

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gunar Martins Pires

**Gestão de Riscos em Software com Métodos
Ágeis: uma Pesquisa Survey**

Uberlândia, Brasil

2025

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gunar Martins Pires

**Gestão de Riscos em Software com Métodos Ágeis: uma
Pesquisa Survey**

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, como parte dos requi-
sitos exigidos para a obtenção título de Ba-
charel em Sistemas de Informação.

Orientador: Prof. Dr. Ronaldo Castro de Oliveira

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia, Brasil

2025

Gunar Martins Pires

Gestão de Riscos em Software com Métodos Ágeis: uma Pesquisa Survey

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Trabalho aprovado. Uberlândia, Brasil, 14 de maio de 2025:

Prof. Dr. Ronaldo Castro de Oliveira
Orientador

Prof. Dr. Daniel Duarte Abdala

Prof. Dr. Mauricio Cunha Escarpinati

Uberlândia, Brasil
2025

Resumo

O desenvolvimento de software, por muito tempo pautado por processos prescritivos e rígidos, enfrentava dificuldades em lidar com mudanças de escopo devido à rápida evolução tecnológica. Com o tempo, esses processos deram lugar a abordagens mais iterativas e flexíveis, conhecidas como métodos ágeis, impulsionados pela publicação e assinatura do “Manifesto Ágil”. Nesse contexto, destaca-se o Scrum, um *framework* ágil que estrutura o trabalho em ciclos curtos (*sprints*), promovendo inspeção contínua e adaptação rápida. A gestão de riscos, por sua vez, refere-se ao conjunto de práticas que visam identificar, analisar, monitorar e responder a eventos que possam impactar negativamente (ou positivamente) os objetivos do projeto. Mesmo em cenários ágeis, essa gestão continua essencial para garantir entregas previsíveis e alinhadas ao valor de negócio. Este trabalho teve como objetivo investigar de que forma a gestão de riscos vem sendo aplicada em projetos de desenvolvimento ágil de software. Para isso, foi realizada uma pesquisa de levantamento (*survey*) com profissionais atuantes na área de TI na cidade de Uberlândia e regiões vizinhas. Espera-se que os dados obtidos contribuam para uma melhor compreensão sobre o nível de maturidade dessas práticas no contexto ágil e forneçam subsídios para estudos futuros sobre a integração entre métodos ágeis e a gestão de riscos.

Palavras-chave: Gestão de Riscos, Métodos Ágeis, Desenvolvimento de *Software*, Scrum, *Survey*.

Lista de ilustrações

Figura 1 – O modelo cascata	14
Figura 2 – O paradigma de prototipação	15
Figura 3 – Modelo espiral típico	16
Figura 4 – O processo unificado	17
Figura 5 – Interações nos processos de gerenciamento de projetos	19
Figura 6 – Estrutura genérica do ciclo de vida de um projeto	21
Figura 7 – Exemplo de ciclo de vida previsível	23
Figura 8 – O framework do Scrum	28
Figura 9 – Visão geral do gerenciamento do risco do projeto	32
Figura 10 – Setor de atuação	44
Figura 11 – Metodologias ágeis mais utilizadas	45
Figura 12 – Comunicação aberta sobre riscos	45
Figura 13 – Identificação proativa dos riscos	46
Figura 14 – Riscos de segurança são tratados com rigor	47
Figura 15 – Importância de riscos técnicos versus riscos de negócio	47
Figura 16 – Priorização dos riscos com matriz de impacto e probabilidade	48
Figura 17 – Discussão e mitigação dos riscos	49
Figura 18 – Inclusão de riscos de negócio	49
Figura 19 – Avaliação quantitativa dos riscos	50
Figura 20 – Ferramentas ou simulações de apoio aos riscos	51
Figura 21 – Revisão dos riscos ao final de cada ciclo	51
Figura 22 – Ferramentas para monitor os riscos	52
Figura 23 – Incidência de riscos não previstos	53
Figura 24 – Documentação e compartilhamento dos riscos	53
Figura 25 – Contribuições da gestão de riscos	54

Lista de abreviaturas e siglas

PMBOK	Project Management Body of Knowledge
PMI	Project Management Institute
TI	Tecnologia da Informação
PU	Processo Unificado
UML	Unified Modeling Language
GAP	Gerenciamento Ágil de Projetos
OOPSLA	Object-Oriented Programming, Systems, Languages and Applications; conferência anual sobre linguagens e sistemas orientados a objetos.
ISO	International Organization for Standardization
IEEE	Institute of Electrical and Electronics Engineers
DS	Design Science; abordagem metodológica centrada na construção e avaliação de artefatos.
DSR	Design Science Research
SLM	Systematic Literature Mapping
XP	Extreme Programming

Sumário

1	INTRODUÇÃO	9
1.1	Conceituação sobre o tema	9
1.2	Objetivos gerais e específicos	10
1.2.1	Objetivo geral	10
1.2.2	Objetivos específicos	10
1.3	Justificativa	10
1.4	Organização do trabalho	11
2	REVISÃO BIBLIOGRÁFICA	12
2.1	Processos de desenvolvimento de software	12
2.1.1	Modelo de processo genérico	13
2.1.2	Modelo de processo prescritivo	13
2.1.2.1	Modelo cascata	14
2.1.2.2	Modelo de processo de prototipação	14
2.1.2.3	Modelo de processo evolucionário	15
2.1.2.4	Modelo de processo unificado	16
2.2	Gerenciamento de projetos de TI	18
2.2.1	Ciclos de vida de um projeto	21
2.2.1.1	Ciclo de vida previsível	22
2.2.1.2	Ciclo de vida adaptativo	23
2.3	Desenvolvimento de software com modelos ágeis	24
2.4	Metodologia do Scrum	27
2.4.1	Origem	27
2.4.2	Definições	27
2.4.2.1	Papéis da equipe Scrum	29
2.4.2.2	Artefatos do Scrum	29
2.4.2.3	Eventos do Scrum	30
2.4.3	Justificativa para a adoção do Scrum	31
2.5	Gestão de riscos em projetos de TI	31
2.5.1	Planejar o gerenciamento dos riscos	32
2.5.2	Identificar os riscos	32
2.5.3	Realizar a análise qualitativa dos riscos	32
2.5.4	Realizar a análise quantitativa dos riscos	33
2.5.5	Planejar as respostas aos riscos	33
2.5.5.1	Estratégias para riscos negativos ou ameaças	34

2.5.5.2	Estratégias para riscos positivos ou oportunidades	34
2.5.6	Controlar os riscos	35
3	TRABALHOS RELACIONADOS	36
3.1	Trabalho 1	36
3.2	Trabalho 2	36
3.3	Trabalho 3	37
3.4	Trabalho 4	38
3.5	Trabalho 5	38
4	APLICAÇÃO DE GESTÃO DE RISCOS EM PROCESSO DE DE- SENVOLVIMENTO ÁGIL	40
4.1	Contextualização	40
4.2	Método de pesquisa	40
4.3	Definição do processo	42
4.4	Aplicação	43
4.5	Resultados	43
4.5.1	Setor de atuação	44
4.5.2	Utilização de metodologias ágeis na empresa	44
4.5.3	Comunicação aberta sobre riscos	45
4.5.4	Identificação proativa dos riscos	46
4.5.5	Riscos de segurança são tratados com rigor	46
4.5.6	Importância de riscos técnicos versus riscos de negócio	47
4.5.7	Priorização dos riscos com matriz de impacto e probabilidade	48
4.5.8	Discussão e mitigação dos riscos	48
4.5.9	Inclusão de riscos de negócio	49
4.5.10	Avaliação quantitativa dos riscos	50
4.5.11	Ferramentas ou simulações de apoio aos riscos	50
4.5.12	Revisão dos riscos ao final de cada ciclo	51
4.5.13	Ferramentas para monitor os riscos	52
4.5.14	Incidência de riscos não previstos	52
4.5.15	Documentação e compartilhamento dos riscos	53
4.5.16	Contribuições da gestão de riscos	54
5	CONCLUSÃO	55
	REFERÊNCIAS	56

APÊNDICES	58
APÊNDICE A – QUESTIONÁRIO	59

1 Introdução

1.1 Conceituação sobre o tema

A gestão de riscos é um processo essencial para o sucesso de um projeto, abrangendo as etapas de planejamento, identificação, análise, definição de respostas e controle dos riscos. Conforme descrito no (PMI, 2013), esse processo busca aumentar a probabilidade e o impacto de eventos positivos, ao mesmo tempo em que se reduz a probabilidade e o impacto de eventos negativos, por meio de uma abordagem sistemática ao longo de todo o ciclo de vida do projeto.

No contexto do desenvolvimento de software, a gestão de riscos assume um papel ainda mais relevante, devido à complexidade técnica, às incertezas dos requisitos e à pressão por prazos curtos e alta qualidade. De acordo com (PRESSMAN; MAXIM, 2021), falhas na gestão desses riscos podem levar a atrasos, retrabalho e insatisfação do cliente. Com base em (SOMMERVILLE, 2011) pode-se dizer que a gestão de riscos eficaz depende da identificação precoce e do acompanhamento contínuo das ameaças e oportunidades ao longo do projeto.

Em ambientes ágeis, a natureza dinâmica e incremental do desenvolvimento de software exige uma abordagem proativa e iterativa para o gerenciamento de riscos. Nesse contexto, o uso de metodologias ágeis se destaca por apoiar todas as etapas desse processo, uma vez que essas abordagens são capazes de se adaptar rapidamente a mudanças e responder a *feedbacks* contínuos, garantindo maior flexibilidade e agilidade à execução dos projetos. O *framework* Scrum, por exemplo, incorpora práticas como reuniões diárias, revisões de *sprint* e retrospectivas que promovem a identificação constante de problemas e riscos ao longo do desenvolvimento (SUTHERLAND, 2014).

Embora os métodos ágeis não incluam um processo formal de gestão de riscos como o definido pelo PMBOK, eles integram mecanismos que contribuem para a mitigação de incertezas. Para (PRIKLADNICKI; WILLI; MILANI, 2014), a agilidade favorece a antecipação de problemas, a rápida adaptação e a transparência entre os membros da equipe, o que pode funcionar como um modelo prático de controle de riscos.

Dessa forma, compreender como a gestão de riscos é aplicada e percebida em projetos conduzidos sob metodologias ágeis torna-se fundamental para avaliar a efetividade dessas práticas. Este trabalho busca investigar essa relação a partir da perspectiva de profissionais da área, contribuindo para o entendimento teórico e prático da gestão de riscos em contextos ágeis.

1.2 Objetivos gerais e específicos

1.2.1 Objetivo geral

O presente trabalho tem como objetivo geral analisar a aplicação de técnicas de gestão de riscos em projetos de desenvolvimento de software que utilizam métodos ágeis, por meio de uma pesquisa de levantamento com profissionais da área de TI atuantes na cidade de Uberlândia e regiões vizinhas.

1.2.2 Objetivos específicos

1. Compreender os fundamentos da gestão de riscos segundo o guia PMBOK e sua aplicação em projetos de software;
2. Estudar os princípios e práticas das metodologias ágeis, com ênfase no *framework* Scrum;
3. Identificar os principais riscos enfrentados durante o desenvolvimento de software em ambientes ágeis;
4. Elaborar e aplicar um questionário de levantamento (*survey*) junto a profissionais da área para coleta de dados sobre a percepção e a prática da gestão de riscos em projetos ágeis;
5. Analisar os dados obtidos com o intuito de correlacionar os métodos ágeis utilizados com as abordagens de mitigação de riscos identificadas;
6. Avaliar as contribuições das práticas ágeis na gestão de riscos em projetos de desenvolvimento de software.

1.3 Justificativa

A gestão de riscos é fundamental para o sucesso de projetos de desenvolvimento de software, especialmente diante da crescente adoção de metodologias ágeis, que priorizam flexibilidade e entregas iterativas. Embora essas abordagens promovam rapidez e adaptação, a integração eficaz da gestão de riscos ainda enfrenta desafios em sua aplicação. Este estudo concentra-se na percepção e aplicação das técnicas de gestão de riscos entre profissionais de TI da cidade de Uberlândia e regiões vizinhas. Dessa forma, busca identificar lacunas e oferecer subsídios para aprimorar as práticas de gestão de riscos em ambientes ágeis, contribuindo tanto para a literatura acadêmica quanto para o desenvolvimento prático do setor, ao oferecer dados concretos sobre o cenário atual e possíveis caminhos de aprimoramento.

1.4 Organização do trabalho

Este trabalho está estruturado em cinco capítulos, além desta introdução. A seguir, apresenta-se um breve resumo dos conteúdos abordados em cada um deles:

No Capítulo 2, é realizada a revisão bibliográfica. Este capítulo tem como objetivo fornecer o embasamento teórico necessário para o entendimento do tema proposto abordando os principais conceitos relacionados aos: processos de desenvolvimento de software, gerenciamento de projetos de TI, métodos ágeis (com ênfase no *framework Scrum*) e gestão de riscos em projetos de tecnologia da informação.

O Capítulo 3 apresenta os trabalhos relacionados, destacando estudos e aplicações semelhantes à proposta deste trabalho, onde são discutidas contribuições relevantes da literatura, com o intuito de contextualizar a pesquisa.

No Capítulo 4, é descrita a aplicação prática da gestão de riscos em um processo de desenvolvimento ágil através de um *survey*. O capítulo contempla a contextualização do cenário estudado, o método de pesquisa, a definição do processo adotado, os procedimentos realizados e os resultados obtidos.

Por fim, o Capítulo 5 apresenta as conclusões do trabalho, sintetizando os principais resultados alcançados, as limitações identificadas durante o desenvolvimento da pesquisa e sugestões para investigações futuras.

2 Revisão Bibliográfica

2.1 Processos de desenvolvimento de software

Inicialmente, a compreensão dos conceitos básicos contribui para a análise de todo o ambiente do projeto. Segundo ([PRESSMAN; MAXIM, 2021](#)), o software é um elemento de sistema lógico e não físico, com características consideravelmente diferentes do hardware: o software não “se desgasta”. Os componentes de software são construídos após mapear as exigências do cliente para o formato de código executável em máquinas, utilizando uma linguagem de programação e processados por um tradutor. Quando um componente de hardware se desgasta, ele é substituído por uma peça de reposição; já no software, não existem peças de reposição. O hardware está suscetível aos fatores prejudiciais do ambiente, mas isso não afeta o software diretamente. Cada defeito de software indica um erro no projeto ou no processo, traduzido em código de máquina executável. Assim, as tarefas de manutenção de software envolvem uma complexidade maior do que as de manutenção de hardware.

Ao longo dos últimos 60 anos, na evolução de produtos e sistemas baseados em computador, o software tem se consolidado como elemento-chave e uma das tecnologias mais influentes. De uma ferramenta especializada em análise de informações e resolução de problemas, o software transformou-se em uma verdadeira indústria ([PRESSMAN; MAXIM, 2021](#)).

A engenharia de software é uma tecnologia em camadas que abrange métodos, ferramentas e processos, adotando uma abordagem sistemática, disciplinada e quantificável, aplicada por uma equipe de desenvolvimento na construção de sistemas complexos baseados em computador, com a camada de processos como fundamento. Um processo é um conjunto de tarefas, ações e atividades realizadas na criação de artefatos (documentos, dados, modelos, relatórios, etc.). Também pode ser considerado um processo como uma abordagem adaptável que permite à equipe de software escolher um conjunto de ações e tarefas, com o objetivo de entregar o software dentro do prazo e com qualidade para o cliente ou para aqueles que irão utilizá-lo ([PRESSMAN; MAXIM, 2021](#)).

A metodologia de processo genérica de software incorpora cinco atividades estruturais, aplicáveis a todos os tipos de processo de software, sendo elas ([PRESSMAN; MAXIM, 2021](#)):

- Comunicação: consiste no diálogo e na colaboração com o cliente e demais envolvidos, com o objetivo de compreender as metas do projeto e reunir os requisitos necessários para definir as funcionalidades e os recursos do software.

- Planejamento: é a atividade responsável por definir um percurso que oriente a equipe ao longo do desenvolvimento. Esse percurso é formalizado no plano de projeto de software, no qual são descritas as tarefas técnicas a serem executadas, os riscos potenciais, os recursos necessários, os artefatos a serem produzidos e o cronograma de trabalho.

- Modelagem: consiste na criação de uma representação preliminar do software, com o objetivo de visualizar a arquitetura geral, a forma como os componentes se integram e outras características relevantes do sistema.

- Construção: é a atividade que envolve tanto a implementação do código quanto a realização de testes, com o objetivo de identificar e corrigir erros durante a codificação.

- Entrega: o software é disponibilizado ao cliente, que o avalia e fornece *feedback* com base em sua experiência e nas funcionalidades observadas.

Essas cinco atividades metodológicas genéricas podem ser aplicadas tanto no desenvolvimento de programas pequenos e simples quanto na engenharia de sistemas computacionais grandes e complexos. Tais atividades são frequentemente utilizadas de forma iterativa, repetindo-se a cada ciclo de desenvolvimento e resultando em incrementos sucessivos do software. Cada incremento disponibiliza uma parte dos recursos e funcionalidades do sistema, tornando-o progressivamente mais completo a cada nova iteração (PRESSMAN; MAXIM, 2021).

