

Avaliação da Maturidade Ágil em Equipes Distribuídas de Nível 2

Arthur Gusmão de Matheus¹, Gyna Beatriz Yun¹, Ana Claudia Rossi¹

¹Faculdade de Computação e Informática – Universidade Presbiteriana Mackenzie
Rua da Consolação, 930, Consolação – 01302-907 – São Paulo – SP – Brazil

arthurgm2002@icloud.com, gynayun@gmail.com, anaclaudia.rossi@mackenzie.br

Resumo. *Este estudo explora a Avaliação de Maturidade Ágil em Equipes Distribuídas no Nível 2, combinando métodos de Estudo de Caso e Pesquisa-Ação com foco nas Áreas-Chave de Processo (KPAs) do Modelo de Maturidade Ágil (AMM). Estas KPAs são categorizadas conforme os riscos do Desenvolvimento Ágil Distribuído (DAD). Foi utilizado um questionário para coletar dados de desenvolvedores de uma equipe de desenvolvimento específica, gerando insights sobre práticas ágeis e suas melhorias contínuas, orientadas pelo ciclo Plan-Do-Check-Act (PDCA). A metodologia resulta em recomendações significativas para aprimorar práticas ágeis em ambientes de trabalho distribuídos.*

Palavras-chave: *Avaliação de Maturidade Ágil, Equipes Distribuídas, Desenvolvimento Ágil Distribuído (DAD), Práticas Ágeis, Ciclo Plan-Do-Check-Act (PDCA).*

Abstract. *This study explores the Agile Maturity Assessment in Distributed Teams at Level 2, combining Case Study and Action Research methods with a focus on the Key Process Areas (KPAs) of the Agile Maturity Model (AMM). These KPAs are categorized based on the risks of Distributed Agile Development (DAD). A questionnaire was used to gather data from developers in a specific development team, generating insights into agile practices and their continuous improvements guided by the Plan-Do-Check-Act (PDCA) cycle. The methodology results in significant recommendations for enhancing agile practices in distributed work environments.*

Keywords: *Agile Maturity Assessment, Distributed Teams, Distributed Agile Development (DAD), Agile Practices, Plan-Do-Check-Act (PDCA) Cycle.*

1. Introdução

Estudos recentes como o de Khalid, Mahmood e Farooq (2021) enfatizam a crescente valorização da agilidade nessas práticas. À medida que as equipes se expandem para ambientes distribuídos, com membros geograficamente dispersos, surgem desafios particulares que exigem adaptação e avaliação contínua das práticas ágeis para garantir eficácia e eficiência.

Shrivastava e Rathod (2017) ressaltam que a comunicação, coordenação e construção de confiança entre membros tornam-se complexas nesses ambientes. O Desenvolvimento Ágil Distribuído (DAD) amplifica esses desafios, introduzindo riscos que

podem comprometer diretamente o desempenho dos projetos. A ausência de um modelo robusto para avaliar e melhorar a maturidade das práticas ágeis em tais ambientes se torna uma barreira significativa.

Diante disso, este estudo propõe o desenvolvimento de um questionário para avaliar a maturidade ágil em equipes distribuídas baseado no Modelo de Maturidade Ágil (AMM), elaborado por Patel e Ramachandran (2009). As respostas serão coletadas utilizando uma escala Likert e refletindo as *Key Process Areas* (KPAs) desse modelo, com foco específico nos desafios identificados por equipes distribuídas que adotam a abordagem ágil em seus processos. Este instrumento permitirá não apenas a avaliação do nível de maturidade ágil, mas também fornecerá insights para melhorias contínuas através do ciclo PDCA.

A relevância deste estudo é evidenciada pela crescente adoção de metodologias ágeis em ambientes de desenvolvimento de software distribuídos, onde a otimização da produtividade e qualidade é imperativa. A implementação de práticas ágeis nesses contextos é desafiadora devido à distância geográfica e necessidade de trabalho remoto [Shrivastava and Rathod 2017]. O AMM oferece um quadro para integrar métodos ágeis e modelos de melhoria de processo, abordando uma lacuna significativa na literatura existente.

