu prejuízo seja minimizado.

O controle do risco implica na execução prévia dos planos de mitigação de risco, embora alguns autores, como Wiegers (2007, p. 83), definam essas atividades como *resolução* de risco.

Infelizmente, a estimativa de esforço no caso de atividades relacionadas a riscos ainda não é tão previsível quanto o esforço de desenvolvimento, pois essas ações podem variar de dar um telefonema até realizar um projeto completo com cronograma, recursos e pessoal próprio.

Assim, para estimar esforço para controle de risco sugere-se aplicar a Lei de Parkinson (Parkinson, 1955) ou Síndrome do Estudante (procrastinação), que diz, entre outras coisas, que uma tarefa se expande até preencher todo o tempo livre. No caso do estudante, trata-se de verificar que dar um prazo de uma semana ou um mês para a conclusão de uma tarefa será indiferente em relação aos resultados, pois a tarefa acabará sendo feita próximo ao final do prazo, ou, caso tenha sido feita no início do período, o estudante gastará o restante do tempo para colocar detalhes, muitas vezes irrelevantes, até que o prazo se esgote.

Então, a sugestão para alocação de prazos para tratamento de riscos é que se use de uma análise subjetiva, como é feito com pontos de história, por exemplo, e que se avalie ao final do prazo estabelecido quais foram os efeitos das atividades de mitigação em relação ao risco. Se o risco tratado teve sua importância baixada de alta para média ou baixa, então as atividades de mitigação cumpriram seu objetivo e, por ora, nada mais precisa ser feito. Caso a importância do risco continue sendo alta, um novo plano deve ser feito e executado em relação ao risco, ou um novo prazo estabelecido para continuar o mesmo plano.

A execução de planos grandiosos pode até fazer com que a probabilidade de um risco se tornar problema chegue bem próxima a zero. Porém, neste caso, o projeto poderá ficar tão caro que talvez não valha mais a pena executá-lo. Assim, o controle do risco deve ser mensurado sempre por uma análise de custo e benefício.

8.9 Comunicação de Riscos

A comunicação na área de gerenciamento de riscos é fundamental, inicialmente nas atividades de identificação. Usualmente o pessoal técnico já tem consciência dos riscos que um projeto vai enfrentar, mas se não forem incentivados a comunicar essa informação, não o farão. Assim, uma reunião de planejamento inicial com toda a equipe é fundamental para que riscos sejam levantados e avaliados pela equipe como um todo.

¹⁰⁹ A hiena pessimista do desenho de Hanna Barbera, que está sempre dizendo ao seu parceiro “Lippy, isso não vai dar certo” (www.hannabarbera.com.br/lippy/lippy.htm).

Poucas crises, estatisticamente, ocorrem de surpresa. Normalmente os problemas vão crescendo lentamente até que se transformem em uma crise. Isso não é diferente em projetos de software. O que vai fazer a diferença entre uma crise que cresce nas sombras até atingir um tamanho que coloque o projeto a perder e um problema que pode ser mitigado em seu início, é a capacidade da equipe de perceber e comunicar justamente esse início.

Duarte (2008)¹¹⁰ apresenta alguns problemas (riscos) em comunicação que podem levar a crises, alguns dos quais são relevantes para empresas de desenvolvimento de software:

- a) Declaração inadequada.
- b) Opção discutível.
- c) Boatos.
- d) Vazamento de informação.
- e) Política interna.
- f) Falta de integração.
- g) Atendimento frágil.
- h) Questões pendentes.
- i) Falta de recursos.
- j) *Briefing* obscuro.
- k) Dados conflitantes.
- l) Postergar decisões.
- m) Funcionários insatisfeitos.
- n) Falta de orientação interna.

A comunicação de riscos não necessariamente evita o problema, mas, como foi dito, pode ajudar para que medidas antecipatórias ou medidas corretivas sejam administradas o mais cedo possível. Além disso, a comunicação adequada de riscos pode ajudar a empresa a compreender melhor o risco e o problema, de forma a incorporar essa análise no patrimônio cognitivo da empresa.

Um dos fatores em comunicação de risco que deve estar sob atenção do gerente é que diferentes interessados terão diferentes percepções sobre o risco. É atribuída a Einstein a frase “o maior problema da comunicação é a ilusão de que ela ocorreu”. Assim, não basta apresentar a lista de riscos aos interessados e esperar que todos entendam o que devem fazer. É necessário, em muitos casos, enfatizar as perdas que podem ocorrer caso o risco não seja adequadamente prevenido. Os desenvolvedores estarão mais interessados nos riscos técnicos que podem dificultar seu trabalho, causando frustração. Os clientes estarão mais interessados nos riscos que poderão atrasar o cronograma ou afetar os custos do projeto. Os usuários estarão interessados em riscos que envolvem a qualidade do sistema. Assim, comunicar os riscos corretamente às partes interessadas é também um processo que deve ser cuidadosamente pensado e executado pelo gerente de projeto.

¹¹⁰ www.comunicacaoecrise.com/palestras/gestaobrasil2008jorge.pdf