2.1.1 Modelo de processo genérico

O modelo de processo genérico é desenvolvido com base nas exigências e, em seguida, transformado em um projeto. As cinco atividades metodológicas, comunicação, planejamento, modelagem, construção e entrega, formam a estrutura dessa metodologia para a engenharia de software. Além disso, um conjunto de atividades de apoio pode ser integrado ao longo do processo, como o acompanhamento e controle do projeto, a administração de riscos, a garantia da qualidade, o gerenciamento de configurações, as revisões técnicas e outras atividades de apoio essenciais ao processo (PRESSMAN; MAXIM, 2021).

2.1.2 Modelo de processo prescritivo

Os modelos de processo prescritivos são definidos como conjuntos estruturados de elementos e fluxos de trabalho previsíveis, com o objetivo de organizar e orientar o desenvolvimento de software. Embora todos os modelos de processo de software possam acomodar as cinco atividades metodológicas genéricas, cada um atribui ênfases distintas a essas atividades e define um fluxo de processo que invoca cada atividade de maneira única (PRESSMAN; MAXIM, 2021).

2.1.2.1 Modelo cascata

O modelo de cascata (ou modelo sequencial linear) adota uma metodologia estruturada e linear para o desenvolvimento de software, iniciando com a especificação dos requisitos do cliente por meio da comunicação, e seguindo pelas fases de planejamento, modelagem, construção e entrega, até culminar no suporte contínuo ao software finalizado. Esse modelo é amplamente reconhecido como um dos mais tradicionais na engenharia de software ([PRESSMAN; MAXIM, 2021](#)).

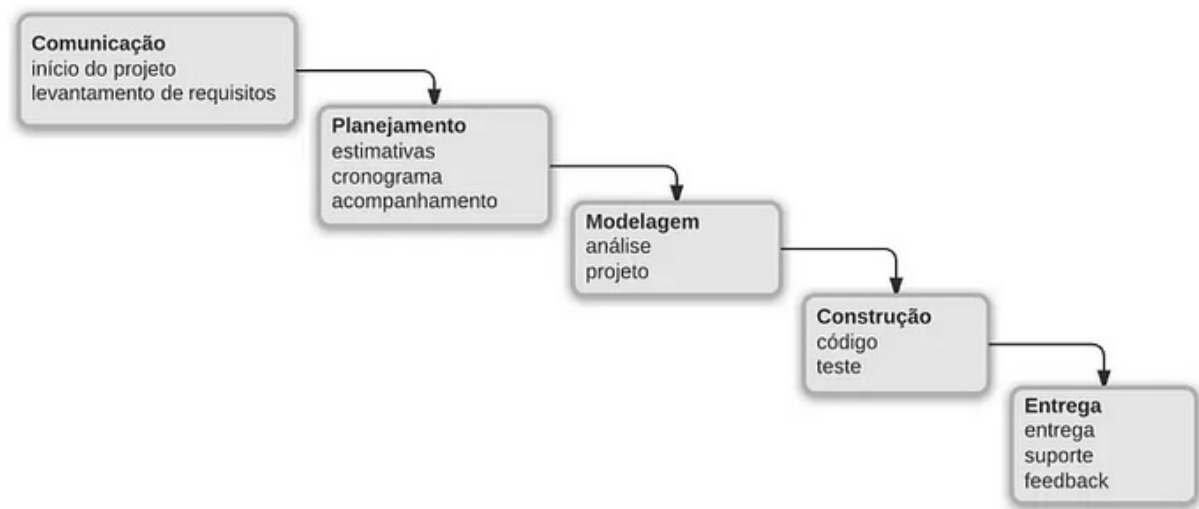


Figura 1 – O modelo cascata.

Fonte: ([PRESSMAN; MAXIM, 2021](#)).

2.1.2.2 Modelo de processo de prototipação

O modelo de prototipação é empregado como um recurso para facilitar a compreensão do produto em desenvolvimento por parte da equipe técnica. Essa abordagem é especialmente útil em contextos nos quais o cliente apresenta apenas objetivos gerais do software, sem detalhar de forma precisa os requisitos funcionais e os recursos desejados, o que pode gerar incertezas entre os desenvolvedores ([PRESSMAN; MAXIM, 2021](#)).

O paradigma de prototipação tem início com a comunicação entre os envolvidos, geralmente por meio de uma reunião, com o objetivo de definir o planejamento do software, incluindo os objetivos gerais, os requisitos iniciais e as áreas que demandam maior atenção. Em seguida, é planejada uma iteração de prototipação rápida, acompanhada da modelagem de um “projeto rápido”, que representa aspectos visuais do software perceptíveis ao usuário, como a organização visual da interface em telas. O protótipo é então construído, entregue e avaliado pelos envolvidos, que fornecem *feedback* para o refinamento contínuo dos requisitos ([PRESSMAN; MAXIM, 2021](#)).

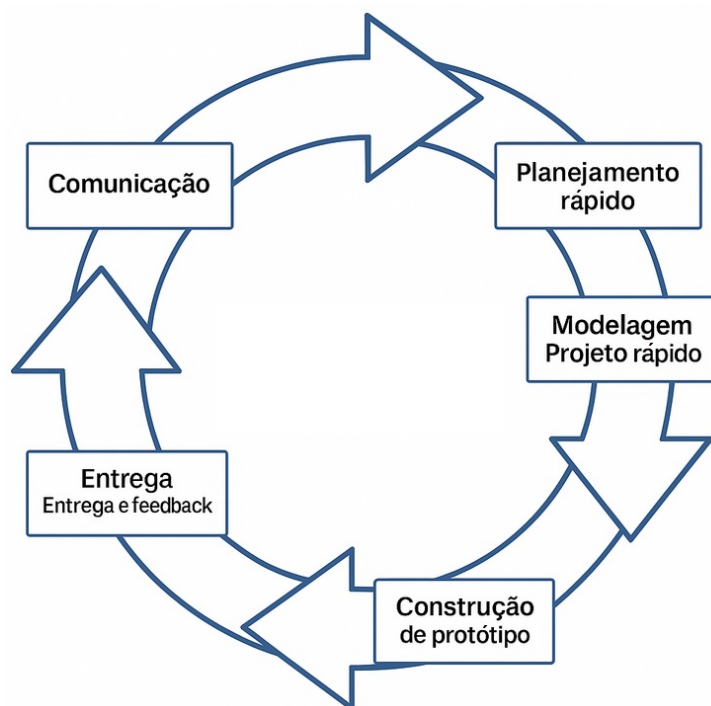


Figura 2 – O paradigma de prototipação.

Fonte: Adaptado de (PRESSMAN; MAXIM, 2021).

2.1.2.3 Modelo de processo evolucionário

Assim como outros sistemas complexos, o software passa por um processo contínuo de evolução. Diante de pressões do mercado e da concorrência, surgiu a necessidade de um modelo de processo especificamente voltado ao desenvolvimento de produtos que possam se expandir e se adaptar ao longo do tempo (PRESSMAN; MAXIM, 2021).

O modelo espiral representa uma abordagem evolucionária ao desenvolvimento de software, integrando a natureza iterativa da prototipação com a disciplina e o controle característicos do modelo cascata. O processo ocorre por meio da criação de sucessivas versões do sistema, iniciando por uma versão inicial simplificada, que pode assumir a forma de um protótipo, e evoluindo gradualmente até alcançar versões mais completas e refinadas (PRESSMAN; MAXIM, 2021).

Esse modelo organiza-se em torno de um conjunto de atividades metodológicas genéricas, definidas pela equipe de desenvolvimento. Cada uma dessas atividades percorre uma trajetória espiral, no sentido horário, iniciando-se no centro. A fase mais interna dessa espiral pode resultar na elaboração de uma especificação inicial do produto. Com o avanço pelas voltas subsequentes, o sistema evolui por meio da construção de protótipos progressivamente mais refinados. Custos e prazos são ajustados com base no *feedback* fornecido pelo cliente após cada entrega. Cabe ao gerente de projeto determinar o número de iterações necessárias para a finalização do software (PRESSMAN; MAXIM, 2021).

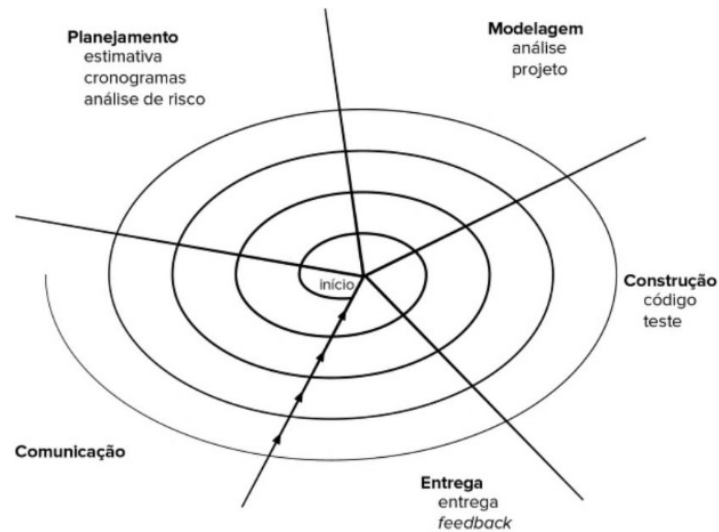


Figura 3 – Modelo espiral típico.

Fonte: (PRESSMAN; MAXIM, 2021).

Ao contrário de outros modelos de processo que se encerram com a entrega do software, o modelo espiral pode ser estendido para abranger todo o ciclo de vida do produto. Essa característica o torna especialmente adequado para o desenvolvimento de sistemas e softwares de grande porte, ao oferecer uma abordagem prática e adaptável. Um de seus diferenciais é o uso da prototipação como mecanismo para mitigação de riscos, já que o modelo exige a avaliação contínua de riscos técnicos em cada fase do projeto, possibilitando sua identificação e tratamento antes que causem impactos significativos (PRESSMAN; MAXIM, 2021).

2.1.2.4 Modelo de processo unificado

O Processo Unificado (PU) busca integrar os pontos fortes dos modelos tradicionais de desenvolvimento de software, adaptando-os a uma abordagem estruturada em torno da UML (*Unified Modeling Language*). A Linguagem de Modelagem Unificada (UML), por sua vez, foi concebida como uma linguagem padrão para modelagem e desenvolvimento de sistemas orientados a objetos. O PU enfatiza a comunicação eficaz com o cliente e a aplicação de métodos sistemáticos para representar a visão do usuário por meio de casos de uso. Além disso, adota um fluxo de trabalho iterativo e incremental, o que o torna especialmente adequado às demandas do desenvolvimento de software moderno (PRESSMAN; MAXIM, 2021).

O Processo Unificado divide-se em cinco “fases”, relacionadas às atividades genéricas, sendo elas: concepção, elaboração, construção, transição e produção (PRESSMAN; MAXIM, 2021). A figura a seguir ilustra essas fases.

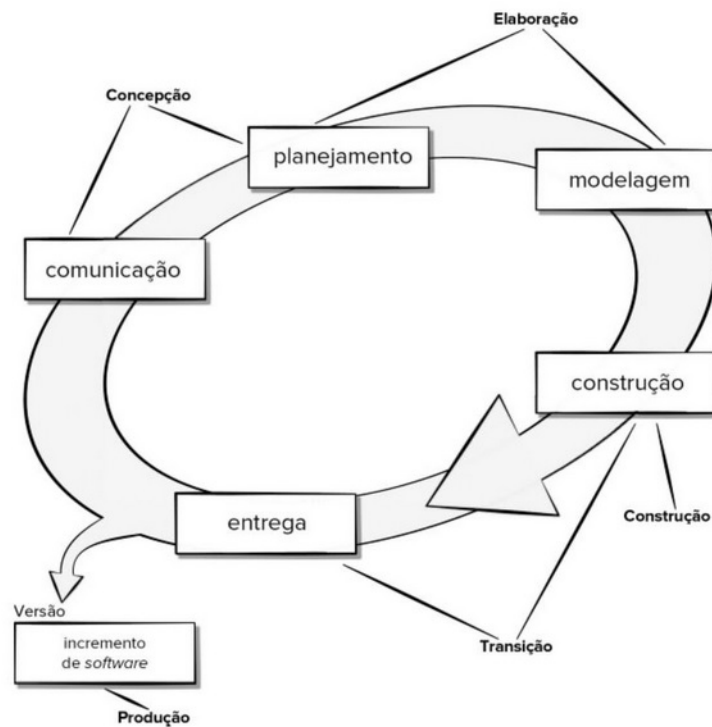


Figura 4 – O processo unificado.

Fonte: (PRESSMAN; MAXIM, 2021).

Na sequência, apresenta-se uma visão geral das fases do Processo Unificado (PRESSMAN; MAXIM, 2021):

- Fase de concepção do PU: nesta fase, ocorre a comunicação inicial com o cliente e o planejamento do projeto. Um conjunto de casos de uso preliminares descreve os principais requisitos de negócio, indicando os recursos e funcionalidades esperados na arquitetura de software segundo a perspectiva do usuário. A partir dessas informações, são realizados o planejamento dos recursos, a avaliação dos principais riscos e a definição de um cronograma preliminar para os incrementos de software.

- Fase de elaboração do PU: é composta por atividades de planejamento e modelagem com base no modelo de processo genérico. Nessa fase, os casos de uso preliminares são refinados e expandidos, e o planejamento inicial é ajustado conforme necessário. Também é criada uma base de arquitetura que contempla diferentes visões do software, incluindo: modelo de caso de uso, modelo de análise, modelo de projeto, modelo de implementação e modelo de entrega.

- Fase de construção do PU: nessa fase, são realizadas as atividades de construção definidas no modelo de processo de software genérico. À medida que os componentes são implementados, testes de unidade são desenvolvidos e executados para cada um deles, além de atividades de integração, como a montagem dos componentes e os testes de integração.

- Fase de transição do PU: abrange os estágios finais da atividade de construção genérica e os primeiros momentos da atividade de implantação, envolvendo a entrega do software e a coleta de *feedback*. Nessa fase, o software é disponibilizado aos usuários para testes, possibilitando o retorno de informações sobre defeitos ou ajustes necessários. Ao final dessa fase, o incremento torna-se uma versão utilizável do software.

- Fase de produção do PU: coincide com a atividade de entrega do processo genérico. Durante essa fase, realiza-se o monitoramento contínuo do uso do software, fornecendo suporte à infraestrutura e avaliando relatórios de defeitos e solicitações de mudanças.

2.2 Gerenciamento de projetos de TI

O gerenciamento de projetos envolve a aplicação de conhecimentos, habilidades, ferramentas e técnicas para atender aos requisitos do cliente. Segundo (SOMMERVILLE, 2011), a gestão de projetos é uma peça fundamental na engenharia de software. Para ele, embora um bom gerenciamento não garanta, necessariamente, o sucesso do projeto, o mau gerenciamento tende a resultar na falha do empreendimento.

Segundo (PMI, 2013), um projeto é um esforço temporário empreendido para criar um produto ou serviço, e, por ser de natureza temporária, deve possuir datas de início e término bem definidas. O término do projeto pode ser definido de diferentes formas, como: quando o cliente determina, quando o objetivo é alcançado, quando não é possível atingir a meta ou quando a necessidade do projeto deixa de existir.

O gerenciamento de projetos é um esforço integrado que exige o alinhamento e a conexão adequados de cada processo e produto com os demais, a fim de facilitar a coordenação. As ações adotadas em um processo tendem a impactar não apenas esse, mas também outros processos relacionados.

Os processos de gerenciamento de projetos são agrupados em cinco categorias, conhecidas como grupos de processos de gerenciamento de projetos (ou simplesmente grupos de processos), que são: iniciação, planejamento, execução, monitoramento e controle, e encerramento (PMI, 2013).

A figura a seguir apresenta o diagrama de fluxo de processos, o qual fornece um resumo geral do fluxo básico e das interações entre os grupos de processos e as partes interessadas específicas. Os processos de gerenciamento de projetos estão interligados por entradas e saídas definidas, de forma que o resultado de um processo se torna a entrada de outro, mesmo que não pertençam ao mesmo grupo de processos.