O trabalho está estruturado da seguinte forma: após essa introdução, a seção 2 apresentará uma revisão da literatura relacionada ao desenvolvimento ágil em equipes distribuídas. A seção 3 detalhará a metodologia de pesquisa empregada, enquanto a seção 4 discutirá os resultados obtidos. Por fim, a seção 5 conterá as considerações finais, destacando as contribuições do estudo para a prática e teoria do desenvolvimento ágil em ambientes distribuídos.

2. Referencial Teórico

2.1. Equipes Distribuídas

Equipes distribuídas podem ser definidas como grupos de profissionais que colaboram em diferentes localidades remotas, contrastando com a centralização em um único local físico. A ascensão das equipes distribuídas se tornou evidente durante a pandemia global de 2019, que impulsionou a adoção do trabalho remoto em larga escala como resposta essencial para manter as operações, assegurando a segurança e a saúde dos colaboradores. A prática, antes limitada a países ocidentais, ganhou dimensão global durante a pandemia. Empresas de diversos setores, inclusive do Oriente Médio e da Ásia, adotaram modelos de trabalho remoto e híbrido para garantir a continuidade dos negócios. A flexibilidade do trabalho remoto permitiu a continuação das operações apesar das restrições de deslocamento e *lockdowns* [Khalid et al. 2021].

Conforme a Lei Trabalhista Brasileira, trabalho remoto é a prestação de serviços fora das dependências da empresa, utilizando tecnologias de informação e comunicação [Brasil 2011]. Isso estabelece que empregados em teletrabalho ou home office possuem os mesmos direitos e deveres dos que atuam presencialmente, desde que haja um contrato de trabalho. Ainda, é possível dizer que o trabalho remoto pressupõe a ausência de atividades presenciais, refletindo as normativas atuais [OLIVEIRA et al. 2021].

Como elucidado por Khalid, Mahmood e Farooq (2021), a pandemia do covid-19

ressaltou a crescente importância das equipes distribuídas no desenvolvimento de software. A capacidade de reunir talentos globalmente promove diversidade e permite a formação de equipes especializadas. Líderes e gestores devem adotar estratégias para fomentar colaboração e produtividade nesse novo paradigma.

2.2. Desenvolvimento Ágil em Equipes Distribuídas

Metodologias ágeis como Scrum e Kanban são adotadas por organizações que buscam flexibilidade e eficiência, enfatizando colaboração e comunicação contínua. No entanto, equipes distribuídas enfrentam desafios como a comunicação virtual, que podem causar atrasos e mal-entendidos [Khalid et al. 2021].

Shrivastava e Rathod (2017) investigam essas dificuldades, sugerindo um framework para gerenciamento de riscos e estratégias de mitigação. Essas equipes incluem não apenas profissionais em locais distintos, mas também aqueles de uma mesma localidade que optam pelo trabalho remoto, proporcionando um equilíbrio entre vida profissional e pessoal e benefícios como expansão do talento disponível e redução de custos. Basicamente, esses autores descrevem 5 categorias relacionadas aos riscos encontrados por equipes distribuídas ao adotarem as práticas ágeis. Por sua vez, cada categoria possui um subconjunto de áreas de risco que focam em fatores ainda mais específicos, conforme mostrado na tabela 1.

Categoria de Risco	Ciclo de Vida do Desenvolvimento de Software	Gestão de Projetos	Consciência de Grupo	Colaboração com Partes Interessadas Externas	Configuração de Tecnologia
Área de Risco	Elicitação de Requisitos	Organização do Projeto	Comunicação	Colaboração com o cliente	Infraestrutura e Recursos
Área de Risco	Gestão de Lançamentos	Planejamento e Controle de Projetos	Coordenação e Colaboração	Envolvimento de Múltiplos Fornecedores	Seleção de ferramentas
Área de Risco	Teste e Integração	Início do Projeto	Confiança		
Área de Risco	Design	Aquisição e Desenvolvimento da Equipe do Projeto			
Área de Risco	Análise de Requisitos e Especificação	Gerenciando a Equipe do Projeto			
Área de Risco	Padrões de Cerimônias Ágeis				

Tabela 1. Categorias e Áreas de Risco. Adaptado [Shrivastava and Rathod 2017]

Em resumo, a compreensão desses desafios não apenas identifica áreas de melhoria, mas também fornece insights cruciais para o desenvolvimento de estratégias eficazes de mitigação. Isso contribui significativamente para o avanço contínuo do gerenciamento de projetos ágeis em contextos distribuídos, promovendo uma implementação mais eficiente e bem-sucedida dessas metodologias em ambientes complexos e descentralizados.

