Capítulo 8

Riscos

Este capítulo discute um dos principais problemas que causam 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 o 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), *o 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 a inviabilizar todo um projeto.

Pode-se dizer, genericamente, que a falta de planejamento em relação aos riscos é 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 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:

- *Identificação*: antes que os riscos possam ser tratados, eles precisam ser identificados. As seções 8.2 e 8.3 apresentam algumas técnicas de identificação de riscos.
- *Análise*: a análise transforma a lista de riscos potenciais em um documento mais útil para o planejador e o gerente de um projeto, pois, a partir dela, os riscos são priorizados, e o planejador e o gerente podem se concentrar nos riscos mais importantes sem perder tempo com os insignificantes. A Seção 8.4 trata da análise de riscos.
- *Planejamento*: o planejamento quanto aos riscos permite ao gerente prevenir problemas, em geral de três formas: 1) planejando e executando planos para reduzir a probabilidade de o risco ocorrer (Seção 8.5.1); 2) planejando e executando planos para reduzir o impacto do risco, caso ocorra (Seção 8.5.2); e 3) planejando as atividades de recuperação de projeto, caso o risco efetivamente tenha ocorrido (Seção 8.6).
- *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).
- *Controle*: em função de mudanças no *status* de um risco, alguns planos podem ter que ser executados. Muitas vezes, talvez seja necessário improvisar a resposta ao problema causado pelo risco (Seção 8.8).
- *Comunicação*: a comunicação é um processo fundamental ao longo de um projeto de software,

especialmente em relação à prevenção e ao tratamento de riscos. Assim, não há uma atividade específica de comunicação em riscos, pois se trata de uma prática que permeia todas as outras atividades (Seção 8.9).

O planejamento de riscos conta com alguns fatores inibidores, que, muitas vezes, fazem a equipe não estar preparada para lidar com imprevistos. Entre outras coisas, existe uma cultura de aversão ao risco. Mas, em projetos de software, nem todo o otimismo do mundo vai proteger o andamento das coisas, especialmente se problemas ocorrerem ao longo dele. De qualquer modo, é necessário estar preparado. Ninguém espera que ocorra um incêndio; mesmo assim, os bombeiros estão sempre de prontidão.

8.1 Plano de Gerência de Riscos

Todo projeto de software deve ter um *plano de gerência de riscos*. Esse plano inclui os vários elementos apresentados neste capítulo. Em resumo, ele deve mostrar:

- Quais são os riscos identificados (Seções 8.2 e 8.3).
- Uma análise qualitativa ou quantitativa de cada risco que indique, por exemplo, a probabilidade de sua ocorrência e de seu impacto sobre o projeto, caso ocorra (Seção 8.4).
- Como a probabilidade de o risco ocorrer pode ser reduzida (plano de redução de probabilidade, Seção 8.5.1).
- Como o impacto do risco, caso ocorra, pode ser reduzido (plano de redução de impacto, Seção 8.5.2).
- O que fazer se o que era um risco se tornar efetivamente um problema (plano de resposta ao risco, Seção 8.6).
- 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 ele ocorrer, seu prejuízo seja reduzido.

Uma característica altamente desejada para um planejador e mesmo para 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, poderá salvar o projeto no futuro ou pelo menos mantê-lo dentro do cronograma e do orçamento previstos. Além disso, nem todos os riscos precisam preocupar o planejador de projeto. Como será visto mais adiante, apenas os riscos de maior exposição merecerão mais atenção.

Outro princípio importante é a antecipação das atividades de maior risco, como preconizado nos modelos Espiral, UP e métodos ágeis. A ideia é a seguinte: se um risco pode 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:

- *Conhecidos*: são aqueles já identificados e para os quais a equipe possivelmente estará preparada.
- *Desconhecidos*: são aqueles que poderiam ter sido descobertos se as medidas de identificação adequadas tivessem sido tomadas.
- *Impossíveis de prever*: são aqueles que não teriam sido identificados nem mesmo com as melhores técnicas de identificação.

Assim, o plano de gerência de riscos vai poder tratar apenas o primeiro grupo 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 de forma organizada a situações totalmente imprevistas.

8.2 Identificação de Riscos

Normalmente, um risco compõe-se de três elementos:

- Uma *causa*, na forma de uma condição incerta ou desconhecida, mas que pode ocorrer com uma determinada probabilidade. Por exemplo, o uso de uma tecnologia nova que não é dominada pela equipe de desenvolvimento.
- Um *problema*, que pode ocorrer em função da causa. Por exemplo, a equipe ter dificuldades sérias para implementar os requisitos usando essa tecnologia.
- Um *efeito* causado pelo problema em um ou mais objetivos do projeto ou iteração. Por exemplo, um atraso no cronograma ou um produto desenvolvido com qualidade inferior à desejada.

Assim, um risco é um problema que tem uma causa e pode ocasionar um efeito. Se esse problema ocorrer, estima-se que um dos objetivos do projeto será impactado, seja no tempo, custo, qualidade, seja em 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 ao estouro dos custos do projeto (um efeito sobre o objetivo do orçamento)" (Dinsmore, Cabanis-Brewin, Abdollahyan, Anselmo, Cota & Cavalieri, 2009).

Mais adiante mostraremos que como a causa do risco está ligada à sua probabilidade de ocorrer, os planos de redução de probabilidade deverão agir na causa. Já o efeito do risco define o seu impacto e assim os planos de redução de impacto deverão agir nos efeitos. Estes dois são os planos de mitigação de risco. Finalmente, a componente "problema" do risco está relacionada à situação resultante e assim, os planos de contingência estarão ligados à tentativa de resolver ou absorver o problema caso ele efetivamente ocorra. A Tabela 8.1 resume estas relações.

Tabela 8.1 Relações entre as componentes do risco e seus planos

Componente do risco	**Propriedade do risco**	**Plano relacionado**
Causa	Probabilidade	Mitigação: redução de probabilidade
Efeito	Impacto	Mitigação: redução de impacto
Problema		Plano de contingência

O PMBOK (PMI, 2017), referência em gerenciamento de projetos, define que o risco é uma condição incerta que pode ter tanto um efeito positivo quanto um efeito negativo sobre o projeto. Assim, existiriam também os *riscos positivos* ou *oportunidades*. Mas, certamente, os riscos 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 serem abordados adequadamente.

O ideal é que o planejador tenha a sua disposição um catálogo com riscos que 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 mencionados. Sugestões para essa discussão são apresentadas nas subseções seguintes.