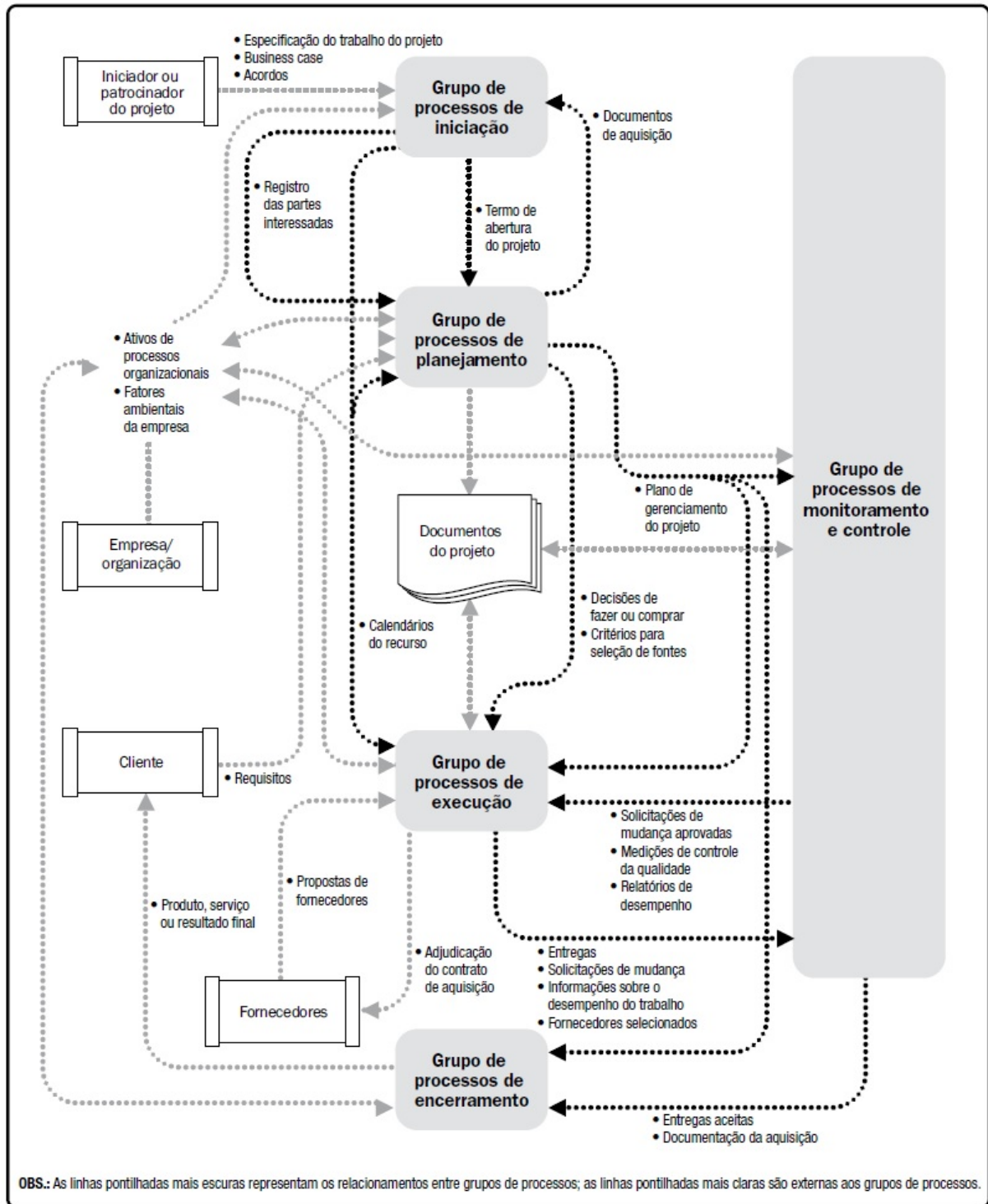


Figura 5 – Interações nos processos de gerenciamento de projetos.

Fonte: (PMI, 2013).

Apresenta-se, então, uma breve descrição de cada grupo de processos (PMI, 2013):

- Grupo de processos de iniciação: compreende os processos executados para definir um novo projeto ou uma nova fase de um projeto existente, por meio da obtenção de autorização formal para seu início. A execução desses processos no início de cada fase

contribui para manter o foco nos objetivos para os quais o projeto foi criado. A autorização é formalizada com a aprovação do termo de abertura, o que oficializa o início do projeto. O principal objetivo desse grupo de processos é alinhar os objetivos do projeto com as expectativas das partes interessadas, fornecendo uma visão geral do escopo e das metas a serem alcançadas.

- Grupo de processos de planejamento: compreende os processos necessários para estabelecer o escopo, definir e refinar os objetivos do projeto, e desenvolver o curso de ações necessário para alcançá-los. O plano de gerenciamento do projeto e os documentos gerados como saídas deste grupo abrangem todos os aspectos relevantes: escopo, prazo, qualidade, comunicações, recursos humanos, riscos, aquisições e o gerenciamento das partes interessadas. O principal benefício desse grupo de processos é delinear a estratégia, a tática e o caminho a ser seguido para garantir a conclusão bem-sucedida do projeto ou da fase.

- Grupo de processos de execução: abrange os processos realizados para executar o trabalho definido no planejamento, com o objetivo de atender às especificações do projeto. Durante essa fase, os resultados obtidos podem exigir atualizações no planejamento, além de sofrer alterações em aspectos como a duração prevista das atividades, a produtividade, a disponibilidade de recursos e a ocorrência de riscos imprevistos.

- Grupo de processos de monitoramento e controle: inclui os processos responsáveis por acompanhar e analisar o progresso e o desempenho do projeto, com o objetivo de identificar, sempre que necessário, mudanças no plano de gerenciamento. O principal benefício desse grupo é possibilitar a medição e análise do desempenho do projeto em intervalos regulares, em ocorrências apropriadas ou em situações excepcionais, a fim de identificar variações em relação ao plano original. Com base nesse monitoramento, a equipe do projeto passa a ter uma visão mais clara da situação geral, podendo identificar áreas que exijam atenção adicional e implementar ações corretivas ou preventivas, assegurando o cumprimento do plano estabelecido.

- Grupo de processos de encerramento: compreende os processos executados para concluir todas as atividades de todos os grupos de processos de gerenciamento do projeto, encerrando formalmente o projeto, a fase ou os contratos associados. Em alguns casos, esses processos também formalizam o encerramento prematuro do projeto, como nos casos de cancelamento, interrupção ou situações críticas que inviabilizam sua continuidade. Quando não é possível finalizar formalmente determinados contratos, pode haver a transferência dessas responsabilidades para outras unidades organizacionais, que providenciarão a finalização das atividades pendentes.

Os grupos de processos não correspondem a fases do ciclo de vida do projeto, embora todos eles possam ser conduzidos dentro de uma única fase (PMI, 2013). À medida que os projetos são divididos em fases ou subcomponentes distintos, como desenvolvimento

do conceito, estudo de viabilidade, concepção, prototipagem, construção, testes, entre outros, os grupos de processos tendem a ser repetidos em cada uma dessas etapas.

2.2.1 Ciclos de vida de um projeto

Considerando o início até o término, um projeto passa por diversas fases, conhecidas como ciclo de vida do projeto. Esse ciclo pode ser documentado em uma metodologia, sendo definido ou adaptado conforme as particularidades da organização, do setor ou da tecnologia empregada, oferecendo assim uma estrutura básica para o gerenciamento do projeto.

Os projetos variam em tamanho e complexidade, mas podem ser mapeados em uma estrutura genérica de ciclo de vida composta pelas seguintes fases: início do projeto, organização e preparação, execução do trabalho do projeto e encerramento do projeto. Essa estrutura pode ser visualizada na figura a seguir:

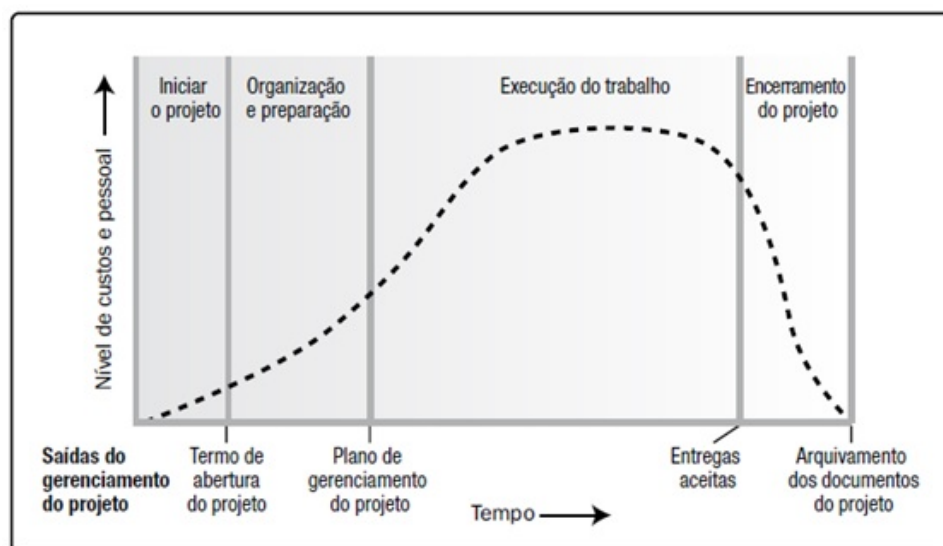


Figura 6 – Estrutura genérica do ciclo de vida de um projeto.

Fonte: (PMI, 2013).

Segundo o (PMI, 2013), a estrutura genérica do ciclo de vida é amplamente utilizada para facilitar a comunicação com a alta administração e demais partes interessadas que não estão familiarizadas com os detalhes técnicos do projeto. Por oferecer uma visão de alto nível, essa estrutura permite a comparação entre projetos distintos, mesmo que não sejam semelhantes entre si. Além disso, considera a fase atual do ciclo de vida do produto e apresenta as seguintes características:

- Os níveis de custo e de alocação de pessoal em um projeto são limitados no início, atingem seu pico durante a execução e diminuem rapidamente à medida que o projeto se aproxima da conclusão, como ilustrado na figura acima. No entanto, essa curva típica pode não se aplicar a todos os projetos. Em alguns casos, há a necessidade de

despesas substanciais já nas fases iniciais para garantir os recursos necessários, ou mesmo a presença de uma equipe completa desde o início do ciclo de vida.

Agora, considerando o tempo decorrido no projeto, é possível analisar outras características (PMI, 2013):

- No início do projeto, os riscos e as incertezas costumam ser mais elevados, tendendo a diminuir ao longo do ciclo de vida à medida que decisões são tomadas e entregas são aceitas. Nesse mesmo estágio inicial, a capacidade de influenciar as características finais do produto, sem causar impactos significativos nos custos, é maior. Contudo, conforme o projeto avança, essa capacidade diminui, enquanto os custos associados a mudanças e correções de erros tendem a aumentar de forma significativa, especialmente nas fases finais.

Apesar de estarem presentes em praticamente todos os ciclos de vida de projetos, essas características podem variar significativamente em intensidade. Os ciclos de vida adaptativos, por exemplo, são estruturados para preservar uma participação ativa das partes interessadas e manter os custos de mudanças relativamente baixos ao longo de todo o projeto, diferentemente dos ciclos de vida mais previsíveis, nos quais essa participação tende a diminuir e os custos aumentam progressivamente (PMI, 2013).

Os ciclos de vida de projetos se distribuem ao longo de um espectro contínuo, que vai desde abordagens previsíveis e orientadas por planejamento, até abordagens adaptativas, caracterizadas pela flexibilidade e resposta a mudanças.

2.2.1.1 Ciclo de vida previsível

No ciclo de vida previsível, o produto e as entregas são definidos no início do projeto, e quaisquer mudanças no escopo são cuidadosamente gerenciadas. Esses ciclos também são denominados ciclos de vida inteiramente planejados, como ilustrado na figura a seguir.

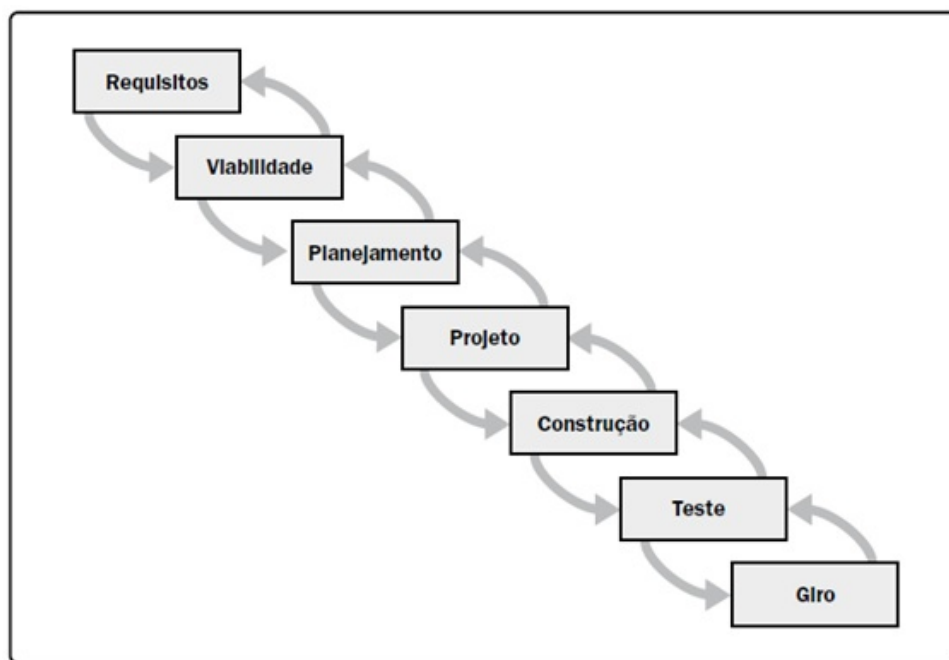


Figura 7 – Exemplo de ciclo de vida previsível.

Fonte: (PMI, 2013).

Esses projetos progridem por meio de uma série de fases sequenciais ou sobrepostas, nas quais cada fase geralmente foca em um subconjunto de atividades do projeto e processos de gerenciamento do projeto (PMI, 2013).

Quando o projeto é iniciado, a equipe concentra-se em definir o escopo geral do produto e do projeto, desenvolvendo um plano de entrega do produto (e das entregas associadas), e, em seguida, dá prosseguimento às fases para a execução do plano dentro desse escopo. As mudanças no escopo do projeto são meticulosamente gerenciadas, exigindo replanejamento e a aceitação formal do novo escopo.

Ciclos de vida previsíveis são geralmente preferidos quando o produto a ser entregue é bem compreendido, há uma base consolidada de práticas na indústria ou quando o valor para as partes interessadas depende da entrega completa do produto (PMI, 2013).

2.2.1.2 Ciclo de vida adaptativo

Ciclos de vida adaptativos, também conhecidos como orientados à mudança ou baseados em métodos ágeis, são projetados para responder a altos níveis de mudança e ao envolvimento contínuo das partes interessadas. Projetos que seguem esse modelo geralmente executam diversos processos em cada iteração, embora as primeiras possam ter um foco maior nas atividades de planejamento (PMI, 2013).

O produto é desenvolvido por meio de múltiplas iterações, sendo que um escopo detalhado é definido para cada uma delas apenas no início de sua execução. O escopo

geral do projeto pode ser desdobrado em um conjunto de requisitos e tarefas a serem executadas, comumente denominado *backlog* do projeto.

No início de uma iteração, a equipe trabalhará para determinar a quantidade de itens altamente prioritários da lista de *backlog* que podem ser entregues na próxima iteração. No final de cada iteração, o produto deve estar pronto para a análise pelo cliente. Os representantes do patrocinador e do cliente devem estar continuamente envolvidos no projeto para fornecer o *feedback* sobre as entregas à medida que elas são criadas, a fim de garantir que o *backlog* do produto reflitam suas necessidades atuais.

Os métodos adaptativos são geralmente preferidos em contextos de rápida mudança, quando os requisitos e o escopo são difíceis de definir antecipadamente e há possibilidade de implementar pequenas melhorias incrementais que agreguem valor às partes interessadas (PMI, 2013).

2.3 Desenvolvimento de software com modelos ágeis

Durante anos, a engenharia de software se inspirou em processos de manufatura para consolidar seus métodos de trabalho, buscando em setores emergentes da indústria da época grande parte das teorias e dos métodos de produção. Para a construção da nova indústria de TI, o papel da indústria automobilística foi de grande importância, com os modelos de produção em série de Henry Ford, amplamente influenciados por Frederick Taylor. Todo o pensamento tradicional da ciência do desenvolvimento de software evoluiu com um novo foco: a padronização de componentes e processos, e a mecanização do movimento (PRIKLADNICKI; WILLI; MILANI, 2014).

Em meados dos anos 90, começaram a surgir processos alternativos de desenvolvimento de software, distintos dos tradicionais, que eram considerados regradados, lentos, burocráticos e inadequados para a natureza da atividade. Esses novos processos foram apelidados de “leves” (*Lightweight*), em oposição aos anteriores, “pesados” (*Heavyweight*). Ambos os tipos de processo eram baseados no desenvolvimento iterativo, no qual requisitos e soluções evoluíam por meio da colaboração entre equipes auto-organizadas, favorecendo o alinhamento entre o desenvolvimento e os objetivos do cliente, e permitindo entregas rápidas e de alta qualidade por meio de um conjunto de boas práticas de engenharia (PRIKLADNICKI; WILLI; MILANI, 2014).

Essas metodologias passaram a ser chamadas de ágeis apenas após 2001, quando um grupo composto por 17 especialistas (desenvolvedores de software, autores e consultores de renome) se reuniu na estação de esqui *Snowbird*, em Utah, nos Estados Unidos, para discutir formas de desenvolver software de maneira mais leve, rápida e centrada nas pessoas. Eles assinaram o “Manifesto para o Desenvolvimento Ágil de Software” (*Manifesto for Agile Software Development*) e criaram o Manifesto Ágil (PRIKLADNICKI;

WILLI; MILANI, 2014).

Os métodos ágeis têm como principal característica a utilização de uma abordagem incremental, tanto para a especificação do sistema quanto para o desenvolvimento e entrega, permitindo que os softwares sejam rapidamente entregues e facilmente adaptáveis a novos requisitos (SOMMERVILLE, 2011).

O Manifesto Ágil possui quatro valores que defendem (PRESSMAN; MAXIM, 2021):

1. Indivíduos e interações acima de processos e ferramentas.
2. Software operacional acima de documentação completa.
3. Colaboração dos clientes acima de negociação contratual.
4. Respostas a mudanças acima de seguir um plano.

