# Plano de Gerenciamento de Riscos

## Gerenciamento de Riscos

*Um projeto, assim como a vida, é incerto.*

Os riscos não devem ser simplesmente listados; eles devem ser identificados para que sejam previstos e se possível:

1. **Diminuídos**: caso sejam ameaças;
2. **Maximizados**: caso sejam oportunidades;
3. **Controlados**: quando houver poucas estratégias para o seu enfrentamento.

O risco controla os planos de iteração no processo de desenvolvimento de software, ou seja, as iterações são planejadas considerando riscos específicos na tentativa de agir sobre os riscos. A lista de riscos é revista periodicamente para avaliar a eficácia das estratégias de resposta a riscos e, consequentemente, orientar as revisões no plano de projeto e nos planos de iteração subsequentes.

> O segredo do gerenciamento de riscos não é esperar que o risco aconteça, e, torne-se um problema ou defeito, para decidir o que fazer em relação a ele.

Assim como, uma mudança de alguns graus no percurso de um voo internacional produz um efeito significativo no local de aterrissagem do avião; de modo semelhante, gerenciar o risco antecipadamente é quase sempre menos dispendioso e penoso do que tentar solucioná-lo depois que virar um fato.

## Estratégias de Gerenciamento de Riscos

Para os **riscos negativos** ou **ameaças**, há três estratégias principais:

- **Prevenção de risco:** Reorganizar o projeto de modo que não seja afetado por um risco.
- **Transferência de risco:** Reorganizar o projeto de modo que alguém ou algo assuma o risco (o cliente, o fornecedor, o banco, um outro elemento etc.).
- **Aceitação de risco:** Decidir conviver com o risco como uma contingência. Monitore o sintoma do risco e escolha um plano de contingência que oriente sobre o procedimento a ser tomado em caso de risco.

No caso dos **riscos positivos** ou **oportunidades**, as opções de estratégias são as seguintes:

- **Exploração de risco**: eliminar a incerteza associada a um risco positivo, fazendo com que a oportunidade efetivamente aconteça.
- **Compartilhamento de risco**: atribuir parte da propriedade do risco a terceiros que possam capturar melhor a oportunidade em benefício do projeto.

- **Melhoramento do risco**: aumentar a exposição ao risco positivo, através do aumento da probabilidade e/ou do impacto caso ocorra. Isso se dá pela identificação e maximização dos acionadores dos riscos de impacto positivo.

Se decidir **aceitar** o risco, pode ser que você ainda deseja reduzi-lo, ou seja, tomar alguma ação imediata para reduzir seu impacto.

# Tipos de Riscos

É importante fazer distinção entre riscos diretos e indiretos, então, em poucas palavras:

1. **Risco Direto**: é aquele que permite um certo grau de controle;
2. **Risco Indireto:** é o que não pode ser controlado.

Embora não se possa ignorar os riscos indiretos, sua consequência é pequena no sentido prático: como não é possível alterá-los, não perca tempo se preocupando com eles. O mundo *pode* acabar amanhã, mas também *pode não* acabar. Então, se não acabar, é melhor que o trabalho não pare!

Algumas vezes, um risco indireto pode realmente ser um risco direto disfarçado. Por exemplo, a dependência de um fornecedor externo em relação a um ou mais componentes. Isso parece ser um risco indireto, mas se forem desenvolvidos planos de contingência para esses componentes, será possível controlar o risco: fornecedores alternativos podem ser escolhidos ou a funcionalidade pode ser desenvolvida por conta própria. Em vários casos, temos mais controle do que imaginamos.

No caso dos riscos indiretos, você deve tentar obter algum tipo de controle sobre eles ou simplesmente reconhecê-los e continuar o trabalho. Não adianta se preocupar com uma situação que você não pode mudar.

## 1. Riscos de Recursos

### 1.1 Organização

- Há um compromisso suficiente neste projeto (incluindo gerenciamento, testadores, QA e outras partes externas, porém envolvidas)?
- Este é o maior projeto desta organização?
- Existe algum processo bem definido para a engenharia de software? Há captura e gerenciamento de requisitos?

### 1.2 Financeiro

- Os recursos financeiros estão disponíveis para a conclusão do projeto?
- Os recursos financeiros foram alocados para treinamento e acompanhamento de mentores?
- Existe alguma limitação em termos de orçamento, por exemplo, existe algum custo fixo estipulado para o sistema ou o sistema está sujeito a cancelamento?
- As estimativas de custo são precisas?

### 1.3 Pessoas

- Há pessoal suficiente disponível?