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

- *Concepção*: riscos de requisitos e de negócio.
- *Elaboração*: riscos de tecnologia e arquitetura de sistema.
- *Construção*: riscos de programação e teste de sistema.
- *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. Podem-se citar as seguintes:

- Uso de *checklists* predefinidos 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 desse tipo é o relatório SEI de taxonomia de riscos

(Carr, Konda, Monarch, Ulrich & Walker, 1993), apresentado na Seção 8.3.

- Reuniões e *brainstormings* com gerente e equipe de projeto com experiência em outros projetos.
- Análise de cenários e lições aprendidas em projetos anteriores com contexto semelhante.

A identificação de riscos deve considerar diferentes fontes, tais como tecnologia, pessoas e o próprio projeto. As subseções seguintes discutem mais detalhadamente estas possíveis fontes.

8.2.1 Riscos Tecnológicos

Os *riscos tecnológicos* estão relacionados a todas as incertezas referentes ao modo como a equipe será capaz de lidar com a tecnologia necessária para realizar o projeto. Quanto menos experiência a equipe tiver com essas tecnologias, maiores serão os riscos.

Projetos que envolvam diferentes sistemas de software e hardware enfrentarão problemas de compatibilidade frequentes. Tornar tais sistemas compatíveis demandará tempo e custo extras ao projeto. Assim, o planejador deve saber se diferentes tecnologias terão de interagir e qual é 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?

8.2.2 Riscos Relacionados a Pessoas

Há vários tipos de interessados em um projeto. Cada um dos papéis de interessado pode produzir um risco característico, como os descritos a seguir:

- *Riscos de pessoal*: projetos são executados por pessoas. Então, perder uma pessoa da equipe de forma permanente ou temporária pode ter um grande impacto na medida em que essa pessoa for insubstituível.
- *Riscos de cliente*: até que ponto o cliente se manterá interessado no projeto? Mudanças políticas ou administrativas poderão afetar o interesse da organização em investir no projeto? Mesmo que a organização mantenha o interesse no projeto, o cliente estará disponível para esclarecer requisitos e realizar testes?
- *Riscos de negócio*: muitas vezes, a empresa até constrói um bom produto no prazo e com o custo definidos, mas mesmo assim o projeto fracassa. Entre outras coisas, a organização pode não ter a habilidade necessária para vender o produto, ou a empresa pode ter essa habilidade, mas o produto não ter efetivamente apelo comercial.
- *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 do custo previstos. Esse tipo de risco costuma se manifestar das seguintes formas:

- *Riscos de requisitos*: a equipe, por ser inexperiente, pode não ter sido capaz de identificar corretamente os requisitos do projeto, o que causará problemas ao longo dele. Requisitos poderão ser insuficientes, excessivos ou incorretos. Ou, ainda, os requisitos podem ser naturalmente instáveis, em razão de características do próprio projeto.
- *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?
- *Riscos de orçamento*: até que ponto a verba necessária para o projeto está garantida? Os custos foram corretamente previstos em projetos anteriores?
- *Riscos de cronograma*: é possível que os prazos sejam alterados e a ordem em que as

funcionalidades devem ser entregues possa mudar? O planejador deve saber em que grau a equipe se mostrou 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 dessa taxonomia, pode ser elaborado um questionário ou *checklist*, a partir do qual riscos podem ser identificados.

São definidas três grandes classes de risco:

- Engenharia do produto.
- Ambiente de desenvolvimento.
- Restrições externas.

Cada uma dessas 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 a elas.

Inicialmente, a Tabela 8.2 apresenta o *checklist* referente à *engenharia do produto*, que tem como elementos de risco os requisitos, o *design*, a codificação e o teste de unidade e a integração, além de outros aspectos específicos de engenharia.

Tabela 8.2 *Checklist* de riscos referentes à engenharia do produto (Fonte: Carr, Konda, Monarch, Ulrich & Walker, 1993)

Área	Questão	Detalhamento
Requisitos	*Estabilidade*: os requisitos podem mudar durante o desenvolvimento?	Os requisitos são estáveis? Se não, quais os efeitos disso no sistema? (qualidade / funcionalidade / cronograma / integração / design / teste)
		As interfaces externas do sistema estão mudando ou vão mudar?
	Completeza: estão faltando requisitos ou estão especificados de forma incompleta?	Existem tópicos a serem esclarecidos nas especificações?
		Existem requisitos que se sabe que deveriam estar nas especificações, mas não estão? Se sim, é possível obter esses requisitos e colocá-los na especificação?
		O cliente tem expectativas ou requisitos que não estão escritos? Se sim, há forma de capturá-los? As interfaces externas são completamente definidas?
	Clareza: os requisitos estão obscuros ou necessitam interpretação?	Você é capaz de entender os requisitos da forma como estão escritos? Se não, há ambiguidades sendo resolvidas satisfatoriamente? Se sim, não há ambiguidades ou problemas de interpretação?
	Validade: os requisitos vão levar ao produto que o cliente tem em mente?	Existem requisitos que podem não especificar exatamente o que o cliente quer? Se sim, como você está resolvendo isso?
		Você e o cliente compreendem a mesma coisa a partir dos requisitos? Se sim, existe um processo para determinar isso?
		Como você valida os requisitos junto ao cliente? Prototipação? Análise? Simulação?
	Exequibilidade: os requisitos são exequíveis de um ponto de vista analítico?	Há requisitos que sejam tecnicamente difíceis de implementar? Se sim, quais são? Se sim, porque eles são difíceis de implementar? Se não, foram feitos estudos de exequibilidade sobre os requisitos? Se sim, qual seu grau de confiança em tais estudos?
	Precedentes: os requisitos especificam algo que nunca foi feito antes, ou que a empresa nunca fez antes?	Existe algum requisito do estado da arte? Tecnologias? Métodos? Linguagens? Hardware? Se não, algum destes é novo para a equipe? Se sim, a equipe tem conhecimento suficiente nestas áreas? Se não, há um plano para adquirir conhecimento nestas áreas?
	Escala: os requisitos especificam um produto maior, mais complexo ou requerendo mais organização do que a empresa tem experiência?	O tamanho e complexidade do sistema são uma preocupação? Se não, você já fez algo deste tamanho e complexidade antes? O tamanho requer uma organização maior do que o usual para a empresa?
Design	*Funcionalidade*: existem problemas potenciais para obter os requisitos funcionais?	Há algum algoritmo especificado que possa não satisfazer os requisitos? Se não, há algum algoritmo ou design que obtenha os requisitos de forma marginal?
		Como você determina a exequibilidade dos algoritmos e design? Prototipação? Modelagem? Análise? Simulação?
	Dificuldade: o design ou implementação serão difíceis de serem realizados?	Alguma parte do design depende de hipóteses otimistas ou não realistas?
		Existe algum requisito que seja difícil de obter um design? Se não, você tem soluções para todos os requisitos? Se sim, quais são os requisitos e porque eles são difíceis?
	Interfaces: as interfaces internas de hardware e software são bem definidas e controladas?	Há interfaces internas bem definidas? Software para software? Software para hardware?
		Há um processo para definir interfaces internas? Se sim, há um processo de controle de mudança para interfaces internas?
		Há hardware sendo desenvolvido em paralelo com o software? Se sim, as especificações do