Além dos quatro valores representados, o Manifesto Ágil também estabelece doze princípios que os complementam, formando os pilares sobre os quais são construídos os chamados Métodos Ágeis (PRIKLADNICKI; WILLI; MILANI, 2014), conforme apresentado a seguir:

1. Nossa maior prioridade é satisfazer ao cliente com entregas contínua e adiantada de software com valor agregado.
2. Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Os processos ágeis tiram vantagem das mudanças, visando à vantagem competitiva para o cliente.
3. Entregar frequentemente software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
4. Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
5. Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessários e confie neles para realizar o trabalho.
6. O método mais eficiente e eficaz de transmitir informação para a equipe e entre a equipe de desenvolvimento é a conversa frente a frente.
7. Software funcional é a medida primária de progresso.
8. Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante sempre.
9. Contínua atenção à excelência técnica e bom design aumenta a agilidade.
10. Simplicidade, a arte de maximizar a quantidade de trabalho não realizado, é essencial.
11. As melhores arquiteturas, requisitos e design emergem de times auto-organizáveis.
12. Em intervalos regulares, o time reflete sobre como se tornar mais eficaz e então refina

e ajusta seu comportamento de acordo.

As ideias fundamentais que orientam o desenvolvimento ágil deram origem aos métodos ágeis, concebidos para sanar fraquezas, reais ou percebidas, da engenharia de software tradicional. Embora o desenvolvimento ágil proporcione diversos benefícios, sua aplicação pode não ser adequada em todos os contextos. Em mercados sujeitos a rápidas transformações, torna-se difícil ou até impossível prever a evolução de um sistema computacional ao longo do tempo. As necessidades dos usuários mudam constantemente, e novas ameaças competitivas podem surgir inesperadamente. Nessas condições, definir os requisitos de forma precisa antes do início do projeto é uma tarefa complexa, a menos, que se adote uma abordagem suficientemente ágil para responder de maneira eficaz a um ambiente de negócios dinâmico. Essa fluidez implica mudanças, e mudanças mal gerenciadas podem representar custos elevados ([PRESSMAN; MAXIM, 2021](#)).

No contexto da engenharia de software, a agilidade vai além da capacidade da equipe de engenharia em responder e assimilar mudanças rápidas. Ela promove a organização estruturada do trabalho em equipe, favorecendo uma comunicação mais eficiente, enfatizando a entrega ágil de software funcional e reduzindo a dependência de artefatos intermediários. Para que essa agilidade seja alcançada, o processo de desenvolvimento deve ser projetado de forma a permitir que a equipe se adapte continuamente e alinhe suas atividades aos objetivos do projeto ([PRESSMAN; MAXIM, 2021](#)).

Em contextos ágeis, a constante evolução e complexidade do desenvolvimento requerem uma abordagem ativa e cíclica para o gerenciamento de riscos. Nesse cenário, as metodologias ágeis se mostram eficientes em todas as fases do processo de gestão de riscos, pois permitem uma adaptação rápida às alterações e respostas frequentes aos *feedbacks*, assegurando maior flexibilidade e dinamismo na condução do projeto.

Segundo ([SOUZA et al., 2021](#)), as organizações passaram a incorporar o uso de projetos ágeis com o objetivo de alcançar metas estratégicas, superando as limitações da perspectiva tradicional de eficiência e respondendo às complexidades do novo cenário por meio da adoção de uma estrutura orientada a projetos. Para enfrentar esse desafio, empresas em busca de expansão e aprimoramento de seus negócios passaram a adotar as técnicas do Gerenciamento Ágil de Projetos (GAP).

Inicialmente, os métodos ágeis foram desenvolvidos com foco no desenvolvimento de software, mas com o tempo passaram a despertar o interesse de empresas que buscavam lançar produtos inovadores e competitivos em prazos reduzidos. Os métodos tradicionais de gerenciamento de projetos, por demandarem um planejamento detalhado antes da execução, acabavam por dificultar esse modelo. Nesse contexto, a agilidade passou a permitir que os projetos organizacionais fossem reconfigurados com flexibilidade, possibilitando a realocação de recursos e adaptações no ambiente conforme as necessidades ([SOUZA et](#)

al., 2021).

A grande quantidade de projetos demandada pelas empresas às suas áreas de projetos pode dificultar o cumprimento dos prazos de entrega e o acompanhamento contínuo das equipes, especialmente quando são utilizados métodos tradicionais de gerenciamento mais rígidos, como o modelo Cascata. A gestão de projetos por meio de métodos ágeis, com sua estrutura leve, dinâmica e adaptável ao longo da execução, tornou-se um dos principais fatores responsáveis pelo sucesso dessa abordagem nas organizações (SOUZA et al., 2021).

2.4 Metodologia do Scrum

2.4.1 Origem

Em 1986, os fundamentos iniciais do *framework* Scrum foram apresentados em um artigo de Hirotaka Takeuchi e Ikujiro Nonaka, publicado na *Harvard Business Review*, intitulado *The New Product Development Game* (O Novo Jogo de Desenvolvimento de Novos Produtos). Os autores traçaram analogias entre equipes de *rugby* de alto desempenho e equipes multidisciplinares, destacando que grupos pequenos e autônomos alcançavam melhores resultados. Essas observações serviram como base conceitual para a criação do Scrum (PRIKLADNICKI; WILLI; MILANI, 2014).

Anos depois, em 1993, na empresa *Easel Corporation*, Jeff Sutherland, John Scumniotales e Jeff McKenna idealizaram, documentaram e implementaram o Scrum, incorporando os estilos de gerenciamento observados por Takeuchi e Nonaka (1986). Assim nasceu o “Scrum”, nome originado de uma das jogadas mais famosas do *rugby*, na qual os jogadores reiniciam uma jogada juntos, empurrando-se com o objetivo de retomar a posse da bola. Dois anos depois, Jeff Sutherland colaborou com Ken Schwaber para formalizar o processo para a indústria mundial de software, com a publicação do primeiro artigo sobre Scrum, intitulado “*SCRUM Development Process*”, na conferência OOPSLA (1995) (PRIKLADNICKI; WILLI; MILANI, 2014).

2.4.2 Definições

O *framework* Scrum é definido como um *framework* ágil que auxilia no gerenciamento de projetos complexos, prescrevendo um conjunto de práticas leves e objetivas, sendo especialmente indicado para projetos de diferentes áreas que enfrentam prazos apertados e requisitos mutáveis. No desenvolvimento de software, essas características são comuns, pois as tecnologias utilizadas estão em constante evolução, e o trabalho a ser realizado deve estar alinhado com essa dinâmica do mercado. Desde então, o Scrum tem se

consolidado como uma das metodologias mais adotadas no mundo ágil ([PRIKLADNICKI; WILLI; MILANI, 2014](#)).

Essa metodologia pode ser dividida em três níveis principais: planejamento (*planning, backlog*), execução por ciclos (*sprint*) e fechamento (*closure*). Cada um desses níveis representa uma etapa fundamental na condução do projeto ágil. A Figura a seguir ilustra essa estrutura:

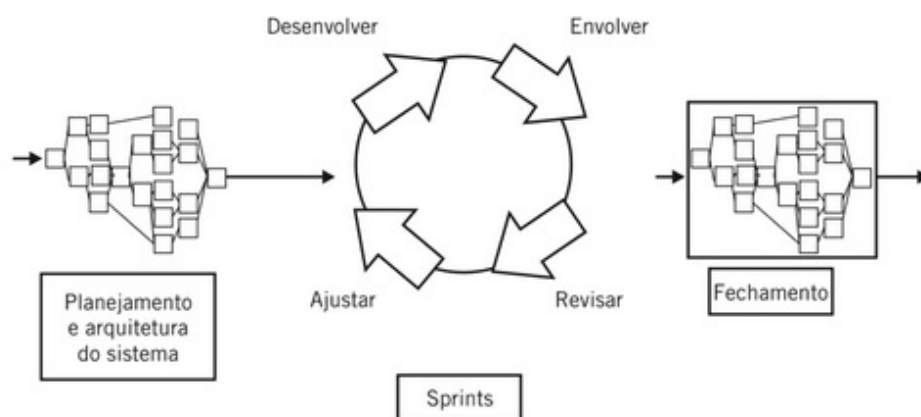


Figura 8 – O *framework* do Scrum.

Fonte: ([PRIKLADNICKI; WILLI; MILANI, 2014](#)).

Essa estrutura representa a origem do *framework* Scrum e o funcionamento de seu fluxo de trabalho. A primeira fase (*Planning*) corresponde à definição das entradas do processo, enquanto a última fase (*Closure*) estabelece as saídas. Ambas apresentam um fluxo linear bem definido, com possíveis iterações no momento do planejamento. Já a fase intermediária, correspondente aos *Sprints*, é caracterizada por um ciclo iterativo e empírico que envolve o desenvolvimento, envolvimento dos participantes, ajustes e revisões contínuas ([PRIKLADNICKI; WILLI; MILANI, 2014](#)).

O trabalho permanece “em andamento” até a fase de fechamento (*Closure*). Durante todo o processo, as necessidades do negócio são consideradas de forma criteriosa, assegurando que o produto desenvolvido seja capaz de responder a influências externas. Isso é possível porque as entregas podem ser ajustadas a qualquer momento. Para viabilizar essa flexibilidade, o trabalho é estruturado em ciclos iterativos (*sprints*) ([PRIKLADNICKI; WILLI; MILANI, 2014](#)).

As *sprints* recebem esse nome por fazerem alusão a corridas de curta distância, representando ciclos breves e intensivos de trabalho. Cada *sprints* corresponde a um intervalo de tempo previamente definido, durante o qual os itens do *sprint backlog* devem ser desenvolvidos. Esses ciclos ocorrem de forma iterativa e, ao final de cada um, o *sprint backlog* é revisado e ajustado conforme necessário ([PRESSMAN; MAXIM, 2021](#)).

Em cada *sprint*, o trabalho é conduzido com base em uma lista priorizada de requisitos, denominada *backlog* do produto. Os *backlogs* funcionam como registros das tarefas pendentes, organizadas por ordem de prioridade. Essa abordagem, centrada na priorização dos requisitos, assegura que a equipe trabalhe nos itens de maior valor e urgência para o cliente. Ao final de cada *sprint*, é entregue um conjunto de funcionalidades concluídas e prontas para uso (PRIKLADNICKI; WILLI; MILANI, 2014).

O Scrum adota uma abordagem empírica, reconhecendo que o problema pode não estar totalmente definido ou compreendido durante a fase de análise, e que os requisitos tendem a mudar ao longo do tempo. Dessa forma, mantém o foco em maximizar a capacidade da equipe de responder de maneira ágil aos obstáculos que surgem durante o projeto (PRIKLADNICKI; WILLI; MILANI, 2014).

Em projetos que utilizam o *framework* Scrum, destacam-se três papéis, três artefatos e quatro cerimônias (ou eventos).

2.4.2.1 Papéis da equipe Scrum

A equipe Scrum é interdisciplinar e auto-organizada, composta por três papéis principais: o *product owner*, o *scrum master* e o *development team*, cada um com responsabilidades específicas dentro do processo.

O dono do produto (*Product Owner*) é responsável por maximizar o valor do produto, definindo as funcionalidades prioritárias e decidindo o que será desenvolvido em cada *sprint* (PRIKLADNICKI; WILLI; MILANI, 2014).

O mestre de Scrum (*Scrum Master*) é o membro da equipe que possui maior conhecimento sobre o *framework* e atua como facilitador, apoiando os demais papéis na adoção e aplicação correta do Scrum, promovendo melhorias nos processos e removendo impedimentos que afetem o desempenho da equipe (PRESSMAN; MAXIM, 2021).

A equipe de desenvolvimento (*Development Team*) é composta, em geral, por três a seis integrantes que se auto-organizam com o objetivo de entregar valor continuamente, desenvolvendo software com qualidade (PRIKLADNICKI; WILLI; MILANI, 2014).

2.4.2.2 Artefatos do Scrum

O Scrum define três artefatos principais que fornecem visibilidade sobre o progresso do projeto e das *sprints*: *product backlog*, *sprint backlog* e *increments*.

O *backlog* do produto (*Product Backlog*) é uma lista atualizada dos requisitos desejados para o produto, criada pela equipe Scrum e gerenciada exclusivamente pelo dono do produto, que pode inserir, remover ou reordenar os itens conforme necessário (PRIKLADNICKI; WILLI; MILANI, 2014).

O *backlog* da *sprint* (*Sprint Backlog*) representa o planejamento estratégico e tático da próxima *sprint* em um nível mais detalhado, tornando visível o trabalho necessário para que a equipe de desenvolvimento atinja a meta. Durante a *sprint*, tarefas podem ser adicionadas ou removidas conforme a necessidade (PRIKLADNICKI; WILLI; MILANI, 2014).

Os incrementos (*Increments*) são representados pelas funcionalidades de software prontas. Ao final de cada *sprint*, a equipe de desenvolvimento entrega um incremento do produto, ou seja, o resultado do que foi produzido durante a *sprint* (PRIKLADNICKI; WILLI; MILANI, 2014).

2.4.2.3 Eventos do Scrum

O Scrum define quatro cerimônias (ou eventos) principais que estruturam o trabalho durante uma *sprint*. Esses eventos são realizados com duração fixa e intervalos regulares permitindo o acompanhamento, a adaptação e a melhoria contínua do processo. São eles: *sprint planning*, *daily scrum*, *sprint review* e *sprint retrospective*.

A reunião de planejamento da *sprint* (*Sprint Planning*) é onde a meta da *sprint* é definida e planejada. A reunião tem uma duração fixa de até oito horas para uma *sprint* de um mês, e para *sprints* menores, o tempo é reduzido de forma proporcional. Este encontro é dividido em duas partes, cada uma com duração fixa correspondente à metade da duração total, em que cada parte responde às seguintes perguntas: “O que será entregue no incremento resultante nesta *sprint*?” e “Como faremos para entregar o incremento nesta *sprint*?” (PRIKLADNICKI; WILLI; MILANI, 2014).

As reuniões diárias do Scrum (*Daily Scrum*) elevam o nível de auto-organização da equipe de desenvolvimento. Elas acontecem por no máximo 15 minutos, onde cada membro responde às seguintes perguntas: “O que fiz desde a última Scrum diária?”, “O que pretendo fazer até a próxima Scrum diária?” e “Existe algo me impedindo de concluir alguma tarefa?” (PRIKLADNICKI; WILLI; MILANI, 2014).

Na revisão da *sprint* (*Sprint Review*), o produto e o projeto são inspecionados e adaptados. Geralmente, a revisão é limitada a uma reunião de 4 horas para um *sprint* de 4 semanas. A atividade principal consiste na demonstração do incremento de software concluído durante o *sprint*, que pode não incluir toda a funcionalidade planejada, mas sim as funções que seriam entregues dentro da janela de tempo estipulada (PRESSMAN; MAXIM, 2021).

A retrospectiva da *sprint* (*Sprint Retrospective*) é o momento dedicado à inspeção e adaptação dos processos empíricos. Normalmente com duração de 3 horas para um *sprint* de 4 semanas, a equipe de desenvolvimento debate questões como: “O que deu certo no *sprint*?”, “O que poderia melhorar?”, “Com o que a equipe se compromete em melhorar

no próximo *sprint*?”(PRESSMAN; MAXIM, 2021).

2.4.3 Justificativa para a adoção do Scrum

Conforme discutido por (PRESSMAN; MAXIM, 2021), diversas abordagens ágeis são aplicadas no desenvolvimento de software, entre elas Kanban, *Extreme Programming* (XP) e Lean. Cada uma apresenta características específicas que se adaptam melhor a determinados contextos. O Kanban, por exemplo, é eficaz para fluxos contínuos de trabalho, enquanto o XP foca na excelência técnica e exige elevada maturidade da equipe. Já o Lean atua como uma filosofia focada na eliminação de desperdícios e otimização de processos.

O Scrum, por sua vez, apresenta uma estrutura iterativa clara, com papéis, eventos e artefatos bem definidos, proporcionando organização e clareza na gestão de projetos. Sua estrutura permite maior adaptabilidade a mudanças e facilita o acompanhamento contínuo da entrega de valor ao cliente (PRESSMAN; MAXIM, 2021).

A antiga *Version One*, atualmente denominada Digital.ai, é uma empresa especializada no desenvolvimento de ferramentas para a gestão de projetos ágeis. Anualmente, ela publica o relatório “*State of Agile*”, uma pesquisa que analisa a adoção e percepção das práticas ágeis no cenário global.

De acordo com o 17º Relatório *State of Agile* (DIGITAL.AI, 2023), o Scrum permanece como a metodologia ágil mais popular em nível de equipe, sendo que 63% dos usuários de *Agile* são equipes Scrum, seja de forma pura ou combinada com outras abordagens. Essa ampla adesão evidencia sua eficácia e aplicabilidade em diferentes cenários organizacionais.

Além disso, conforme argumenta (SUTHERLAND, 2014), o Scrum é particularmente eficiente em contextos complexos, nos quais os requisitos do projeto são incertos ou estão em constante evolução. Sua simplicidade conceitual, aliada à robustez de suas práticas, contribui para sua ampla adoção tanto no mercado quanto no meio acadêmico.

Assim, a escolha do Scrum como referência neste trabalho justifica-se por sua prevalência no setor de tecnologia, por sua aplicabilidade prática e por oferecer uma base sólida para a análise de riscos e a estruturação do modelo proposto.