- Elas possuem capacidades e experiência apropriadas?
- Elas já trabalharam juntas antes?
- Elas acreditam no sucesso do projeto?
- Há representantes dos usuários disponíveis para as revisões?
- Há especialistas de domínio disponíveis?

### 1.4 Tempo

- O planejamento é realista?
- A funcionalidade pode ser gerenciada pelo escopo para cumprir as programações?
- Quando é a data de liberação?
- Há tempo para "*fazer isso corretamente*"?

## 2. Riscos do Negócio

- E se um concorrente conseguir obter primeiro a liderança no mercado?
- E se o orçamento para o projeto estiverem comprometidos (uma outra forma de fazer esta pergunta é "*O que pode garantir recursos financeiros adequados*")?
- O valor projetado para o sistema é maior que o custo projetado? (não se esqueça de considerar o valor temporal do dinheiro e o custo de capital).
- E se não puderem ser feitos contratos com os principais fornecedores?

## 3. Riscos de Escopo

- É possível medir o sucesso?
- Existe algum consenso sobre como medir o sucesso?
- Os requisitos são relativamente estáveis e foram bem compreendidos?
- O escopo do projeto é estável ou continua sendo expandido?
- As escalas de tempo de desenvolvimento do projeto são curtas e inflexíveis?

## 4. Riscos Tecnológicos

- A tecnologia foi aprovada?
- Os objetivos de reutilização são razoáveis?

  - Um produto de trabalho deve ser utilizado uma vez antes de poder ser reutilizado.
  - É possível que, somente após vários releases, um componente esteja estável o suficiente para ser reutilizado sem causar mudanças significativas.

- Os volumes de transações nos requisitos são razoáveis?
- As estimativas de taxa de transação merecem crédito? Elas são muito otimistas?
- Os volumes de dados são razoáveis? Os dados podem ser mantidos nos servidores disponíveis atualmente? Se os requisitos indicarem que uma máquina em específico ou um sistema de um departamento fará parte do projeto, os dados podem ser mantidos nesse local de forma razoável?
- Há requisitos técnicos diferentes ou desafiadores que exijam que a equipe de projeto resolva problemas com os quais não está familiarizada?
- O sucesso depende de produtos, serviços ou tecnologias novas ou não experimentadas, ou de hardware, software ou técnicas novas ou não aprovadas?
- Existem dependências externas das interfaces com outros sistemas, inclusive aqueles fora da corporação? As interfaces necessárias existem ou devem ser criadas?
- Há requisitos de disponibilidade e segurança extremamente inflexíveis, por exemplo: "o sistema nunca deve falhar"?
- Os usuários do sistema são inexperientes em relação ao tipo de sistema que está sendo desenvolvido?
- Há um risco crescente devido ao tamanho ou à complexidade do aplicativo ou à inovação da tecnologia?
- Existe algum requisito para suporte ao idioma nacional?
- É possível projetar, implementar e executar este sistema? Alguns sistemas são muito grandes ou complexos para funcionarem apropriadamente.

## 5. Riscos de Planejamento

A experiência mostra que 85% dos riscos causam um impacto direto ou indireto no planejamento e, portanto, causam implicitamente um impacto no custo. É possível que 5% causem apenas um impacto no custo. O restante não causa impacto direto no custo nem na programação, mas, na qualidade, por exemplo.

Se o prazo de entrega for considerado um empecilho, faça liberações gradativamente. Evite fazer uma liberação enorme na tentativa de cumprir a programação.

Alguns projetos têm prazos finais realmente "inalteráveis". O software para analisar ao vivo o resultado de uma eleição durante a noite, por exemplo, terá pouco valor se for lançado na semana seguinte à eleição. Ou o software pode tornar-se obsoleto em relação aos dos concorrentes: eles lançam um produto melhor que o seu, enquanto você ainda está no meio da construção. De repente, você não está mais no jogo e não pode fazer quase nada em relação a isso. Entretanto, normalmente poucos projetos têm um prazo de entrega tão crítico. Os atrasos na maioria das vezes afetam o custo. Em geral, faça com que o compromisso com a programação seja igual à melhor estimativa e considere alguma contingência razoável.

*compromisso = estimativa + contingência*

Algumas pessoas sugerem definir as expectativas de planejamento do mesmo modo que a estratégia de recuo, ou seja, baseá-las nos planos de contingência, porém isso é pessimista demais, pois *nem* todos os riscos irão realmente se concretizar.