		hardware estão mudando? Se sim, todas as interfaces com o software já foram definidas? Se sim, há modelos de design de engenharia que possam ser usados para testar o software?
	Performance: existem tempos de resposta ou taxas de transferência rigorosos?	Existem problemas com performance? Taxa de transferência? Escalonamento de eventos de tempo real assíncronos? Respostas em tempo real? Tempo para recuperação de falhas (*recovery timeline*)? Tempo de resposta? Tempo de resposta, acesso e número simultâneo de usuários da base de dados?
		Foi feita uma análise de performance? Se sim, qual o seu nível de confiança na análise feita? Se sim, você tem um modelo para rastrear a performance através do design e implementação?
	Testabilidade: o produto é difícil ou impossível de testar?	O software vai ser fácil de testar?
		O design inclui características que facilitam o teste?
		Os testadores se envolveram com a análise dos requisitos?
	Restrições de hardware: existem restrições apertadas no hardware alvo?	Arquitetura?
		Capacidade de memória?
		Taxa de transferência?
		Resposta em tempo real?
		Tempo para recuperação de falhas?
		Performance da base de dados?
		Funcionalidade?
		Confiabilidade?
		Disponibilidade?
	Software não desenvolvido: há problemas com software usado, mas não desenvolvido pela equipe?	Você está reusando ou fazendo reengenharia em software não desenvolvido pela equipe? Se sim, você prevê algum problema? Documentação? Performance? Funcionalidade? Prazo de entrega? Personalização?
		Se estiver usando COTS, há algum problema com este tipo de software? Documentação insuficiente para determinar interfaces, tamanho ou performance? Performance fraca? Requer uma grande parcela de memória ou da base de dados? É difícil de interfacear com o software? Não foi testado sistematicamente? Não está livre de defeitos? Não foi adequadamente mantido? O vendedor demora para responder?
		Você prevê algum problema com a integração de atualizações ou revisões de COTS?
Codificação e teste de unidade	*Exequibilidade*: a implementação do design é difícil ou impossível?	Existem partes do produto não completamente especificadas no design?
		Os algoritmos e design são fáceis de implementar?
	Teste: os níveis e tempos especificados para os testes de unidade são adequados?	Você começa o teste de unidade antes de verificar código com respeito ao design?
		Testes de unidade suficientes são especificados?
		Há tempo suficiente para realizar todo o teste de unidade que você acha necessário?
		Serão feitos relaxamentos nos testes de unidades se houver problemas de cronograma?
	Codificação/Implementação: há problemas com codificação e implementação?	Os designs e especificações estão em um nível adequado para permitir a codificação?
		O design muda à medida que o código é escrito?
		Há restrições de sistema que tornam o código difícil de ser feito? Tempo? Memória? Armazenamento externo?
		A linguagem de programação é adequada para este projeto?
		O projeto usa mais de uma linguagem? Se sim, há compatibilidade entre o código produzido pelos diferentes compiladores?
		O computador de desenvolvimento é o mesmo onde o sistema vai rodar? Se não, há diferenças relativas à compilação entre os dois computadores?
		Se estiver sendo desenvolvido hardware, suas especificações são suficientes para o desenvolvimento do software?
		As especificações do hardware mudam a medida que o software é codificado?
Integração e teste	*Ambiente*: o ambiente de integração e teste é adequado?	Haverá hardware suficiente para uma adequada integração e teste?
		Há problemas para desenvolver cenários realísticos e testar dados para demonstrar algum dos requisitos? Tráfego de dados especificado? Resposta em tempo real? Tratamento de eventos assíncronos? Interação multiusuário?
		Você é capaz de verificar a performance nas suas instalações? Se sim, isso é suficiente para todos os testes?
	Produção: a definição de interfaces e instalações são inadequadas ou o tempo insuficiente?	O hardware alvo estará disponível quando necessário?
		Critérios de aceitação foram acordados com o cliente para todos os requisitos? Se sim, há um acordo formal?
		As interfaces externas foram definidas, documentadas e transformadas em baselines?
		Há algum requisito que seja difícil de testar?
		Foi especificada integração de produto suficiente?
		Foi alocado tempo suficiente para integração e teste do produto?
		Se usar COTS, os dados do vendedor serão aceitos na verificação dos requisitos alocados a COTS? Se sim, o contrato é claro neste ponto?
	Sistema: a integração de sistema é descoordenada, a definição de interface é pobre ou as instalações são inadequadas?	Integração de sistema suficiente foi especificada?
		Tempo suficiente para integração de sistema e testes foi previsto?
		O produto será integrado a um sistema existente? Se sim, haverá um período de transição definido? Se não, como vai-se garantir que o programa funcionará corretamente depois de integrado?
		A integração do sistema vai ocorrer nas instalações do cliente?
Aspectos	*Manutenibilidade*: a	A arquitetura, design ou código criam dificuldades de manutenção?