2.5 Gestão de riscos em projetos de TI

O risco de um projeto pode ser definido como um evento ou condição incerta que provocará um efeito, seja positivo ou negativo, em um ou mais objetivos do projeto, como escopo, cronograma, custo e qualidade (PMI, 2013). No Guia (ISO, 2009), risco é descrito como a “combinação da probabilidade de um evento e sua consequência”. Esta

definição amplamente utilizada é aplicada no principal padrão de engenharia de software para gestão de riscos (IEEE, 2006).

A gestão de riscos é um processo essencial para o sucesso de um projeto. Ela envolve os processos de planejamento, identificação, análise, planejamento de resposta e controle de riscos. O objetivo principal desse gerenciamento é aumentar a probabilidade e o impacto de eventos positivos, ao mesmo tempo em que diminui a probabilidade e o impacto de eventos negativos no projeto (PMI, 2013).

A Figura a seguir apresenta uma visão geral desse gerenciamento:



Figura 9 – Visão geral do gerenciamento do risco do projeto.

Fonte: Adaptado de (PMI, 2013).

2.5.1 Planejar o gerenciamento dos riscos

O planejamento do gerenciamento dos riscos define como conduzir as atividades de gerenciamento de riscos de um projeto. Seu benefício é garantir que o grau, tipo e visibilidade do gerenciamento de riscos sejam proporcionais aos riscos e à importância do projeto para a organização. A saída desse processo é o plano de gerenciamento de riscos, que é vital para a comunicação e para garantir o apoio das partes interessadas na execução eficaz do projeto (PMI, 2013).

2.5.2 Identificar os riscos

Identificar os riscos é o processo de determinar os riscos que podem afetar o projeto e documentar suas características. A documentação dos riscos e o conhecimento adquirido proporcionam à equipe do projeto a capacidade de antecipar eventos. Para isso, são realizadas análises baseadas nas informações de entrada dos planos de: riscos, custos, cronograma, qualidade e recursos humanos, com o objetivo de gerar, de forma precisa, o registro dos riscos (PMI, 2013).

2.5.3 Realizar a análise qualitativa dos riscos

Realizar a análise qualitativa dos riscos é o processo de priorização dos riscos para análise ou ação adicional, por meio da avaliação e combinação de sua probabilidade de

ocorrência e impacto. Dessa forma, os gerentes de projeto conseguem reduzir o nível de incerteza e concentrar-se nos riscos de alta prioridade (PMI, 2013).

Os riscos podem ser priorizados para uma análise quantitativa posterior e para o planejamento de respostas, com base na sua classificação. Essa classificação é determinada pela avaliação da probabilidade e do impacto de cada risco. A importância de cada risco e a prioridade de atenção são frequentemente avaliadas por meio de uma tabela de referência ou uma matriz de probabilidade e impacto, que especifica as combinações de probabilidade e impacto, resultando em classificações de risco como baixa, moderada ou alta prioridade (PMI, 2013).

2.5.4 Realizar a análise quantitativa dos riscos

Realizar a análise quantitativa dos riscos é o processo que avalia numericamente o impacto dos riscos identificados nos objetivos gerais do projeto, gerando informações quantitativas que auxiliam na tomada de decisões e ajudam a reduzir o grau de incerteza do projeto (PMI, 2013).

O processo de realizar a análise quantitativa dos riscos avalia o impacto dos riscos nos objetivos do projeto, sendo executado nos riscos priorizados pelo processo de análise qualitativa, que são considerados como tendo um impacto potencial significativo nas demandas concorrentes do projeto. Esse processo é utilizado principalmente para avaliar o efeito agregado de todos os riscos que afetam o projeto (PMI, 2013).

As técnicas comumente utilizadas para análise de riscos incluem abordagens orientadas ao evento e ao projeto, como: análise de sensibilidade (que determina quais riscos têm maior impacto potencial no projeto), análise do valor monetário esperado (um conceito estatístico que calcula o resultado médio quando o futuro envolve cenários que podem ou não ocorrer), além de modelagem e simulação (PMI, 2013).

2.5.5 Planejar as respostas aos riscos

Planejar as respostas aos riscos é o processo de desenvolvimento de opções e ações para maximizar as oportunidades e reduzir as ameaças aos objetivos do projeto. O benefício desse processo é abordar os riscos conforme sua prioridade, alocando recursos e atividades no orçamento, cronograma e plano de gerenciamento do projeto, quando necessário. Para cada risco, deve-se selecionar a estratégia, ou uma combinação de estratégias, com maior probabilidade de sucesso. Ferramentas de análise de riscos, como a análise da árvore de decisão, podem ser utilizadas para escolher as respostas mais adequadas (PMI, 2013).

No planejamento das respostas aos riscos, serão consideradas duas formas de risco: os riscos negativos, também conhecidos como ameaças, e os riscos positivos, chamados de

oportunidades (PMI, 2013).

2.5.5.1 Estratégias para riscos negativos ou ameaças

Estratégias para riscos negativos ou ameaças: existem quatro abordagens principais para lidar com esses riscos, que podem causar impactos negativos nos objetivos do projeto, caso ocorram. São elas: prevenir, transferir, mitigar e aceitar (PMI, 2013).

Na estratégia de prevenir, a equipe do projeto atua para eliminar a ameaça ou proteger o projeto contra seu impacto, podendo, para isso, alterar o plano de gerenciamento do projeto a fim de eliminar totalmente o risco. Na estratégia de transferir, o impacto de uma ameaça é repassado a terceiros, juntamente com a responsabilidade pela sua resposta. Já na estratégia de mitigar, a equipe do projeto toma medidas antecipadas para reduzir a probabilidade de ocorrência ou o impacto do risco antes que ele se concretize (PMI, 2013).

Já a última estratégia, aceitar, pode ser aplicada tanto a riscos negativos (ameaças) quanto a riscos positivos (oportunidades). Nessa abordagem, a equipe do projeto reconhece a existência do risco, mas decide não agir, a menos que ele de fato ocorra. A aceitação pode ser passiva ou ativa. A aceitação passiva não exige nenhuma ação além de documentar a estratégia e revisar periodicamente o risco para garantir que ele não se altere significativamente, deixando a equipe livre para lidar com o problema caso ele se concretize. Já a aceitação ativa geralmente envolve a criação de uma reserva de contingência, em termos de tempo, orçamento ou recursos, para ser utilizada caso o risco venha a se materializar (PMI, 2013).

Cada uma dessas estratégias de resposta ao risco exerce uma influência distinta sobre as condições dos riscos. A escolha da estratégia deve considerar a probabilidade de ocorrência e o impacto do risco nos objetivos do projeto. As estratégias de prevenção e mitigação são, em geral, mais adequadas para riscos críticos de alto impacto, enquanto as estratégias de transferência e aceitação tendem a ser mais apropriadas para ameaças menos críticas e de menor impacto (PMI, 2013).

2.5.5.2 Estratégias para riscos positivos ou oportunidades

Estratégias para riscos positivos ou oportunidades: existem quatro estratégias recomendadas para lidar com riscos que podem ter impactos potencialmente positivos nos objetivos do projeto. São elas: explorar, melhorar, compartilhar e aceitar (PMI, 2013).

Na estratégia de explorar, a organização busca garantir que a oportunidade se concretize, selecionando, por exemplo, os profissionais mais talentosos para o projeto, a fim de reduzir o tempo de conclusão, ou adotando novas tecnologias para diminuir custos e prazos. Já a estratégia de melhorar visa aumentar a probabilidade de ocorrência

e o impacto positivo da oportunidade, como ao alocar mais recursos a uma atividade para concluí-la mais rapidamente. Por fim, na estratégia de compartilhar, transfere-se total ou parcialmente a responsabilidade pela oportunidade a um terceiro que possua maior capacidade de aproveitá-la em benefício do projeto. Por fim, a estratégia de aceitar consiste em estar disposto a aproveitar uma oportunidade caso ela ocorra, sem, no entanto, buscá-la de forma ativa (PMI, 2013).

2.5.6 Controlar os riscos

Controlar os riscos é o processo de implementar os planos de resposta, acompanhar os riscos identificados, monitorar os riscos residuais, identificar novos riscos e avaliar a eficácia do processo de gerenciamento de riscos ao longo de todo o projeto. O principal benefício desse processo é aumentar a eficiência na abordagem dos riscos durante todo o ciclo de vida do projeto, buscando otimizar continuamente as respostas adotadas (PMI, 2013).

Os documentos do projeto podem ser atualizados como resultado do processo de controle dos riscos. Essas atualizações não se limitam ao registro de riscos, podendo incluir também os resultados de reavaliações, auditorias e revisões periódicas, bem como os resultados reais dos riscos do projeto e das respostas aplicadas (PMI, 2013).

3 Trabalhos Relacionados

Neste capítulo, apresenta-se uma revisão dos principais trabalhos relacionados à temática abordada. São analisados estudos que tratam da gestão de riscos e de métodos ágeis, abrangendo abordagens teóricas, metodológicas e resultados relevantes. A seguir, destacam-se as contribuições mais significativas para o desenvolvimento desta pesquisa e suas implicações para o campo de estudo.

3.1 Trabalho 1

Gestão do Conhecimento potencializando a Gestão de Riscos de Projetos de Desenvolvimento de Software Ágil utilizando o SCRUM (SILVA; ZAIDAN; BRAGA, 2023).

Este artigo se baseia em uma revisão de literatura, tendo como principais conceitos a gestão de projetos ágeis no desenvolvimento de software com o Scrum, além da gestão de riscos e gestão do conhecimento.

A pesquisa adota a metodologia DS (*Design Science*) e o método prescritivo DSR (*Design Science Research*), que tem como resultado a criação de um artefato: um modelo de gestão do conhecimento que potencializa a gestão de riscos no desenvolvimento de software com o Scrum.

Com base na revisão de literatura, foi realizada uma análise quantitativa por meio da aplicação de um questionário a um grupo de pessoas. Os resultados obtidos indicaram que todas as etapas da gestão do conhecimento podem impactar positivamente a gestão de riscos. Após a análise dos dados, foi desenvolvido um modelo focado na gestão do conhecimento e em como atuar na gestão de riscos em projetos ágeis de software. Esse modelo foi validado utilizando a técnica de grupo focal, abrangendo todos os papéis descritos pelo Scrum (SILVA; ZAIDAN; BRAGA, 2023).

3.2 Trabalho 2

O gerenciamento de riscos em projetos gerenciados por abordagens ágeis: uma revisão sistemática da literatura (FERNANDES; JR, 2021).

Este artigo apresenta uma análise exploratória e uma revisão da literatura sobre gerenciamento de risco e abordagens ágeis. O objetivo deste estudo foi investigar a evolução da relação entre gestão de riscos e métodos ágeis, buscando compreender os principais riscos e as características dos modelos de gerenciamento de riscos em projetos gerenciados por abordagens ágeis.

Os métodos de pesquisa utilizados para atingir os resultados deste artigo incluíram uma análise exploratória, por meio de um estudo bibliométrico (que permite identificar padrões, tendências e gerar visões sobre os conhecimentos produzidos em uma área específica, utilizando ferramentas de análise de citação e cocitação), e uma revisão sistemática da literatura (que sumariza o conhecimento existente sobre gerenciamento de riscos, com foco na aplicação desse tema em projetos gerenciados por abordagens ágeis, validando seu sucesso e mapeando os principais riscos envolvidos).

Este estudo contribui teoricamente ao explorar os temas relacionados ao gerenciamento de riscos em abordagens ágeis de gerenciamento de projetos. Na prática, demonstra a importância desse gerenciamento para o sucesso dos projetos ágeis, além de identificar, nesses projetos, os principais riscos e os modelos adotados para sua gestão ([FERNANDES; JR, 2021](#)).

3.3 Trabalho 3

Relação entre as boas práticas descritas no PMBOK e as Metodologias Ágeis para o Desenvolvimento de Softwares ([MACHADO; OLIVEIRA, 2022](#)).

Neste artigo, é apresentado um estudo baseado nos guias oficiais do PMBOK e do Scrum. Na fundamentação teórica, são explorados conceitos e técnicas com base no PMBOK. Em seguida, é feita a definição conceitual das metodologias ágeis, com foco no Scrum. A partir disso, estabelece-se um conjunto mínimo de boas práticas e de documentação para o gerenciamento de projetos ágeis, fundamentado no PMBOK.

Os métodos utilizados para alcançar os resultados finais foram organizados da seguinte forma: definição do objetivo, tipo e processo da pesquisa e, por fim, análise dos dados. Trata-se de um trabalho de natureza aplicada, com interesse prático na resolução de problemas reais. Possui caráter exploratório e, predominantemente, uma abordagem qualitativa, uma vez que grande parte dos resultados da pesquisa não pode ser traduzida em números, havendo uma relação dinâmica entre o objeto de estudo e o pesquisador.

O principal interesse da pesquisa foi compreender como a metodologia de gerenciamento de projetos PMBOK pode ser utilizada em conjunto com métodos ágeis, em especial com a metodologia Scrum. Para isso, foi realizado um comparativo entre o Scrum e o PMBOK, considerando seus processos, documentos, ferramentas e técnicas, com o objetivo de identificar características semelhantes ou diferenças na execução dos processos em ambas as abordagens. Segundo ([MACHADO; OLIVEIRA, 2022](#)), a inserção de boas práticas do Guia PMBOK nas lacunas do Scrum fortalece o gerenciamento de projetos ágeis, tornando viável a proposta de uso conjunto entre o Guia PMBOK e a metodologia Scrum.

3.4 Trabalho 4

Um survey sobre o uso de metodologias ágeis por *startups* do estado do Ceará (BANDEIRA, 2024)

O artigo apresenta uma análise sobre o uso de metodologias ágeis em *startups* de software, a partir da perspectiva de gestores que lideram equipes de desenvolvimento no estado do Ceará.

O método de pesquisa utilizado foi um *survey* do tipo transversal, no qual os participantes foram convidados a fornecer informações referentes a um momento específico no tempo. O questionário autoaplicável foi elaborado no Google Forms e distribuído pela internet, utilizando o LinkedIn como meio de contato com diversas *startups* da área.

A pesquisa teve como objetivo compreender os benefícios da aplicação de metodologias ágeis, como a otimização de processos, a melhoria no controle de prazos e um desenvolvimento mais eficaz, com menor incidência de erros nos softwares. Destacaram-se os métodos ágeis Scrum, Lean e Kanban como os mais adotados pelas *startups* participantes. A pesquisa também evidenciou os principais desafios enfrentados, como a dificuldade de adaptação dos colaboradores e a falta de experiência prévia com essas metodologias (BANDEIRA, 2024).

3.5 Trabalho 5

Managing Risks in Agile Methods: a Systematic Literature Mapping (GARCIA; HAUCK; HAHN, 2022)

Este artigo apresenta um mapeamento sistemático da literatura (SLM) sobre o gerenciamento de riscos em métodos ágeis, realizado ao longo de um período específico, com o objetivo de identificar como as organizações integram práticas explícitas de gerenciamento de riscos nessas abordagens. A maioria dos estudos primários analisados foi aplicada na indústria, buscando evidências de como a introdução dessas práticas pode trazer benefícios aos métodos ágeis.

A pesquisa teve como objetivo geral responder à seguinte questão: “Como as organizações de software integram práticas explícitas de gerenciamento de risco em métodos ágeis?”. Para isso, a questão principal foi desdobrada em quatro perguntas de investigação mais específicas: “Quais são os estudos que tratam da integração entre gerenciamento de risco e métodos ágeis?”, “Qual é o contexto de uso dessas práticas?”, “Quais práticas de gerenciamento de risco são introduzidas em métodos ágeis?” e “Quais tipos de riscos são gerenciados?”.

Com base nesse mapeamento, foi possível concluir que o método de pesquisa mais

utilizado é o estudo de caso, sendo que os estudos primários selecionados foram majoritariamente aplicados na indústria, com o uso do *framework* Scrum. Os riscos mais frequentemente identificados estavam relacionados ao cronograma e à comunicação. Dessa forma, constatou-se que a inclusão de práticas de gerenciamento de riscos em projetos ágeis trouxe resultados positivos, sem comprometer os princípios da agilidade, reduzindo impactos negativos e aumentando as chances de sucesso dos projetos (GARCIA; HAUCK; HAHN, 2022).

4 Aplicação de gestão de riscos em processo de desenvolvimento ágil

4.1 Contextualização

Conforme descrito na metodologia de pesquisa, neste capítulo será realizada a aplicação e análise da pesquisa de levantamento (*Survey*). O objetivo é explorar a dinâmica do *framework* Scrum na gestão de riscos, focando nas percepções, experiências e interpretações dos participantes, incluindo equipes de desenvolvimento, *scrum masters*, *product owners* e demais *stakeholders*. Para isso, busca-se avaliar o grau de concordância dos respondentes em relação às afirmações do questionário, por meio da escala Likert.

Pelo Scrum ser iterativo e incremental, é um método que facilita a adaptação às mudanças, permitindo que os riscos sejam identificados e mitigados de forma contínua e colaborativa.

Para (FERNANDES; JR, 2021), a gestão de riscos no contexto ágil pode envolver aspectos como riscos de escopo, riscos de recursos, riscos técnicos, riscos de qualidade e riscos de comunicação. Os riscos podem ser identificados durante as reuniões diárias, *sprint reviews* e retrospectivas, permitindo que as equipes ajustem o curso do projeto com agilidade.

Esta contextualização fundamenta a importância do estudo da gestão de riscos dentro do ambiente ágil, reforçando a relevância da pesquisa para compreender como esses processos são percebidos e aplicados por profissionais atuantes em projetos de desenvolvimento de software.