Os riscos de programação são integrados a algumas ferramentas de estimativa e custo. Por exemplo, no modelo COCOMO (*Constructive Cost Model*), vários geradores de custo são fatores de risco reais, tais como:

- complexidade (cplx)
- restrições de tempo real (time)
- restrições de armazenamento (stor)
- experiência (Vexp)
- disponibilidade de ferramentas apropriadas (tool)
- pressão de programação (sced)

O objetivo do Plano de Gerenciamento de Risco é garantir que os riscos do projeto sejam devidamente identificados, analisados, documentados, mitigados, monitorados e controlados. Ele descreve a abordagem que será usada para identificar, analisar, priorizar, monitorar e mitigar riscos.

# 1. Sumário de Riscos

Considerando o mesmo sistema da atividade FURPS, escreva um texto com uma breve descrição do projeto e um resumo do risco total envolvido no projeto

Descrição do Projeto - Cockpit: Sistema de Gerenciamento de Chamados de Serviços de TI

O projeto do Cockpit é um ambicioso empreendimento voltado para o desenvolvimento de um sistema de gerenciamento de chamados de serviços de TI altamente especializado. Este sistema tem como objetivo principal garantir a entrega eficiente e dentro dos padrões de qualidade estabelecidos, alinhando-se estrategicamente com os níveis de SLA (Acordo de Nível de Serviço) contratados com nossos clientes. O Cockpit oferece uma plataforma completa e abrangente, permitindo o acompanhamento em tempo real do ciclo de vida de cada chamado, desde a sua criação até a sua resolução, oferecendo uma visão completa das operações de suporte técnico.

Resumo do Risco Total Envolvido no Projeto:

O projeto do Cockpit apresenta um conjunto significativo de riscos, dada a sua natureza complexa e a importância crítica para a eficácia das operações de serviços de TI. A seguir, um resumo dos principais riscos envolvidos no projeto:

1. Risco de desvio de SLA: A gestão de SLA é central para o sucesso do Cockpit. Qualquer desvio em relação aos níveis de serviço acordados pode resultar em insatisfação do cliente e perda de negócios.

2. Complexidade Técnica: O desenvolvimento do Cockpit envolve tecnologias avançadas e integração com outras ferramentas de TI. Isso apresenta o risco de atrasos devido a desafios técnicos inesperados.

3. Risco de Segurança de Dados: Dado que o sistema conterá informações críticas, a segurança de dados é de extrema importância. Qualquer violação de segurança pode ter implicações graves, incluindo perda de dados e danos à reputação.

4. Aderência Regulatória: Os requisitos regulatórios em relação ao gerenciamento de dados e operações de TI estão em constante evolução. O não cumprimento das regulamentações pode resultar em multas substanciais.

5. Escassez de recursos de desenvolvimento: Garantir a disponibilidade de uma equipe altamente qualificada e suficiente é crucial. Escassez de recursos pode levar a atrasos no projeto.

6. Aceitação do Usuário Final: A aceitação e a adoção do Cockpit pelos usuários finais são fundamentais. A resistência à mudança ou a falta de treinamento adequado podem representar um risco significativo.

7. Integração com Ferramentas Externas: A criação de hyperlinks para outras ferramentas e sistemas requer uma integração sólida. Problemas na integração podem afetar a funcionalidade do sistema.

8. Mudanças de Escopo: Mudanças constantes nos requisitos ou no escopo do projeto podem levar a atrasos e custos adicionais.

## 2. Tarefas de Gerenciamento de Riscos

Faça uma breve descrição das tarefas de gerenciamento de riscos a serem executadas durante o projeto. Nesta seção, você deve descrever o seguinte:

**A.** A abordagem a ser adotada para identificar riscos e como a lista de riscos será analisada e priorizada;

**B.** As estratégias de gerenciamento de riscos que serão usadas, incluindo estratégias de diminuição, anulação e/ou prevenção para os riscos mais significativos;

**C.** Como o status de cada risco significativo e as respectivas atividades de diminuição serão monitorados;

**D.** Cronogramas de revisão e relatório de riscos. Uma revisão dos riscos deve fazer parte da revisão de aceitação de cada iteração ou fase.

Identificação de Riscos:

1. Workshops de Brainstorming: Realização de sessões com a equipe de projeto para identificar riscos potenciais relacionados ao desenvolvimento do Cockpit. Considerando todas as áreas funcionais e requisitos do sistema.

2. Revisão de Documentação: Análise à documentação existente, incluindo especificações, requisitos e documentação técnica, para identificar riscos específicos relacionados ao sistema.

3. Consultas Externas: Consulta a especialistas internos e externos, se necessário, para obter insights sobre riscos específicos no gerenciamento de chamados de serviços de TI e sistemas semelhantes.

Análise e Priorização de Riscos:

1. Matriz de Impacto x Probabilidade: Criação de uma matriz que avalie o impacto potencial de cada risco em relação à sua probabilidade de ocorrência. Isso ajudará a priorizar os riscos mais significativos.