específicos de engenharia	implementação será difícil de entender e manter?	A equipe de manutenção foi envolvida cedo no processo?
		A documentação do produto é suficiente para que seja mantido por uma organização externa?
	Confiabilidade: os requisitos de confiabilidade ou disponibilidade são difíceis de obter?	Existem requisitos de confiabilidade?
		Existem requisitos de disponibilidade? Se sim, os prazos de recuperação de falhas são um problema?
	Riscos à segurança (*safety*): os requisitos relacionados a riscos à segurança são inexequíveis ou não demonstráveis?	Existem requisitos relacionados a riscos à segurança? Se sim, você vê alguma dificuldade em obtê-los?
		Será difícil verificar a satisfação destes requisitos?
	Segurança do sistema (*security*): os requisitos de segurança do sistema são mais rigorosos do que o usual?	Existem requisitos de segurança no estado da arte ou sem precedentes?
		O sistema deve seguir regulamentos estritos de segurança estabelecidos por agência governamentais (como, por exemplo, Orange Book)?
		Você já implementou este nível de segurança antes?
	Fatores humanos: o sistema será difícil de usar por conta de interface com usuário mal definida?	Você vê alguma dificuldade em satisfazer os requisitos de fatores humanos? Se não, como você garante que vai satisfazer estes requisitos?
		Se estiver usando prototipação, é um protótipo do tipo *throw-away*? Se não, está fazendo prototipação evolucionária? Se sim, você tem experiência neste tipo de desenvolvimento? Se sim, há versões preliminares entregáveis? Se sim, isso complica o controle de mudança?
	Especificações: a documentação é adequada para o design, implementação e teste do sistema?	A especificação dos requisitos do software é adequada para o design do sistema?
		A especificação dos requisitos de hardware é adequada para o design e implementação do sistema?
		As interfaces externas necessárias foram bem especificadas?
		As especificações de teste são adequadas para testar completamente o sistema?
		Se já estiver na fase de implementação ou após ela, as especificações de design são adequadas para implementar o sistema? 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 esses elementos de risco são apresentadas na Tabela 8.3.

Tabela 8.3 *Checklist* de riscos referentes ao ambiente de desenvolvimento (Fonte: Carr, Konda, Monarch, Ulrich & Walker, 1993)

Área	**Questão**	**Detalhamento**
Processo de desenvolvimento	*Formalidade*: a implementação será difícil de entender e manter?	Há mais de um modelo de desenvolvimento sendo usado? Espiral? Cascata? Incremental? Se sim, a coordenação entre eles é um problema?
		Há planos formais e controlados para todas as atividades de desenvolvimento? Análise de requisitos? Design? Codificação? Integração e teste? Instalação? Garantia de qualidade? Gerenciamento de configuração? Se sim, os planos especificam bem o processo? Se sim, os desenvolvedores são familiarizados com os planos?
	Adequação: o processo é adequado para o modelo de desenvolvimento?	O processo de desenvolvimento é adequado para este produto?
		O processo de desenvolvimento é suportado por um conjunto de procedimentos, métodos e ferramentas compatíveis?
	Controle de processo: o processo de desenvolvimento é executado, monitorado e controlado com métricas? Os locais de desenvolvimento distribuídos são coordenados?	Todos seguem o processo de desenvolvimento? Se sim, como isso é garantido?
		Você consegue mensurar se o processo de desenvolvimento está atingindo suas metas de qualidade e produtividade?
		Se há locais de desenvolvimento distribuídos, há coordenação adequada entre eles?
	Familiaridade: os membros do projeto têm experiência no uso do processo? O processo é compreendido por toda a equipe?	As pessoas estão confortáveis com o processo de desenvolvimento?
	Controle de produto: há mecanismos para controlar a mudança no produto?	Existe um mecanismo de controle de rastreabilidade de requisitos, que rastreia requisitos desde sua especificação até os casos de teste?
		O mecanismo de rastreabilidade é usado na avaliação de impacto de mudanças de requisitos?
		Existe um mecanismo formal de controle de mudança? Se sim, ele cobre todas as mudanças nas baselines de requisitos, design, código e documentação?
		As mudanças em qualquer nível são mapeadas para cima até o nível de sistema e para baixo até o nível de teste?
		Há análise adequada quando novos requisitos são adicionados ao sistema?
		Você tem algum meio de rastrear interfaces?
		Os planos e procedimentos de teste são atualizados como parte do processo de mudança?
Sistema de	*Capacidade*: existe poder de processamento,	Há estações de trabalho e poder de processamento para toda a equipe?

desenvolvimento	estações de trabalho, memória e espaço de armazenamento suficientes?	Há capacidade suficiente para fases que se entrelaçam como codificação, integração e teste?
	Adequação: o sistema de desenvolvimento dá suporte a todas as fases, atividades e funções?	O sistema de desenvolvimento suporta todos os aspectos do projeto? Análise de requisitos? Análise de performance? Design? Codificação? Teste? Documentação? Gerenciamento de configuração? Requisitos e gerenciamento de rastreabilidade?
	Usabilidade: qual a facilidade de uso do sistema de desenvolvimento?	As pessoas avaliam que o sistema de desenvolvimento é fácil de usar?
		Existe boa documentação sobre o sistema de desenvolvimento?
	Familiaridade: existe pouca experiência anterior da empresa ou membros da equipe com o sistema de desenvolvimento?	As pessoas já usaram estas ferramentas e métodos antes?
	Confiabilidade: o sistema de desenvolvimento sofre de erros, paralização, e capacidade de backup nativa insuficiente?	O sistema de desenvolvimento é considerado confiável? Compilador? Ferramentas de desenvolvimento? Hardware?
	Suporte ao sistema: existe suporte ágil de vendedor ou especialista no sistema de desenvolvimento?	A equipe foi treinada no uso das ferramentas?
		Você tem acesso a especialistas no seu uso?
		Os vendedores respondem rapidamente aos problemas?
	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?	Você vai entregar o sistema e desenvolvimento ao cliente? Se sim, recursos, cronograma e orçamento foram alocados para esta entrega?
Processo de gerência	*Planejamento*: o planejamento é oportuno, líderes técnicos foram incluídos e planos de contingência realizados?	O desenvolvimento é gerenciado de acordo com o plano? Se sim, as pessoas apagam incêndios com frequência?
		As pessoas de todos os níveis estão incluídas no planejamento de seu próprio trabalho?
		Existem planos de contingência para riscos conhecidos? Se sim, como você determina a hora de ativar estes planos?
		Questões de longo prazo foram adequadamente tratadas?
	Organização de projeto: os papeis e relacionamentos estão claros?	A organização do desenvolvimento é efetiva?
		As pessoas compreendem os seus papeis e os papeis dos outros?
		As pessoas sabem quem tem autoridade para quê?
	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?	O projeto tem gerentes experientes? Gerenciamento de projeto de software? Com desenvolvimento de software? Com o processo de desenvolvimento em uso? Com o domínio da aplicação? Com projetos deste tamanho ou complexidade?
	Interfaces da equipe de desenvolvimento: existem interfaces fracas com o cliente, outros contratados ou gerentes de nível superior?	A gerência comunica problemas para cima e para baixo na linha hierárquica?
		Os conflitos com o cliente são documentados e resolvidos no tempo apropriado?
		A gerência envolve as pessoas apropriadas nos encontros com o cliente? Líderes técnicos? Desenvolvedores? Analistas?
		A gerência trabalha de forma que todas as facções do lado do cliente estejam representadas nas decisões sobre funcionalidade e operação?
		É uma boa política apresentar uma visão otimista ao cliente ou gerentes de nível superior?
Métodos de gerência	*Monitoramento*: as métricas de gerência são definidas e o progresso do desenvolvimento rastreado?	Existem relatórios de status estruturados periódicos? Se sim, as pessoas obtêm respostas aos seus relatórios de status?
		A informação apropriada é reportada aos níveis organizacionais corretos?
		O progresso é comparado com o plano? Se sim, a gerência tem uma visão clara sobre o que está acontecendo?
	Gerenciamento de pessoal: o pessoal do projeto é treinado e usado adequadamente?	O pessoal é treinado nas habilidades requeridas pelo projeto? Se sim, isso é parte do plano do projeto?
		Há pessoas alocadas ao projeto que não possuem o perfil necessário?
		É fácil para os membros do projeto conseguir ação da gerência?
		Os membros em todos os níveis estão conscientes de seu status versus plano?
		As pessoas sentem que é importante manter o plano?
		A gerência consulta as pessoas antes de tomar decisões que afetam o trabalho delas?
	Garantia de qualidade: existem procedimentos e recursos adequados para garantir a qualidade do produto?	A função de garantia de qualidade tem pessoal adequado para o projeto?
		Existem mecanismos adequados para garantir a qualidade? Se sim, todas as áreas e fases possuem procedimentos de qualidade? Se sim, as pessoas estão acostumadas a trabalhar com estes procedimentos?
	Gerenciamento de configuração: os procedimentos de gerenciamento de configuração, mudança e instalação são adequados?	Há um sistema gerenciador de configuração adequado?
		A função de gerenciamento de configuração tem pessoal adequado?
		É necessária coordenação com algum sistema instalado? Se sim, há gerenciamento de configuração adequado para o sistema instalado? Se sim, o sistema de gerenciamento de configuração sincroniza o seu trabalho com as mudanças no local?
		O sistema vai ser instalado em múltiplos locais? Se sim, o sistema de gerenciamento de configuração permite múltiplos locais?
Ambiente de trabalho	*Atitude em relação a qualidade*: há falta de orientação relacionada ao trabalho com qualidade?	Todos os níveis de pessoal estão orientados para procedimentos de qualidade?