4.2 Método de pesquisa

De acordo com (WAZLAWICK, 2020), o método de pesquisa consiste na sequência de passos necessários para demonstrar que o objetivo proposto foi atingido. Assim, o presente trabalho será um método de pesquisa realizada por *survey* na gestão de riscos aplicado aos métodos ágeis no desenvolvimento de softwares com foco em abordagens quantitativas.

Segundo (MINEIRO, 2020), *Survey* é um termo comumente traduzido como “levantamento”, é um tipo de investigação com a finalidade de oferecer descrições estatísticas de pessoas por meio de perguntas buscando identificar padrões gerais aplicadas a uma amostra com abordagem quantitativa.

O objetivo do levantamento é produzir descrições, predominantemente quantitativas ou numéricas, sobre alguns aspectos de uma população, coletando dados por meio de perguntas feitas às pessoas. Tais dados são bases para posteriores análises.

A pesquisa quantitativa pode ser empregada para quantificar perfis populacionais, indicadores socioeconômicos, preferências, comportamentos dos indivíduos, entre outros, possibilitando ao pesquisador obter valores descritivos dos dados coletados que são alcançados por meio de análises e cálculos. Com uma abordagem de quantificação, ou seja, faz referência com dimensões de intensidade. Nesse sentido, o interesse do pesquisador se orienta por dimensionar, analisar e avaliar a aplicabilidade de recursos ou técnicas ou até mesmo introduzir uma variável na coleta de dados para um registro quantitativo (RODRIGUES; OLIVEIRA; SANTOS, 2021).

A investigação através de amostras reúne basicamente três passos: amostragem, formulação de perguntas aos sujeitos e coleta de dados. As perguntas devem ser objetivas, curtas, evitar colocar dois ou mais assuntos na mesma pergunta para não confundir o respondente, uso de questões fechadas em relação as questões abertas que permitem que o pesquisador obtenha respostas imprevistas. O uso de formulários é uma das maneiras mais simples de construir questões combinando as repostas em escala (MINEIRO, 2020).

No século 20 começa-se a formular questões padrão para mensurar fenômenos subjetivos. Em 1932, Rensis Likert elaborou técnicas de escala para medir fenômenos subjetivos, níveis de concordância (criando a chamada Escala Likert). A Escala Likert é um tipo de escala de resposta psicométrica usada em questionários, costumeiramente empregada em pesquisas de opinião, nela os perguntados especifica seu nível de concordância com uma afirmação (MINEIRO, 2020).

Para (MEIRELES, 2024), as respostas oferecerem uma escala de pontos como descrições verbais que contemplam extremos como: “Concordo totalmente” e “Discordo totalmente”, permitindo a descoberta de diferentes níveis de intensidade da opinião das pessoas a respeito de um mesmo assunto ou tema. Afirmativas são apresentadas e o respondente é convidado a emitir o seu grau de concordância com aquela frase marcando na escala a resposta que mais traduz sua opinião. Dentre as opções de respostas, pode-se considerar modelos comuns como a escala de 5 pontos, tendo assim:

- 1) Discordo totalmente
- 2) Discordo
- 3) Indiferente (ou Neutro)
- 4) Concordo
- 5) Concordo totalmente

Essas opções de resposta são conhecidas como diferencial semântico variando qualita-

tivamente em grau, do mais baixo ao mais elevado. Em geral, a Escala Likert deve conter um número ímpar e uma opção neutra (mantendo a simetria entre o número de categorias positivas em relação às categorias negativas e uma opção neutra) sobre uma questão do questionário.

4.3 Definição do processo

Foi elaborado um questionário de levantamento (*Survey*), com 25 perguntas, dividido em duas partes e estruturado no *Google Forms*. A primeira parte é composta por 5 perguntas fechadas e a segunda por 20 questões avaliadas por meio de uma escala Likert de 5 pontos, variando de “Discordo totalmente” a “Concordo totalmente”.

Na primeira parte do questionário, foram analisados os seguintes elementos:

1. Cargo/função na empresa;
2. Tamanho da empresa;
3. Setor de atuação (desenvolvimento, infraestrutura, segurança, análise de dados, IA);
4. Metodologias (Scrum, Kanban, SAFe, *Extreme programming* (XP), *Lean/lean Startup*);

A segunda parte do questionário foi elaborada com base nos capítulos anteriores e inspirada no artigo “*A Model for Assessing and Mitigating Knowledge Sharing Risks in Agile Software Development*” (GHOBADI; MATHIASSEN, 2016), cuja abordagem trata de aspectos críticos da gestão de conhecimento em contextos ágeis. Essa contribuição foi considerada pertinente por contemplar riscos recorrentes e aplicáveis à realidade dos profissionais entrevistados.

Apresentam-se a seguir alguns riscos relacionados ao desenvolvimento de software (GHOBADI; MATHIASSEN, 2016):

1. Falta de conhecimento de negócios.
2. Falta de familiaridade com desenvolvimento.
3. Requisitos ambíguos ou insuficientes.
4. Falta de familiaridade com princípios e valores ágeis.
5. Falta de recursos de TI.
6. Falta de motivação.
7. Disponibilidade e participação inadequadas do cliente.
8. O *Product Owner* não compartilha o *feedback* do cliente com a equipe de desenvolvimento.
9. Planejamento e organização inadequados nas práticas ágeis.
10. Planejamento inadequado e documentação insuficiente.

Os riscos apresentados foram adaptados e correlacionados genericamente às práticas ágeis, com o objetivo de possibilitar que um público mais amplo, sem conhecimento específico da metodologia Scrum (utilizada como referência em algumas afirmativas), pudesse responder ao questionário.

A escolha pela divisão do questionário em duas partes visa, inicialmente, à caracterização da empresa, fundamental para compreender o contexto sobre o perfil dos respondentes e a utilização de métodos ágeis na organização, possibilitando a correlação das respostas com variáveis contextuais. Em seguida, busca-se a obtenção de dados perceptivos acerca dos processos de gestão de riscos, por meio de uma escala Likert de 5 pontos, amplamente utilizada em pesquisas quantitativas para mensurar o grau de concordância dos respondentes de maneira objetiva e padronizada.

O questionário completo está disponível para avaliação no Apêndice A.

4.4 Aplicação

A aplicação do *Survey* ocorreu no mês de Abril de 2025. O questionário foi enviado de forma assíncrona via e-mail para os cursos de: Bacharelado em Ciência da Computação, Bacharelado em Sistemas de Informação dos Campus Santa Mônica e Campus Monte Carmelo, Bacharelado em Engenharia de Computação e Bacharelado em Gestão da Informação da Universidade Federal de Uberlândia, e para o Uberhub (iniciativa/comunidade que impulsiona o ecossistema de *startups* e inovação de Uberlândia). O foco do *survey* foi direcionado a profissionais que estão atuando ou estagiando em empresas de Tecnologia da Informação (TI) localizadas em Uberlândia e regiões vizinhas.

O questionário ficou disponível por aproximadamente duas semanas. Dentre um total estimado de 2.500 a 3.000 possíveis respondentes, foram obtidas 61 respostas nesse curto período.

4.5 Resultados

Com base nos dados coletados por meio do *Survey*, realizou-se uma análise estatística descritiva das respostas fornecidas pelos participantes. O objetivo foi identificar o grau de concordância dos respondentes em relação à gestão de riscos no uso de metodologias ágeis, no contexto dos projetos desenvolvidos nas empresas em que atuam.

Assim, para esta seção, das 25 perguntas do questionário, foram selecionadas as 16 mais relevantes. Dessas, 2 referem-se à caracterização da empresa (Apêndice A – Parte 1), e as outras 14 tratam do processo de gestão de riscos (Apêndice A – Parte 2), abrangendo

planejamento, identificação, análise qualitativa e quantitativa, controle e percepção de valor. A análise de cada uma dessas questões é apresentada a seguir.

4.5.1 Setor de atuação

O objetivo desta questão foi identificar a área principal de atuação dos respondentes, permitindo contextualizar os resultados do *survey* quanto ao perfil profissional e técnico predominante entre os participantes.

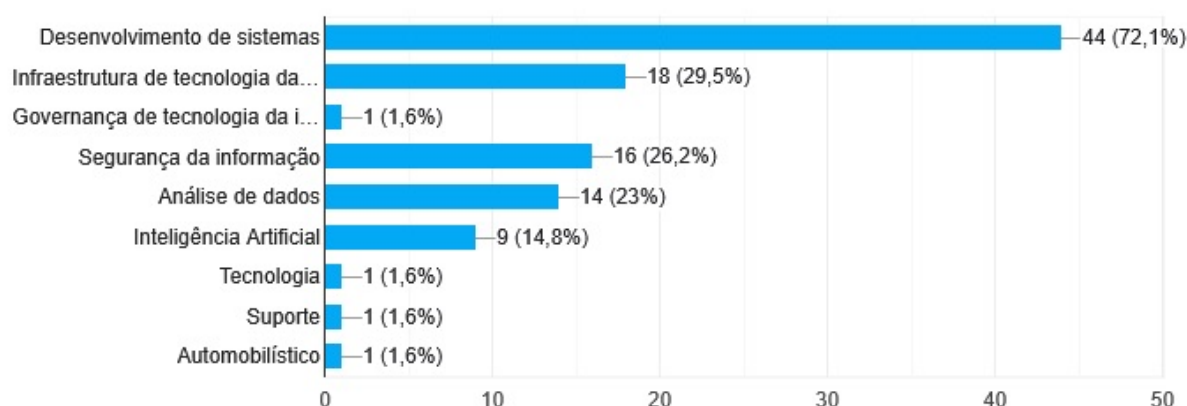


Figura 10 – Setor de atuação.

Os dados indicam que a maioria atua em áreas voltadas ao desenvolvimento de sistemas, seguidas por setores relacionados à infraestrutura, segurança da informação, análise de dados e inteligência artificial. Isso reforça que os achados da pesquisa refletem principalmente a realidade de equipes envolvidas diretamente na construção, manutenção e proteção de soluções tecnológicas.

4.5.2 Utilização de metodologias ágeis na empresa

O objetivo da questão é investigar a adoção de metodologias ágeis nas empresas onde os profissionais atuam, buscando entender o contexto organizacional em que as práticas de gestão de riscos são aplicadas.

Verificou-se, por meio da análise, que a maioria dos profissionais participantes (90%) trabalha em empresas que utilizam metodologias ágeis. Esse dado confirma a presença consolidada de abordagens ágeis no cenário atual dos projetos, o que reforça a importância de avaliar como a gestão de riscos se adapta a esse modelo de trabalho dinâmico e iterativo.

Dentre os respondentes que afirmaram utilizar metodologias ágeis, buscou-se identificar quais delas são mais empregadas pelas empresas. Essa identificação permite com-

preender quais *frameworks* têm maior influência nas práticas e percepções relacionadas à gestão de riscos em ambientes ágeis.

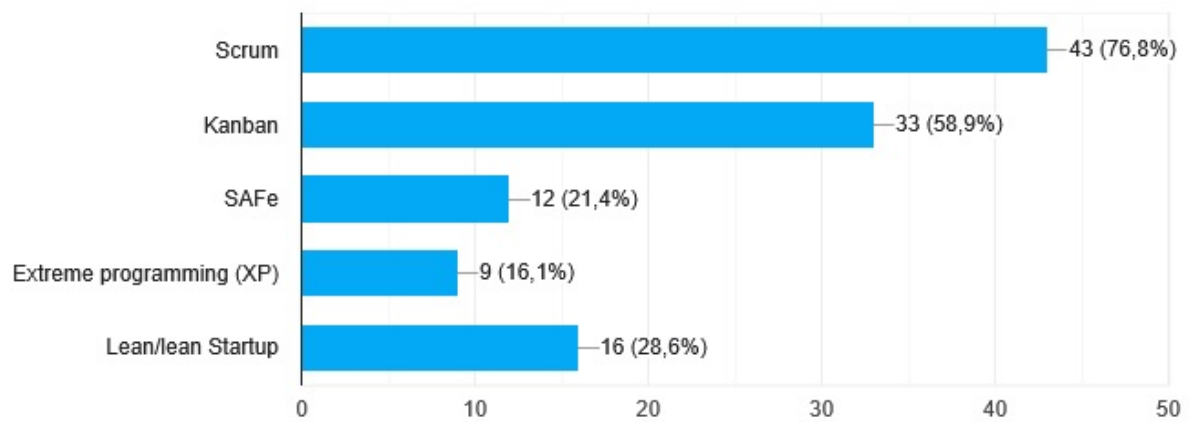


Figura 11 – Metodologias ágeis mais utilizadas.

As metodologias Scrum e Kanban foram as mais citadas entre os respondentes que utilizam abordagens ágeis. Essa predominância sugere que grande parte das práticas de planejamento, acompanhamento e mitigação de riscos relatadas no *survey* ocorre dentro desses dois *frameworks*, amplamente reconhecidos pela estruturação clara de papéis, eventos e entregas iterativas.

4.5.3 A cultura da empresa encoraja a comunicação aberta sobre riscos.

A proposta desta afirmativa foi identificar se o ambiente organizacional favorece a transparência na gestão de riscos, fator essencial para a detecção precoce e o tratamento eficaz de ameaças aos projetos. A comunicação aberta é um dos pilares para a construção de uma cultura preventiva, especialmente em contextos que utilizam metodologias ágeis, onde a colaboração contínua é valorizada.

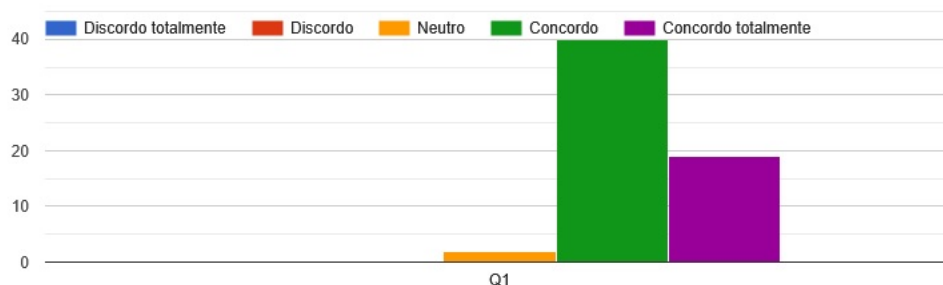


Figura 12 – Comunicação aberta sobre riscos.

Com base nas respostas obtidas, observou-se que a maioria dos participantes demonstrou grau elevado de concordância, indicando que suas organizações tendem a pro-

mover um espaço seguro para a discussão de riscos. Esse resultado sugere uma estrutura cultural favorável à adoção de boas práticas de gestão de riscos nos projetos analisados.

4.5.4 A equipe identifica riscos de forma proativa durante as atividades de planejamento (ex: *Planning*) e a avaliação de desenvolvimento ao final de cada ciclo (ex: Retrospectiva).

Buscou-se, com esta afirmativa, compreender a proatividade da equipe na identificação de riscos, especialmente durante os momentos chave de planejamento e avaliação de desenvolvimento, que são essenciais para o sucesso da gestão de riscos em ambientes ágeis. A identificação precoce de riscos é fundamental para garantir que os problemas sejam tratados antes que se tornem obstáculos significativos aos objetivos do projeto.

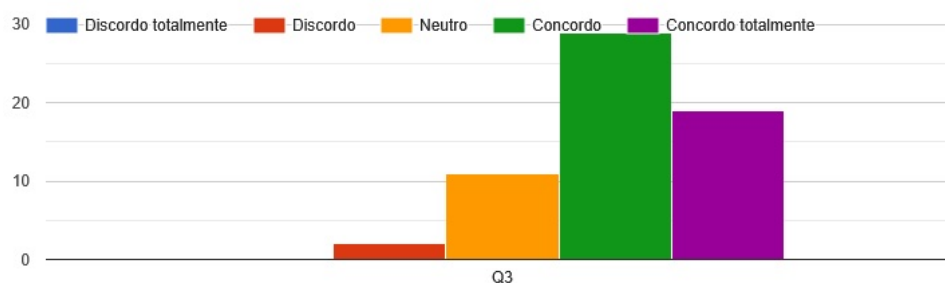


Figura 13 – Identificação proativa dos riscos.

Com os resultados obtidos, observou-se que a maioria dos participantes demonstrou concordância com a afirmação, com destaque para os que concordaram totalmente e os que concordaram, indicando que as equipes, na maioria das vezes, estão atentas e se antecipam à identificação de riscos. A presença de algumas respostas neutras e discordantes sugere que, em determinados casos, ainda existem oportunidades de melhoria nas práticas de identificação de riscos, talvez em algumas equipes ou contextos específicos. De forma geral, os resultados indicam uma tendência positiva na abordagem proativa para a gestão de riscos.

4.5.5 Riscos relacionados a normas e regulamentações de segurança são tratados com rigor.

A afirmativa teve como intuito investigar o comprometimento das equipes com a gestão de riscos associados a requisitos legais e normativos, especialmente aqueles ligados à segurança da informação e conformidade regulatória. O tratamento rigoroso desses riscos é essencial para assegurar que os projetos estejam em conformidade com obrigações legais e padrões de segurança, prevenindo penalidades e impactos negativos à reputação.