2. Avaliação Qualitativa e Quantitativa: Realização de uma análise qualitativa e quantitativa dos riscos para entender melhor suas implicações e efeitos financeiros.

Estratégias de Gerenciamento de Riscos:

1. Mitigação: Desenvolvimento de estratégias de mitigação para os riscos mais significativos. Isso pode envolver a implementação de medidas preventivas ou de contingência para reduzir o impacto ou a probabilidade dos riscos.

2. Transferência: Avaliação da possibilidade de transferir alguns riscos para terceiros, como seguradoras ou

fornecedores, quando aplicável.

3. Aceitação: Em alguns casos, pode ser decidido aceitar um determinado nível de risco, desde que seja devidamente documentado e aceito pela equipe de gerenciamento.

Monitoramento de Riscos:

1. Registro de Riscos: Manter um registro detalhado de todos os riscos identificados, incluindo descrições, classificações, estratégias de mitigação e responsáveis.

2. Status e Atualizações: Regularmente, revisaremos o status de cada risco significativo durante reuniões de projeto. Atualizando as informações à medida que o projeto avança e novas informações se tornam disponíveis.

3. Relatórios de Riscos: Preparação de relatórios de riscos que destaquem os riscos mais críticos, seu status atual e as atividades de mitigação em andamento. Esses relatórios devem ser compartilhados com a equipe de gerenciamento e partes interessadas relevantes.

Cronogramas de Revisão e Relatório de Riscos:

1. Revisões Regulares: Agendamento de revisões regulares de riscos como parte das revisões de iteração ou fase do projeto. Isso garantirá que os riscos sejam constantemente monitorados e atualizados.

2. Relatórios de Aceitação: Inclusão de revisão de riscos como parte do processo de aceitação de cada iteração ou fase do projeto. Isso permitirá que a equipe de gerenciamento tome decisões informadas sobre o andamento do projeto.

## 3. Orçamento

Especifique em reais o orçamento disponível para o gerenciamento dos riscos do projeto

Valores mensais R$57.120,00:

QA: R$5.500,00
Analista DevOps: R$12.000,00
Analista de Cloud: R$8.200,00
Serviços Cloud:
Serviços (AWS):

- EC2 principal: R$25.000,00
- EC2 de contingência: R$800,00
- Backup Full 4x/mês: R$400,00
- Backup Incremental 2x/dia: R$20,00
- S3: R$500,00
- ERS: R$1.500,00
- VPC: R$200,00
- GuardDut: R$500,00

Serviços Testes:

- PyTest: R$0,00

- Locust: R$0,00

Valor Anual: R$691.440,00

## 4. Itens de Risco a Serem Gerenciados

Crie uma lista dos itens de risco que foram identificados; uma das melhores práticas do setor é publicar e manter visível uma lista dos 10 principais riscos que são considerados significativos o bastante para o projeto empregar recursos para o seu gerenciamento. Você poderá manter uma lista maior se assim for exigido pela prática organizacional ou pelo contrato.

1. **Risco de Segurança Cibernética:** Vulnerabilidades no software que podem ser exploradas por hackers ou ameaças cibernéticas, resultando em violações de dados, perda de informações confidenciais e danos à reputação da empresa.

2. **Risco de Falhas no Software:** Bugs, erros de programação e problemas de desempenho que podem causar falhas no software, interrupções nos serviços e insatisfação do usuário.

3. **Risco de Conformidade:** Não atender aos padrões de segurança, regulamentações legais ou requisitos do setor, o que pode resultar em penalidades legais e problemas de conformidade.

4. **Risco de Integração:** Dificuldades na integração do software com outros sistemas ou aplicativos, causando conflitos de dados ou interrupções no funcionamento do software.

5. **Risco de Dependência de Terceiros:** Dependência de bibliotecas de terceiros, serviços de nuvem ou componentes externos que podem ser descontinuados, causando problemas de compatibilidade.

6. **Risco de Recursos Humanos:** Falta de habilidades ou recursos adequados para desenvolver, manter e suportar o software, o que pode levar a atrasos e problemas de qualidade.

7. **Risco de Desempenho:** Subdimensionamento da infraestrutura de hardware, resultando em desempenho inadequado do software sob carga.

8. **Risco de Backup e Recuperação:** Falhas nos processos de backup e recuperação de dados, resultando em perda de informações críticas.

9. **Risco de Garantia da Qualidade:** Falhas nos processos de teste e garantia da qualidade que podem resultar em bugs não detectados no software.

10. **Risco de Disponibilidade:** Problemas que podem afetar a disponibilidade contínua do software, como falhas de servidores ou serviços.