		O cronograma costuma ficar no caminho da qualidade?
	Cooperação: falta espírito de equipe? Os conflitos resultantes requerem intervenção da gerência?	As pessoas trabalham cooperativamente através das fronteiras funcionais?
		As pessoas trabalham cooperativamente na direção de metas comuns?
		A intervenção da gerência às vezes é necessária para fazer as pessoas trabalharem juntas?
	Comunicação: há pouca consciência a respeito de metas ou missão e comunicação pobre de informação técnica entre membros da equipe e gerentes?	Há boa comunicação entre os membros do projeto? Gerentes? Líderes técnicos? Desenvolvedores? Testadores? Gerente de configuração? Equipe de qualidade?
		Os gerentes são receptivos às comunicações da equipe? Se sim, você se sente a vontade para pedir ajuda ao seu gerente? Os membros são capazes de comunicar riscos mesmo sem ter uma solução pronta?
	Moral: há uma atmosfera não produtiva ou não criativa?	As pessoas sentem que não há reconhecimento ou recompensa por trabalho feito acima das expectativas?
		Como está a moral no projeto? Se não estiver bem, qual é o principal motivo para a baixa moral?
		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 apresentados na Tabela 8.4.

Tabela 8.4 *Checklist* de riscos referentes às restrições externas (Fonte: Carr, Konda, Monarch, Ulrich & Walker, 1993)

Área	**Questão**	**Detalhamento**
Recursos	*Cronograma*: o cronograma é inadequado ou instável?	O cronograma tem sido estável?
		O cronograma é realístico? Se sim, o método de estimação é baseado em dados históricos? Se sim, o método de estimação funcionou bem no passado?
		Existe alguma coisa para a qual um cronograma realístico não foi preparado? Análises e estudos? Garantia de qualidade? Treinamento? Cursos e treinamento em manutenção? Equipamentos? Sistema de desenvolvimento que vai ser entregue?
		Há dependências externas que poderiam impactar o cronograma?
	Pessoal: a equipe é inexperiente, sem conhecimento de domínio, sem habilidades ou em número insuficiente?	Há alguma área em que esteja faltando pessoal qualificado? Engenharia de software e métodos de análise de requisitos? Especialistas em algoritmos? Design e métodos de design? Linguagens de programação? Métodos de integração e teste? Confiabilidade? Manutenibilidade? Disponibilidade? Fatores humanos? Gerenciamento de configuração? Garantia de qualidade? Ambiente alvo? Nível de segurança? COTS? Reuso de software? Sistema operacional? Banco de dados? Domínio de aplicação? Análise de performance? Aplicações de tempo crítico?
		Há pessoal adequado para formar a equipe de projeto?
		A equipe é estável?
		Você tem acesso às pessoas certas quando precisa delas?
		Os membros da equipe já implementaram sistemas deste tipo?
		O projeto depende de algumas poucas pessoas?
		Há problemas para obter pessoal?
	Orçamento: o orçamento é insuficiente ou instável?	O orçamento é estável?
		O orçamento é baseado em uma estimativa realística? Se sim, o método de estimativa é baseado em dados históricos? Se sim, o método funcionou bem no passado?
		Funcionalidades ou características do sistema já foram removidas para reduzir custo?
		Há alguma coisa para a qual um orçamento adequado não foi alocado? Análises e estudos? Garantia de qualidade? Treinamento? Cursos de manutenção? Equipamento? Sistema de desenvolvimento a ser entregue?
		Mudanças no orçamento acompanham mudanças nos requisitos? Se sim, esta é uma parte padrão do processo de controle de mudança?
	Instalações: as instalações são adequadas para construir e entregar o produto?	As instalações de desenvolvimento são adequadas?
		O ambiente de integração é adequado?
Contrato	*Tipo de contrato*: o tipo de contrato é uma fonte de risco para o projeto?	Que tipo de contrato foi feito? (custo mais prêmio, preço fixo, ...). Isto apresenta algum problema?
		O contrato representa uma carga para algum aspecto do projeto? Declarações de trabalho? Especificações? Descrições de itens de dados? Partes? Envolvimento excessivo de cliente?
		A documentação requerida é uma carga? Quantidade excessiva? Cliente detalhista e exigente? Longo ciclo de aprovações?
	Restrições: o contrato causa alguma restrição?	Há problemas com direitos de dados?
	Dependências: o projeto tem dependência de algum produto ou serviço externo que podem afetar o produto?	Contratados associados?
		Contratado principal? (se você for um subcontratado)
		Subcontratados?
		Vendedores ou fornecedores?
		Equipamento ou software desenvolvido pelo cliente?
Interfaces	*Cliente*: há problemas com o	O ciclo de aprovação do cliente é adequadamente rápido? Documentação? Revisões de projeto?