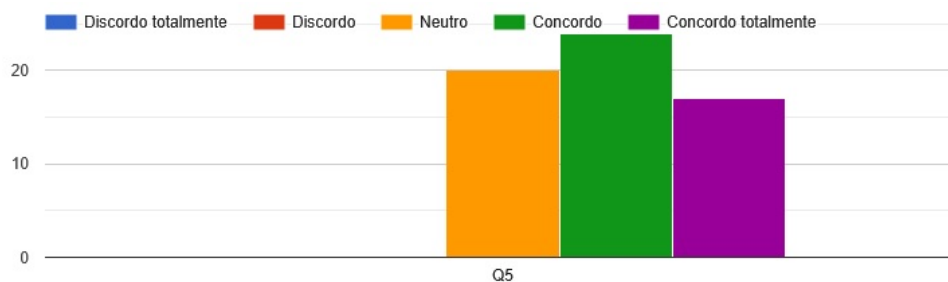


Figura 14 – Riscos de segurança são tratados com rigor.

De acordo com os 61 respondentes, 41 demonstraram concordância com a afirmativa, sendo 24 respostas em “Concordo” e 17 em “Concordo totalmente”, o que representa cerca de 67,2%. Os demais 20 respondentes (aproximadamente 32,8%) optaram pela alternativa “Neutro”, e nenhuma resposta foi registrada nas opções de discordância. Esses dados indicam uma forte percepção de rigor no tratamento dos riscos relacionados à conformidade e segurança, embora a presença expressiva de respostas neutras sugira que, em algumas equipes, esse rigor ainda pode não ser plenamente evidente ou sistematizado.

4.5.6 Riscos técnicos (ex: dúvida técnica) são tratados com a mesma importância que riscos de negócio.

Pode-se avaliar com essa afirmativa, o equilíbrio no tratamento dos diferentes tipos de riscos dentro dos projetos ágeis, especialmente a equivalência entre riscos técnicos e riscos de negócio. Garantir que ambos sejam tratados com o devido grau de importância é essencial para evitar impactos tanto na viabilidade técnica quanto na entrega de valor ao cliente.

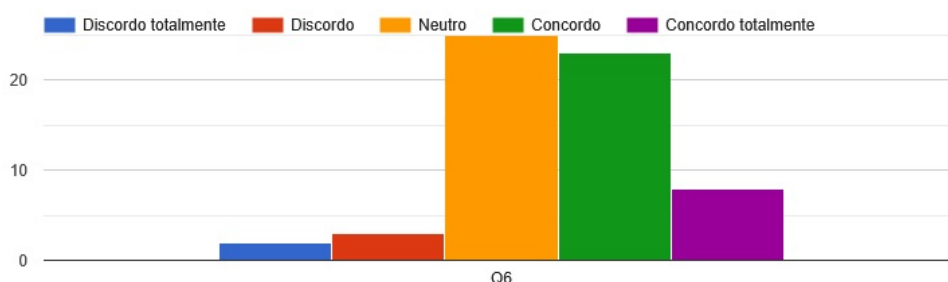


Figura 15 – Importância de riscos técnicos versus riscos de negócio.

Dentre os respondentes, 31 manifestaram concordância com a afirmativa, sendo 23 em “Concordo” e 8 em “Concordo totalmente”, o que representa aproximadamente 50,8%. Em contrapartida, 30 respostas (cerca de 49,2%) foram divididas entre “Neutro” (25), “Discordo” (3) e “Discordo totalmente” (2), revelando uma divisão perceptível nas práticas adotadas pelas equipes. Os dados sugerem que, embora haja reconhecimento

quanto à importância dos riscos técnicos, ainda existem contextos em que essa equiparação não é plenamente praticada, indicando possíveis assimetrias na priorização dos riscos.

4.5.7 A equipe classifica e prioriza os riscos identificados com base em uma matriz de impacto e probabilidade.

Esta afirmativa foi elaborada para verificar se as equipes adotam práticas sistemáticas de priorização de riscos, por meio de ferramentas como a matriz de impacto e probabilidade, fundamentais para orientar decisões eficazes em ambientes ágeis.

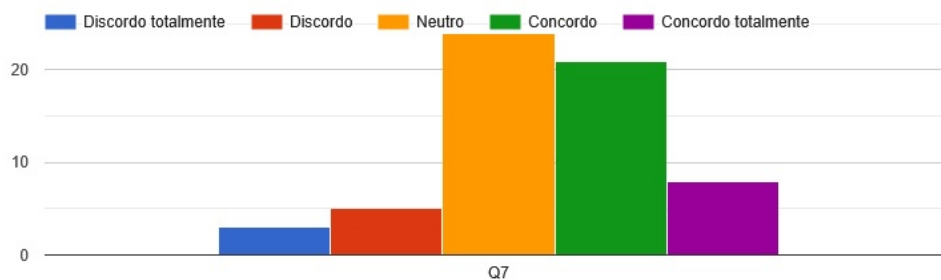


Figura 16 – Priorização dos riscos com matriz de impacto e probabilidade.

Os resultados apontaram que uma parte significativa dos respondentes manifestou posição neutra, enquanto outra parcela relevante demonstrou concordância com a afirmativa. Esse cenário sugere que o uso de métodos formais de classificação de riscos ainda não está plenamente consolidado nas equipes. Assim, os dados podem refletir um estágio em que as equipes ainda não estão plenamente preparadas para aplicar práticas eficazes de gestão de riscos em ambientes ágeis. A ausência de uma priorização sistemática pode afetar a efetividade das ações de mitigação nos projetos.

4.5.8 A liderança do projeto (Ex: *Scrum Master* ou Gerente de Projetos) facilita discussões sobre riscos e promove a mitigação (diminuição da probabilidade) de riscos no time.

Buscou-se com esta afirmativa compreender o papel da liderança na promoção de práticas de gestão de riscos dentro das equipes, especialmente no que diz respeito ao estímulo a discussões abertas e à busca por ações de mitigação. Em metodologias ágeis, a atuação de figuras como o *Scrum Master* é essencial para criar um ambiente colaborativo e voltado à melhoria contínua.

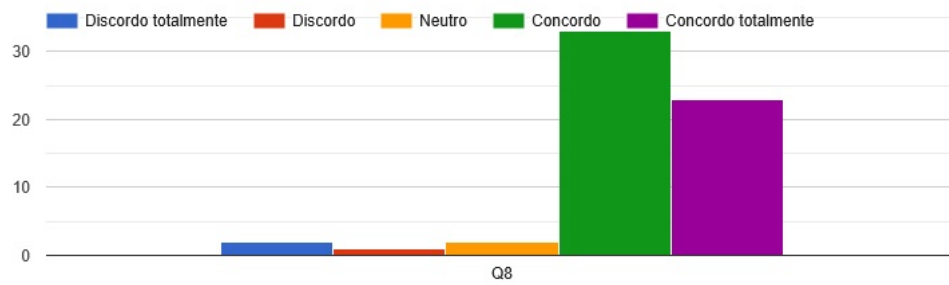


Figura 17 – Discussão e mitigação dos riscos.

De acordo com as respostas obtidas, a grande maioria dos participantes demonstrou concordância com a afirmativa, sendo as opções “Concordo” e “Concordo totalmente” as mais escolhidas. Esse resultado sugere que, nas equipes analisadas, a liderança tem exercido um papel ativo na identificação e mitigação de riscos, o que pode refletir um bom nível de maturidade em relação à governança de projetos ágeis.

4.5.9 O responsável pelo projeto (ex: *Product Owner*) inclui riscos de negócio no refinamento da lista de requisitos a serem desenvolvidos.

A afirmativa em destaque teve como foco avaliar o envolvimento da liderança do projeto no processo de integração dos riscos de negócio durante o refinamento do *backlog*. A consideração desses riscos nesse momento é essencial para garantir entregas mais alinhadas às prioridades estratégicas e reduzir surpresas durante o desenvolvimento.

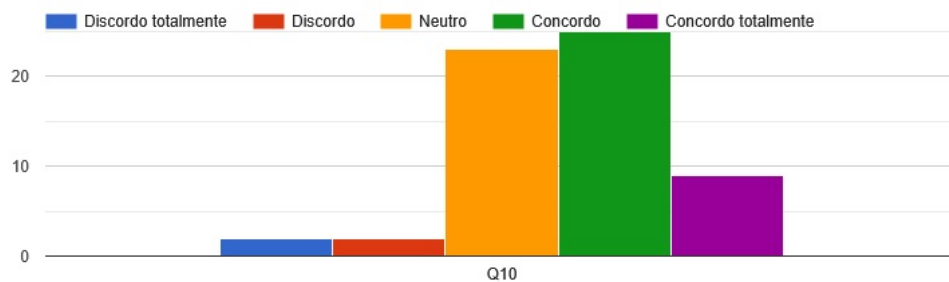


Figura 18 – Inclusão de riscos de negócio.

Considerando as respostas coletadas (61), 34 demonstraram concordância com a afirmativa, sendo 25 em “Concordo” e 9 em “Concordo totalmente”, o que representa aproximadamente 55,7% das respostas. Por outro lado, 27 respondentes (cerca de 44,3%) se dividiram entre “Neutro” (23), “Discordo” (2) e “Discordo totalmente” (2), indicando que essa prática ainda não é adotada de forma consistente em todas as equipes. Os resultados revelam uma tendência positiva, mas também apontam para a necessidade de maior integração entre o *Product Owner* e a gestão de riscos no processo de planejamento.

4.5.10 Os riscos identificados nos projetos são analisados quantitativamente de acordo com os objetivos do projeto, ou seja, com estimativas numéricas do impacto do risco em termos de prazo, custo etc.

O foco dessa afirmativa é verificar se as equipes realizam análises quantitativas de riscos, uma prática que permite mensurar com maior precisão os impactos potenciais sobre prazos, custos e qualidade. Essa abordagem é particularmente relevante para projetos que demandam maior controle e previsibilidade, mesmo em contextos ágeis.

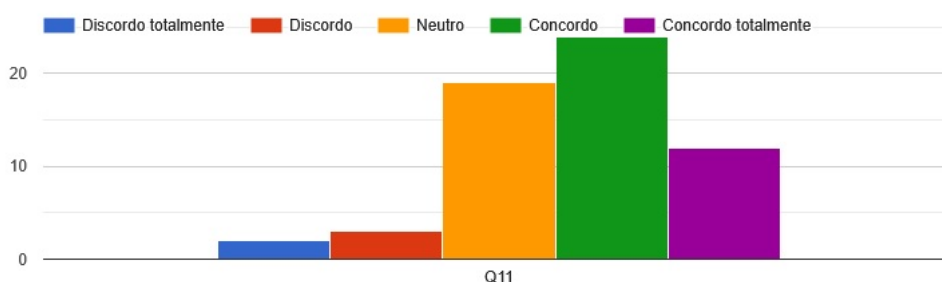


Figura 19 – Avaliação quantitativa dos riscos.

A distribuição das respostas revelou uma predominância das opções “Concordo” e “Neutro”, seguidas por uma menor parcela de “Concordo totalmente”. Esse resultado pode indicar que, embora haja algum nível de aplicação de análise quantitativa, ela ainda não é amplamente consolidada entre os participantes. A presença significativa de respostas neutras também sugere possíveis lacunas no uso consistente desse tipo de abordagem nas equipes avaliadas.

4.5.11 Ferramentas ou simulações são utilizadas, quando pertinentes, para apoiar a análise dos riscos (ex: análise de Monte Carlo, estimativas em três pontos).

Essa afirmativa visou avaliar se as equipes utilizam ferramentas ou simulações específicas para realizar uma análise mais aprofundada e quantitativa dos riscos, como as metodologias de Monte Carlo ou estimativas em três pontos. A utilização dessas ferramentas pode trazer maior precisão na identificação e mitigação de riscos, proporcionando uma abordagem mais assertiva para a gestão de riscos em projetos ágeis.

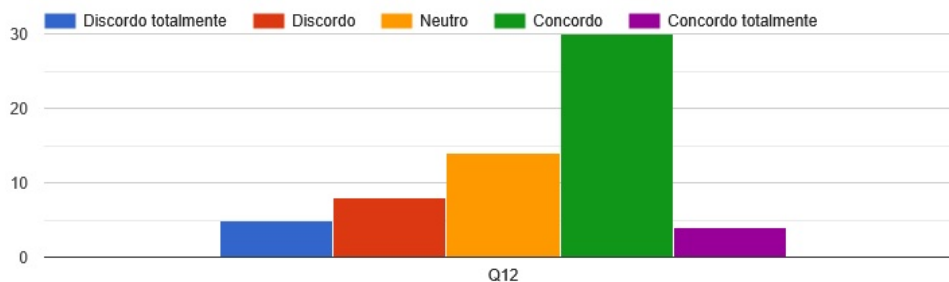


Figura 20 – Ferramentas ou simulações de apoio aos riscos.

Dentre os respondentes, 34 indicaram concordância com a afirmativa, sendo 30 respostas em “Concordo” e 4 em “Concordo totalmente” (aproximadamente 55,7%). No entanto, 27 participantes (cerca de 44,3%) se distribuíram entre as opções “Neutro” (14), “Discordo” (8) e “Discordo totalmente” (5). Esses resultados sugerem que, apesar de uma boa parte das equipes adotar ou considerar o uso de ferramentas e simulações na gestão de riscos, ainda há uma proporção significativa de profissionais que não utilizam essas abordagens de maneira consistente. Isso pode indicar uma necessidade de maior familiarização com essas técnicas ou até mesmo uma resistência a sua aplicação no contexto ágil.

4.5.12 Os riscos identificados são revisados regularmente ao final de cada ciclo de desenvolvimento (ex: Sprint/Iteração de desenvolvimento).

O propósito desta afirmativa é examinar se as equipes adotam práticas regulares de revisão de riscos ao final de cada ciclo de desenvolvimento, como *sprints* ou iterações. Esse hábito é fundamental em metodologias ágeis, nas quais a adaptação contínua e o aprendizado iterativo são incentivados para garantir a melhoria progressiva dos processos e resultados.

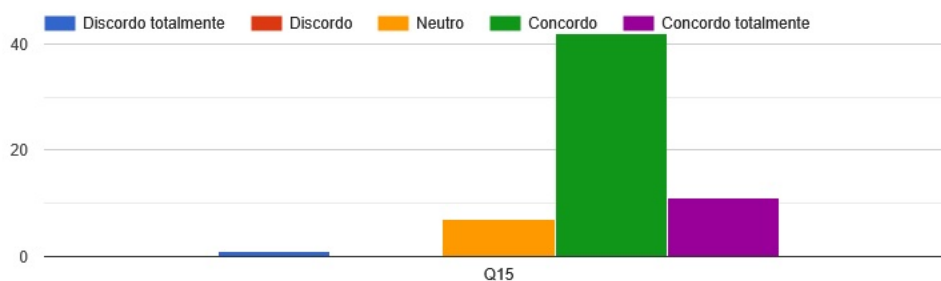


Figura 21 – Revisão dos riscos ao final de cada ciclo.

As respostas demonstraram uma predominância significativa da opção “Concordo”, indicando que, para a maioria dos participantes, a revisão periódica de riscos faz parte da rotina ao final de cada ciclo. Esse resultado sugere um alinhamento com os princí-

pios ágeis, nos quais a inspeção frequente e a resposta a mudanças são valorizadas como mecanismos essenciais de gestão eficaz.

4.5.13 São utilizadas ferramentas específicas para monitorar os riscos (ex: Jira, *Risk Matrix*).

A finalidade desta afirmativa é identificar se as equipes adotam o uso de recursos ou sistemas dedicados ao monitoramento de riscos ao longo dos projetos. A adoção dessas ferramentas é considerada um indicativo de maior estruturação e consistência na gestão contínua de riscos em ambientes ágeis.

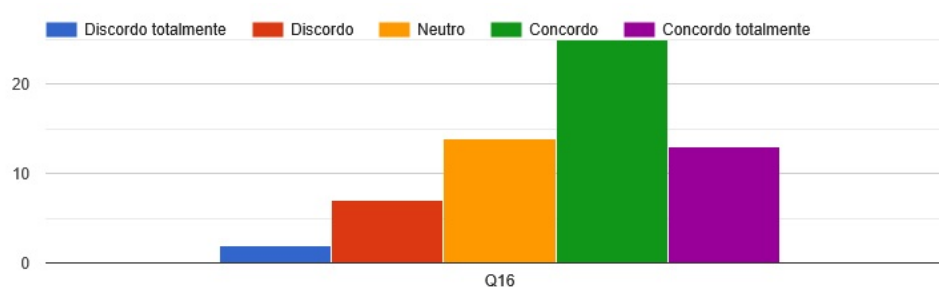


Figura 22 – Ferramentas para monitor os riscos.

A maioria dos participantes assinalou as opções “Concordo” e “Concordo totalmente”, cerca de 62,3%, evidenciando que o uso de ferramentas específicas é relativamente presente entre os respondentes. No entanto, a quantidade significativa de respostas neutras, bem como a presença de discordâncias, cerca de 37,7%, sugere que essa prática ainda não é plenamente consolidada. Isso pode indicar diferenças no grau de maturidade ou na priorização do monitoramento de riscos entre as equipes avaliadas.

4.5.14 A incidência de riscos não previstos (surpresas) é mínima no contexto da equipe.

Esta afirmativa foi formulada para investigar a percepção dos respondentes sobre a ocorrência de riscos não antecipados durante o desenvolvimento dos projetos, ou seja, aquelas surpresas que podem afetar o andamento do trabalho. A identificação e gestão eficaz dos riscos ajudam a minimizar a ocorrência de imprevistos, e essa avaliação oferece um indicativo de como as equipes percebem a previsibilidade dos projetos em termos de riscos não identificados.

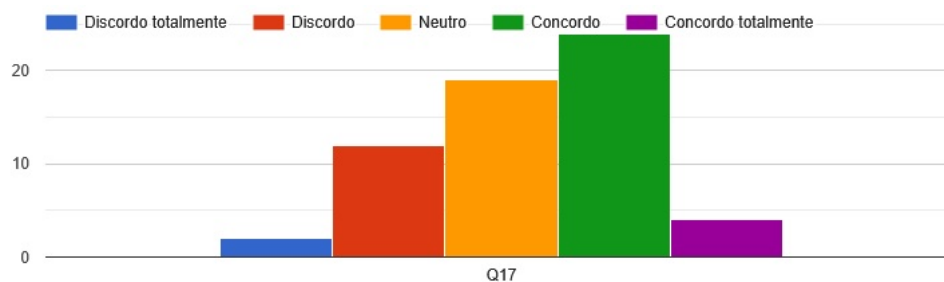


Figura 23 – Incidência de riscos não previstos.

Entre os 61 participantes, a maior parte dos respondentes manifestou concordância, sendo 24 respostas em “Concordo” e 4 em “Concordo totalmente” (totalizando aproximadamente 45,9%). No entanto, um número considerável de respostas se distribuiu entre “Neutro” (19), “Discordo” (12) e “Discordo totalmente” (2), somando cerca de 54,1%. Esses dados sugerem que, embora uma boa parte dos respondentes perceba que a incidência de riscos não previstos é minimizada, ainda existe uma parcela significativa que indica que, em suas equipes, surpresas ou riscos imprevistos ainda são uma preocupação relevante. Isso indica que, mesmo em equipes ágeis, o gerenciamento proativo de riscos precisa ser mais consistente e abrangente.

4.5.15 Os riscos são documentados e compartilhados com todas as pessoas envolvidas no desenvolvimento (ex: *Stakeholders*).

O objetivo desta afirmativa é avaliar a formalização na gestão de riscos, especificamente quanto à documentação e à transparência na comunicação com os envolvidos no projeto. A prática de registrar e compartilhar riscos é fundamental para assegurar que todas as partes interessadas estejam cientes dos possíveis impactos e possam contribuir para decisões mais alinhadas e estratégicas.

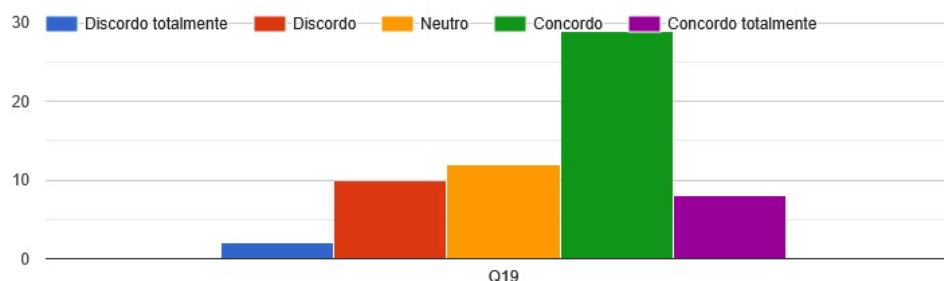


Figura 24 – Documentação e compartilhamento dos riscos.

Analisando as respostas de 61 participantes, a maioria indicou concordância com a afirmativa, sendo 29 respostas em “Concordo” e 8 em “Concordo totalmente” (totalizando aproximadamente 60,6%). No entanto, um número considerável de respostas se distribuiu entre “Neutro” (12), “Discordo” (10) e “Discordo totalmente” (2), que somam cerca de

39,4%, revelando que, embora a prática de documentar e compartilhar riscos esteja presente em muitas equipes, ainda há espaço para melhorias na padronização e consistência dessa comunicação. Esses dados sugerem que a gestão de riscos, apesar de reconhecida, nem sempre é plenamente integrada aos mecanismos formais de registro e transparência do projeto.

4.5.16 A equipe percebe que a gestão de riscos contribui para entregas mais previsíveis e alinhadas às necessidades do negócio.

Por fim, esta afirmativa buscou compreender a percepção dos respondentes sobre o impacto da gestão de riscos nos resultados dos projetos, especialmente no que se refere à previsibilidade e ao alinhamento com os objetivos do negócio. Essa visão é relevante para avaliar se as práticas adotadas vão além da identificação e controle, refletindo também na qualidade das entregas e na geração de valor.

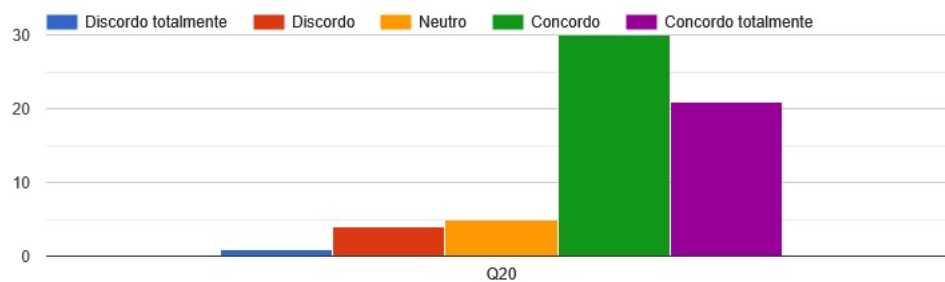


Figura 25 – Contribuições da gestão de riscos.

A maioria das respostas concentrou-se nas opções “Concordo” e “Concordo totalmente”, sugerindo que os participantes reconhecem a contribuição da gestão de riscos para a entrega de produtos mais previsíveis e adequados às necessidades da organização. Isso reforça a ideia de que, mesmo em ambientes ágeis, onde mudanças são esperadas, a gestão de riscos pode ser percebida como um diferencial estratégico para o sucesso dos projetos.

5 Conclusão

O presente trabalho teve como objetivo estudar a gestão de riscos aplicada a métodos ágeis, buscando compreender seus benefícios e desvantagens durante o desenvolvimento de projetos. Para atingir esse objetivo, foi realizada uma pesquisa de levantamento (*survey*), aplicada ao longo de duas semanas, com a participação de profissionais atuantes na cidade de Uberlândia. A análise dos dados coletados permitiu identificar resultados significativos.

Os dados revelaram que a maioria dos profissionais participantes atua no setor de desenvolvimento de sistemas e que grande parte das empresas representadas já adota metodologias ágeis, com destaque para o uso de Scrum e Kanban. Em relação à gestão de riscos, os resultados indicaram uma percepção positiva quanto à comunicação aberta e à identificação proativa de riscos durante as etapas ágeis. No entanto, práticas como a priorização sistemática de riscos e o uso de análises quantitativas ainda não estão plenamente consolidadas, evidenciando oportunidades de aprimoramento na aplicação prática da gestão de riscos nesses ambientes. Além disso, observou-se que, embora haja atenção aos riscos regulatórios e de negócio, ainda há inconsistências na documentação, compartilhamento e monitoramento junto aos *stakeholders*, o que pode comprometer a efetividade das ações de mitigação e a tomada de decisão ao longo do projeto.

O questionário foi enviado para cerca de 2.500 a 3.000 possíveis participantes, mas a amostra obtida foi reduzida, com apenas 61 respostas registradas no período de duas semanas em que ficou disponível.

O método de levantamento, ou *survey*, apresenta vantagens relevantes, como a capacidade de alcançar um grande número de possíveis participantes em um curto período de tempo e baixo custo operacional, além de permitir a padronização das respostas, o que facilita a análise estatística. Contudo, também possui limitações significativas, uma vez que sua eficácia depende da participação voluntária dos respondentes, a qual pode ser afetada por fatores como desinteresse, falta de tempo ou, conforme observado neste estudo, o não acesso frequente aos e-mails.

Embora a amostra tenha sido reduzida, os resultados obtidos oferecem contribuições relevantes e podem servir como base para futuros estudos. Recomenda-se, em pesquisas posteriores, a ampliação da amostra, incluindo mais empresas e colaboradores de diferentes cidades, além de diversificar os canais de coleta de dados e estender o período de aplicação do questionário. Essas medidas podem proporcionar uma análise mais abrangente e aprofundada do tema.

Referências

BANDEIRA, J. Y. F. Um survey sobre o uso de metodologias ágeis por startups do estado do ceará. 2024. Disponível em: <https://repositorio.ufc.br/bitstream/riufc/78845/1/2024_tcc_jyfbandeira.pdf>. Citado na página 38.

DIGITAL.AI. **17th State of Agile Report**. 2023. <<https://stateofagile.com/>>. Citado na página 31.

FERNANDES, P. J. M. A.; JR, R. R. O gerenciamento de riscos em projetos gerenciados por abordagens ágeis:: uma revisão sistemática da literatura. 2021. Disponível em: <<https://dialnet.unirioja.es/servlet/articulo?codigo=8363599>>. Citado 3 vezes nas páginas 36, 37 e 40.

GARCIA, F. V.; HAUCK, J. C. R.; HAHN, F. N. R. Managing risks in agile methods: a systematic literature mapping. 2022. Disponível em: <<https://people.cs.pitt.edu/~chang/seke/seke22paper/paper123.pdf>>. Citado 2 vezes nas páginas 38 e 39.

GHOBADI, S.; MATHIASSEN, L. A model for assessing and mitigating knowledge sharing risks in agile software development. 2016. Citado na página 42.

IEEE. **IEEE Standard 16085-2006. IEEE/ISO/IEC 16085:2006 Systems and Software Engineering—Software Life Cycle Processes—Risk Management**. 2006. Citado na página 32.

ISO. **ISO Guide 73:2009, Risk Management: Vocabulary**. 2009. Citado na página 31.

MACHADO, E. E. M.; OLIVEIRA, C. C. de. Relação entre as boas práticas descritas no pmbok e as metodologias ágeis para o desenvolvimento de softwares. 2022. Disponível em: <<https://periodicos.faex.edu.br/index.php/e-Locucaao/article/download/446/293>>. Citado na página 37.

MEIRELES, M. Validação de escala likert: 1-conceito. 2024. Disponível em: <<https://dialnet.unirioja.es/descarga/articulo/9701684.pdf>>. Citado na página 41.

MINEIRO, M. Pesquisa de survey e amostragem: Aportes teoricos elementares. 2020. Disponível em: <<https://periodicos2.uesb.br/index.php/reed/article/view/7677>>. Citado 2 vezes nas páginas 40 e 41.

PMI, P. M. I. **UM GUIA DO CONHECIMENTO EM GERENCIAMENTO DE PROJETOS (GUIA PMBOK®)**. Quinta edição. [S.l.: s.n.], 2013. Citado 13 vezes nas páginas 9, 18, 19, 20, 21, 22, 23, 24, 31, 32, 33, 34 e 35.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software: uma abordagem profissional**. 9. ed.. ed. [S.l.: s.n.], 2021. Citado 13 vezes nas páginas 9, 12, 13, 14, 15, 16, 17, 25, 26, 28, 29, 30 e 31.

PRIKLADNICKI, R.; WILLI, R.; MILANI, F. **Métodos Ágeis para Desenvolvimento de Software**. [S.l.: s.n.], 2014. Citado 7 vezes nas páginas 9, 24, 25, 27, 28, 29 e 30.

- RODRIGUES, T. D. de F. F.; OLIVEIRA, G. S. de; SANTOS, J. A. dos. As pesquisas qualitativas e quantitativas na educação. 2021. Disponível em: <<https://revistaprisma.emnuvens.com.br/prisma/article/view/49/41>>. Citado na página 41.
- SILVA, C. R.; ZAIDAN, F.; BRAGA, J. L. Gestão do conhecimento potencializando a gestão de riscos de projetos de desenvolvimento de software ágil utilizando o scrum. 2023. Disponível em: <<https://periodicos.ufmg.br/index.php/fronteiras-rc/article/view/40348>>. Citado na página 36.
- SOMMERVILLE, I. **Engenharia de Software**. 9. ed.. ed. [S.l.: s.n.], 2011. Citado 3 vezes nas páginas 9, 18 e 25.
- SOUZA, K. C. de; SILVEIRA, M. U. da; SOUZA, C. F. S. de; LIMA, D. A. da S. Princípios ágeis no gerenciamento de projetos de uma indústria farmacêutica. v. 7, 2021. Citado 2 vezes nas páginas 26 e 27.
- SUTHERLAND, J. **Scrum: A Arte de Fazer o Dobro do Trabalho na Metade do Tempo**. [S.l.: s.n.], 2014. Citado 2 vezes nas páginas 9 e 31.
- WAZLAWICK, R. S. **Metodologia de Pesquisa para Ciência da Computação**. 3. ed.. ed. [S.l.: s.n.], 2020. Citado na página 40.

Apêndices

APÊNDICE A – Questionário

Introdução

Este formulário faz parte de uma pesquisa acadêmica desenvolvida como requisito para a conclusão do curso de Bacharelado em Sistemas de Informação. O objetivo do estudo é analisar como ocorre o processo de gestão de riscos em projetos que utilizam metodologias ágeis no ambiente organizacional.

As informações coletadas serão utilizadas exclusivamente para fins acadêmicos, de forma anônima e confidencial, garantindo o sigilo dos participantes e das empresas envolvidas.

Agradecemos antecipadamente pela sua contribuição para este estudo.

1. Características da Empresa

Cargo/Função:

- ☐ Scrum Master/Agile Coach
- ☐ Product Owner
- ☐ Desenvolvedor(a)/Time de Dev
- ☐ Gerente de Projetos/Processos
- ☐ Garantia da Qualidade (QA)
- ☐ Outro: _____

Tamanho da Empresa:

- ☐ Pequena (até 50 funcionários)
- ☐ Média (51–500 funcionários)
- ☐ Grande (acima de 500 funcionários)

Setor de Atuação:

- ☐ Desenvolvimento de sistemas
- ☐ Infraestrutura de tecnologia da informação
- ☐ Governança de tecnologia da informação
- ☐ Segurança da informação
- ☐ Análise de dados
- ☐ Inteligência Artificial
- ☐ Outro: _____

A empresa a qual você faz parte utiliza metodologia ágil?

- ☐ Sim
- ☐ Não

Caso a resposta anterior seja positiva, qual(is) metodologia(s)?

- ☐ Scrum
- ☐ Kanban
- ☐ SAFe
- ☐ Extreme programming (XP)
- ☐ Lean/lean Startup
- ☐ Outro: _____

2. Processo de Gestão de Riscos

Instruções:

Por favor, leia atentamente cada uma das afirmações a seguir e indique seu grau de concordância, utilizando a escala de resposta que varia de “*Discordo totalmente*” a “*Concordo totalmente*”.

Afirmativa	Discordo totalmente	Discordo	Neutro	Concordo	Concordo totalmente
Q1) A cultura da empresa encoraja a comunicação aberta sobre riscos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q2) Existe um plano de ação definido para mitigar (identificar e reduzir) riscos críticos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q3) A equipe identifica riscos de forma proativa durante as atividades de planejamento (ex: Planning) e na avaliação de desenvolvimento ao final de cada ciclo (ex: Retrospectiva).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q4) Os riscos emergentes são discutidos durante as reuniões diárias da equipe.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q5) Riscos relacionados a normas e regulamentações de segurança são tratados com rigor.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q6) Riscos técnicos (ex: dúvida técnica) são tratados com a mesma importância que riscos de negócio.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q7) A equipe classifica e prioriza os riscos identificados com base em uma matriz de impacto e probabilidade.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Afirmativa	Discordo totalmente	Discordo	Neutro	Concordo	Concordo totalmente
Q8) A liderança do projeto (ex: Scrum Master ou gerente de projetos) facilita discussões sobre riscos e promove a mitigação (diminuição da probabilidade) de riscos no time.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q9) A equipe ajusta a lista de requisitos a serem desenvolvidos (Backlog do projeto) com base nos riscos identificados.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q10) O responsável pelo projeto (ex: Product Owner) inclui riscos de negócio no refinamento da lista de requisitos a serem desenvolvidos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q11) Os riscos identificados nos projetos são analisados quantitativamente de acordo com os objetivos do projeto, ou seja, com estimativas numéricas do impacto do risco em termos de prazo, custo etc.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q12) Ferramentas ou simulações são utilizadas, quando pertinentes, para apoiar a análise dos riscos (ex: análise de Monte Carlo, estimativas em três pontos).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q13) A planilha de riscos identificados é adaptada continuamente com base nos feedbacks da avaliação final dos ciclos de desenvolvimento (ex: Retrospectivas).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Afirmativa	Discordo totalmente	Discordo	Neutro	Concordo	Concordo totalmente
Q14) A empresa registra uma base de conhecimento com os riscos passados em projetos anteriores para evitar repetições.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q15) Os riscos identificados são revisados regularmente ao final de cada ciclo de desenvolvimento (ex: Sprint/Iteração de desenvolvimento).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q16) São utilizadas ferramentas específicas para monitorar os riscos (ex: Jira, Risk Matrix).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q17) A incidência de riscos não previstos (surpresas) é mínima no contexto da equipe.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q18) Os riscos identificados impactam diretamente o planejamento dos prazos, custos ou qualidade das entregas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q19) Os riscos são documentados e compartilhados com todas as pessoas envolvidas no desenvolvimento (ex: Stakeholders).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Q20) A equipe percebe que a gestão de riscos contribui para entregas mais previsíveis e alinhadas às necessidades do negócio.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>