de Comunicação Externas	cliente como ciclo longo de aprovação de documentos, comunicação fraca ou conhecimento inadequado do domínio?	Revisões formais?
		Você sempre segue em frente antes de receber aprovação do cliente?
		O cliente entende os aspectos técnicos do sistema?
		O cliente entende software?
		O cliente interfere com processos ou pessoas?
		O gerente trabalha com o cliente para alcançar entendimentos e decisões mútuas em tempo aceitável? Entendimento sobre requisitos? Critérios de teste? Ajustes de cronograma? Interfaces?
		Quão efetivos são seus mecanismos para chegar a um acordo com o cliente? Grupos de trabalho (contratuais)? Encontros de intercâmbio técnico (contratuais)?
		Todas as facções do cliente são envolvidas para se chegar a acordos? Se sim, há um processo formalmente definido?
		A gerência apresenta um quadro realista ou otimista ao cliente?

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

8.4 Análise de Riscos

Uma vez identificados os riscos potenciais, a análise de riscos vai determinar quais são verdadeiramente relevantes para que se gastem tempo e dinheiro com sua prevenção. Em geral, 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:

- *Probabilidade*: é a chance de que o risco realmente se torne um problema. Caso se disponha de séries históricas, a probabilidade pode ser medida quantitativamente. Por exemplo, se 35% dos projetos anteriores sofreram atraso no cronograma, então pode-se 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.
- *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. 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, sem inviabilizar os objetivos do projeto).
- *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 propriedade dos riscos, convém também usá-la para o cálculo da exposição do risco.
- *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 seu impacto aumentados.

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

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

Quando a probabilidade e o impacto do risco são avaliados qualitativamente, pode-se usar uma tabela como a Tabela 8.5, em que, a partir da combinação da probabilidade pelo impacto, se chega a um valor para a exposição do risco.

TABELA 8.5 Forma de cálculo para a exposição de um risco

		Probabilidade		
		Alta	**Média**	**Baixa**
Impacto	**Alto**	Alta exposição	Alta exposição	Média exposição
	Médio	Alta exposição	Média exposição	Baixa exposição
	Baixo	Média exposição	Baixa exposição	Baixa exposição

Normalmente, os atributos de riscos são definidos em função da experiência e do conhecimento de especialistas. Mas, se houver um registro de projetos anteriores, pode-se ter uma expectativa mais concreta sobre a ocorrência de alguns riscos. Assim, muitas vezes, o planejador de projeto poderá 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 exposição, os planos de mitigação são definidos ainda na fase de planejamento do projeto. Para riscos de média exposição, os planos são definidos e guardados para serem aplicados caso essa exposição aumente ao longo do projeto.

Há dois tipos de planos de mitigação: *plano de redução de probabilidade* e *plano de redução de impacto*. Os primeiros agem nas causas e os outros agem nos efeitos do risco.

Para exemplificar a apresentação de planos de mitigação, será usado um conjunto de riscos identificados e analisados conforme a Tabela 8.6. 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. Os riscos foram identificados e avaliados por uma turma de alunos da disciplina de Análise e Projeto de Sistemas II, na UFSC, em 2011. Eles foram avaliados em função de sua probabilidade (P) e impacto (I), e a exposição (E), a resultante foi calculada de acordo com a Tabela 8.5, com os valores *baixo* (B), *médio* (M) e *alto* (A). Os riscos aparecem na tabela ordenados de acordo com sua exposição.

TABELA 8.6 Identificação e análise de riscos de um projeto fictício

Id	**Causa**	**Risco**	**Efeito**	**Prob.**	**Imp.**	**Exp.**
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 longo.	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	A	M	A

Id	Causa	Risco	Efeito	Prob.	Imp.	Exp.
			pode ter que decorar muitos comandos diferentes.			
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 *touchscreen.*	Poderá ser desenvolvida uma ferramenta com usabilidade falha.	Problemas com usuário final (desinteresse).	M	M	M
pe1	O projeto será desenvolvido por bolsistas.	Bolsistas não veem o projeto como carreira.	Podem-se perder desenvolvedores ao longo do projeto, necessitando de 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

Id	Causa	Risco	Efeito	Prob.	Imp.	Exp.
t7	O código gerado pela ferramenta, por *default*, 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 exposição alta tiveram planos de mitigação elaborados de forma sumária e executados. Os riscos de exposição média deveriam ter seus planos apenas elaborados e guardados caso a exposição do risco aumentasse. Os riscos de baixa exposição deveriam apenas continuar sendo monitorados e, caso sua exposição aumentasse para média, deveriam ter seus planos elaborados (e executados, caso a exposição passasse para alta).

Nas subseções seguintes são discutidos estes planos de mitigação bem como apresentados exemplos de planos simplificados para os riscos indicados na Tabela 8.6.

8.5.1 Plano de Redução de Probabilidade de Risco

O *plano de redução de probabilidade de risco* consiste nas ações identificadas como necessárias para diminuir a probabilidade de que um risco ocorra. Esse tipo de plano deve agir nas *causas* do risco, ou seja, na segunda coluna da Tabela 8.6. Por exemplo, se é bem provável que a equipe tenha dificuldade com uma nova tecnologia a ser usada, podem-se realizar cursos de treinamento para diminuir a probabilidade de isso se tornar um estorvo ao andamento do projeto.

Riscos classificados como de alta exposição devem ter planos 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 preveem que atividades de estudo e mitigação de riscos sejam previstas e incorporadas ao projeto.

No caso de riscos de média exposição, é recomendado apenas elaborar os planos para serem aplicados mais tarde caso seja necessário.

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

Considerando o exemplo da Tabela 8.6, os planos simplificados de redução de probabilidade poderiam ser elaborados como mostrado na Tabela 8.7.

TABELA 8.7 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. Inspecionar requisitos. Procurar produtos semelhantes na Internet e analisá-los. Planejar o desenvolvimento de protótipos.
pr2	O tempo de desenvolvimento pode ser longo.	Pode haver concorrentes que lancem produtos antes.	Planejar desenvolvimento orientado a cronograma com entregas de versões parciais usáveis em intervalos de dois 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 catalogá-los. Definir hierarquia de comandos e comandos baseados

Id	Causa do risco	Risco	Plano de redução de probabilidade
			em contextos para reduzir a quantidade de comandos necessários.
pe1	Ainda não se sabe se será possível contratar equipe com experiência nas tecnologias.	Necessidade de treinamento.	Publicar anúncio solicitando currículos para pessoas com as habilidades desejadas.
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.	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 não 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 gráficos 3D ou em níveis de profundidade.	Não existe um padrão ou referência 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 *touchscreen.*	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 não veem o projeto como carreira.	Verificar o valor de salário de mercado. Verificar a possibilidade de oferecer salários mais atraentes. Verificar a 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.

Deve-se observar que os planos não necessariamente garantem que os riscos vão ser mitigados. Por exemplo, o plano para o risco l1 prevê verificar a existência de soluções livres. Mas não há garantia de que elas existam.

8.5.2 Plano de Redução de Impacto de Risco

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

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

Novamente, os riscos de alta exposição determinam a existência de um plano detalhado de redução de impacto com atividades previstas, prazos e responsáveis. Riscos de média exposição poderão

receber apenas uma lista de ações a serem efetuadas em caso de necessidade, sem necessariamente identificar responsáveis ou prazos, e os riscos de baixa exposição serão apenas monitorados.

Considerando o exemplo da Tabela 8.6, os planos de redução de probabilidade poderiam ser elaborados como mostrado na Tabela 8.8.

TABELA 8.8 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 constante estudo 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 *design* de interface alternativo, considerando gestos e alguma forma de eliminar possíveis ambiguidades em gestos. Implementar sistema de ajuda *on-line* 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 se 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 metaprocesso adaptável para a ferramenta.
t6	Comandos de voz podem não ser entendidos corretamente.	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 a 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 veem o projeto como carreira.	Podem-se perder desenvolvedores ao longo do projeto, necessitando de 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

Id	Risco	Efeito	Plano de redução de impacto
			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, também pode haver certa ambiguidade entre o plano ser de redução de impacto ou de 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 de um ou outro tipo, será mais interessante que ele crie apenas um conjunto de planos de mitigação de risco, pois sua classificação como redução de probabilidade ou redução de impacto não é tão importante quanto 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 realizadas caso o risco efetivamente se torne um problema[1].

[1]A esse respeito, os fãs de Scott Adams poderão apreciar o filme disponível no QR Code acima. [QRC 8.1]

Esses planos também devem ser incorporados ao plano do projeto, embora de forma opcional, pois sua execução nem sempre será necessária.

A resposta a um risco, depois que ele já se tornou um problema, pode ocorrer de diferentes formas:

- *Por eliminação*: procura-se eliminar o problema e o risco, alterando, por exemplo, o escopo do projeto, renegociando contratos, reestruturando a equipe, repensando tecnologias etc.
- *Por transferência*: procura-se transferir o problema e o risco a outra parte, por exemplo, subcontratando outra organização para desenvolver a parte do sistema que apresenta o risco. Isso pode ser feito tanto como ação de resposta ao risco como na forma de ação de mitigação, ou seja, prevenção.
- *Por aceitação*: simplesmente aceitam-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 um custo para a organização desenvolvedora. Assim, sempre será necessário avaliar qual é a estratégia de melhor custo/benefício antes de se implementar qualquer decisão.

8.7 Monitoramento de Riscos

O processo de gerência de um projeto envolve conscientizar o gerente de que, embora existam planos de mitigação e contingência de riscos, esses riscos são mais volúveis e imprevisíveis do que os requisitos. Portanto, novos riscos podem surgir ou ser descobertos – ou sua exposição pode mudar – com o passar do tempo. Assim, a equipe deverá estar sempre reavaliando o conjunto de riscos e sua exposição.

Para monitorar riscos adequadamente, é necessário documentá-los preferencialmente em um sistema eletrônico de controle de riscos. Essa opção é quase sempre a mais recomendada, porque o monitoramento do risco implica 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.

O sistema gerenciador de riscos deverá conter algumas informações fundamentais. Wiegers (2007) apresenta uma proposta de estrutura para monitoramento de risco. A lista a seguir contém os elementos desta estrutura proposta e também o *status* que não consta da estrutura original:

- *ID*: o identificador do risco. Pode ser 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 sua descrição mude significativamente.
- *Descrição*: a descrição do risco. Possivelmente a melhor forma de descrevê-lo seja através da tríade causa/risco/efeito. A descrição do risco normalmente não varia com o tempo, mas nada impede que isso aconteça, especialmente se forem obtidas novas informações sobre ele.
- *Probabilidade*: diz respeito à chance de o risco se tornar um problema. Pode ser usada a escala probabilística (0 a 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 esse 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 de o risco se tornar um problema, mas qualquer risco pode ter sua probabilidade aumentada em função de causas imprevistas.
- *Impacto*: refere-se à perda ou ao dano ocasionados caso o risco se torne um problema. Aqui também pode ser usada a escala numérica ou a escala alto/médio/baixo. Da mesma forma, os planos de mitigação poderão alterar o impacto do risco, caso ele ocorra. O impacto previsto também pode variar com o tempo à medida que novos conhecimentos sobre o risco vêm à tona, embora talvez não varie tanto quanto a probabilidade.
- *Exposição*: é o produto da probabilidade pelo impacto. Conforme visto na Seção 8.4, os riscos de exposição alta (ou numericamente acima de um limite predeterminado) devem ter seus planos de mitigação executados para que a probabilidade ou o impacto sejam baixados, baixando também a exposição do risco.
- *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 observá-la, reportando-a o quanto antes.
- *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 exposição alta ou média. Esses planos são descritos como uma lista de atividades a serem executadas (um subprojeto) e, no caso dos riscos de exposição alta, devem ter responsáveis e prazos atribuídos.
- *Responsável*: é a pessoa que deve responder pelos planos de mitigação e contingência, caso o risco seja de alta exposição. Planos diferentes poderão ter responsáveis diferentes.
- *Prazo*: é a data na qual os planos de mitigação deverão ter sido devidamente executados, caso o risco seja de alta exposição. No caso de abordagens iterativas, como UP ou métodos ágeis, o prazo poderá ser descrito como o final de uma das iterações.
- *Status*: é o estado do risco. Mesmo riscos de baixa probabilidade podem, por azar, se tornar problemas. Então, o *status* de um risco poderá ser atribuído independentemente de sua exposição. Um risco poderá ser somente um problema *potencial* (por *default*) ou um problema *atual*, caso o problema 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 problema não seja resolvido ou absorvido com os planos de contingência, presume-se que ele esteja fora de controle; nesse caso, provavelmente o projeto não vai durar muito tempo, a não ser que um novo tratamento seja iniciado.

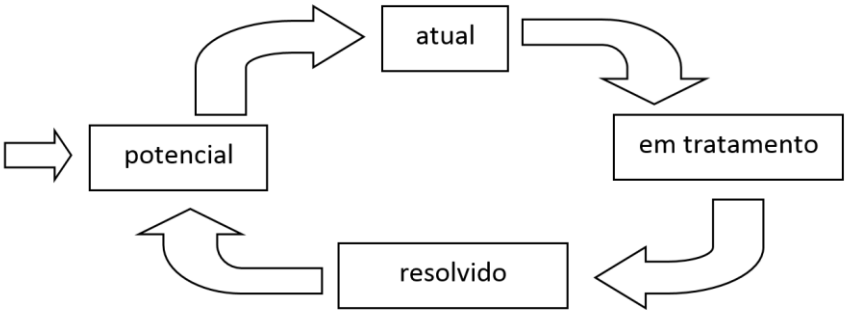

Figura 8.1 Um possível ciclo de vida para o *status* de um risco

Essa lista de riscos deve ser ordenada de forma que os riscos de maior exposição estejam no topo (mais visíveis) e os de menor exposição estejam no final da lista. Se novos riscos de alta ou média exposição surgirem, os de menor exposição devem ser movidos para baixo na lista. Isso é sugerido porque, se a lista de riscos for grande demais, deve-se manter em destaque no início da lista aqueles que são os de maior exposição.

É interessante colocar a descrição do risco na forma causa/problema/impacto, porque isso mostra mais claramente sua estrutura. Por vezes, os analistas apresentam o risco de forma incompleta.

Pode ser vantajoso ter uma pessoa exclusivamente no papel de gerente de riscos de um projeto, aliviando o trabalho do gerente de projeto. Isso é interessante porque talvez seja demais pedir a uma única pessoa que faça um projeto dar certo e, ao mesmo tempo, se preocupe com tudo o que pode (e vai) dar errado. O gerente de risco seria uma pessoa com um pessimismo saudável (sem os exageros de Hardy Har Har)[2]

[2]Hiena pessimista do desenho de Hanna Barbera que está sempre dizendo ao seu parceiro "Lippy, isso não vai dar certo".

, que vai estar sempre analisando as abordagens e situações e avaliando seus riscos.

No caso dos métodos ágeis, as reuniões diárias de *Scrum* podem ser um excelente momento para a equipe rapidamente 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, pelo menos, que seu prejuízo seja minimizado.

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

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

Assim, para estimar esforço para controle de risco, sugere-se considerar a Lei de Parkinson (Parkinson, 1995) 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 de um mês para a conclusão de uma tarefa será indiferente em relação aos resultados, pois a tarefa acabará sendo feita em período próximo ao final do prazo ou, caso tenha sido feita no início do período, o estudante gastará o restante do tempo para incluir detalhes, muitas vezes irrelevantes, antes que o prazo se esgote.

Desse modo, a sugestão para alocação de prazos para tratamento de riscos é que se utilize uma análise subjetiva, como é feito com pontos de histórias, por exemplo, e que ao final do prazo estabelecido se avalie quais foram os efeitos das atividades de mitigação em relação ao risco. Se o risco tratado teve sua exposição 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 exposição do risco continue

sendo alta, um novo plano deve ser traçado e executado em relação ao risco, ou um novo prazo deve ser estabelecido para a continuação desse mesmo plano.

A execução de planos grandiosos pode até fazer que a probabilidade de um risco se tornar problema chegue bem próximo a zero. Porém, nesse caso, o projeto poderá ficar tão caro que talvez não valha mais a pena executá-lo. Assim, o controle do risco deve ser sempre mensurado 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. Em geral, a equipe técnica já tem consciência dos riscos que um projeto vai enfrentar, mas, se não forem incentivados a passar essa informação adiante, 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.

Estatisticamente, poucas crises ocorrem de repente. Os problemas costumam ir crescendo lentamente até se transformarem em uma crise, e 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 justamente a capacidade da equipe de perceber e comunicar esse problema rapidamente.

A comunicação de riscos não necessariamente evita o problema, mas, como já foi dito, pode ajudar a fazer que medidas antecipatórias ou corretivas sejam administradas o mais cedo possível. Além disso, a comunicação adequada de riscos pode ajudar a organização a compreender melhor o risco e o problema, de forma a incorporar essa análise ao seu patrimônio cognitivo.

Um dos fatores em comunicação de risco que demandam a atenção do gerente é que diferentes interessados têm diferentes percepções sobre o risco. É atribuída a George Bernard Shaw 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 nos riscos que poderão atrasar o cronograma ou afetar os custos do projeto; os usuários em riscos que envolvem a qualidade do sistema. Assim, comunicar os riscos de maneira correta às partes interessadas é também um processo que deve ser cuidadosamente pensado e executado pela equipe.

REMISSIVO DO CAPÍTULO

acoplagem, 12
análise, 1, 2, 6, 9, 10, 11, 12, 13, 14, 20
causa, 3, 11, 18, 19
checklist, 4
comunicação, 1, 2, 10, 11, 20
concepção, 4
construção, 4
controle de risco, 17, 19
cronograma, 1, 2, 3, 5, 7, 9, 10, 11, 13, 15, 16, 19, 20
efeito, 3, 18
elaboração, 4
exposição, 1, 2, 12, 13, 14, 15, 17, 18, 19, 20
gerente de projeto, 1, 2, 18, 19
impacto, 1, 2, 3, 4, 8, 11, 12, 13, 14, 15, 16, 17, 18, 19
Lei de Parkinson, 19

mitigação, 1, 2, 3, 12, 13, 14, 17, 18, 19, 20
monitoramento, 1, 2, 12, 17
orçamento, 1, 2, 3, 5, 9, 10, 11, 16
planejamento, 1, 2, 4, 9, 12, 20
plano de contingência, 17
plano de gerência de riscos, 2
primeiro indicador, 18
probabilidade, 1, 2, 3, 11, 12, 13, 14, 15, 16, 18, 20
problema, 1, 2, 3, 7, 8, 11, 17, 18, 20
processo, 1, 2, 5, 6, 7, 8, 9, 11, 13, 15, 16, 17, 19, 20
proximidade, 12
rastreamento, 2
requisitos, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17
risco, 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
status, 2, 9, 17, 18, 19
transição, 4