2.3. Modelo de Maturidade Ágil (AMM)

O *Agile Maturity Model* (AMM) de Patel e Ramachandran (2009) traz uma estrutura para avaliar e melhorar práticas ágeis de desenvolvimento de software. Para isso, o modelo conta com cinco níveis de maturidade e várias áreas de processo-chave (KPAs), oferecendo um guia para a implementação de melhorias contínuas. Os níveis de maturidade são:

- **Nível 1: Inicial** - As práticas ágeis não são gerenciadas de forma consistente.

- **Nível 2: Gerenciado** - As práticas ágeis são gerenciadas de forma consistente, mas ainda não são totalmente integradas ao processo de desenvolvimento de software.
- **Nível 3: Definido** - As práticas ágeis são totalmente integradas ao processo de desenvolvimento de software e são gerenciadas de forma consistente.
- **Nível 4: Quantitativamente Gerenciado** - As práticas ágeis são gerenciadas de forma quantitativa e são continuamente melhoradas com base em dados e métricas.
- **Nível 5: Otimizado** - As práticas ágeis são continuamente melhoradas e otimizadas para atender às necessidades específicas da organização.

Como dito, o AMM oferece um guia para a implementação de melhorias contínuas. Essa ferramenta é baseada em um ciclo de incrementos que inclui a avaliação, planejamento, implementação e monitoramento das ações de melhoria. E, além do já citado gerenciamento de riscos, cada uma das KPAs incluem outras áreas como o gerenciamento de requisitos, planejamento de iteração e gerenciamento de configuração [Patel and Ramachandran 2009].

Assim sendo, esse modelo passa a ser uma ferramenta valiosa para avaliar e melhorar as práticas ágeis de desenvolvimento de software. Ao utilizá-lo, é possível avaliar o nível de maturidade ágil da equipes distribuídas de desenvolvimento. Entretanto, esta pesquisa considera apenas o nível 2 do AMM a fim de avaliar se uma equipe distribuída é gerenciada ou não.

2.4. Ciclo PDCA (*Plan-Do-Check-Act*)



Figura 1. Ciclo PDCA. Adaptado [Werkema 2013]

O Ciclo PDCA é uma metodologia de gestão de qualidade e melhoria contínua. No desenvolvimento ágil, facilita a avaliação da maturidade ágil e a implementação de melhorias. Ele é composto por quatro etapas essenciais: Planejar (*Plan*), Fazer (*Do*), Verificar (*Check*) e Agir (*Act*) [Pacheco et al. 2012].

No contexto de desenvolvimento, o PDCA desempenha um papel crucial na avaliação da maturidade ágil. Cada etapa do ciclo é aplicada da seguinte forma:

- **Planejar:** Nesta fase, a equipe planeja a avaliação do seu nível de maturidade, definindo o escopo da avaliação, as práticas ágeis a serem avaliadas e as métricas relevantes.
- **Executar:** A equipe coleta dados, realiza entrevistas e observa a implementação das práticas. Essa etapa envolve a medição da conformidade com as procedimentos ágeis estabelecidos.

- **Verificar:** Após a coleta de dados, a equipe verifica os resultados da avaliação em relação aos critérios do modelo de maturidade. Isso auxilia na identificação de lacunas e áreas que requerem melhorias na adoção das práticas ágeis.
- **Agir:** Com base nas descobertas da avaliação, a equipe toma medidas para implementar melhorias. Isso pode envolver ajustes nos processos, treinamento da equipe e aprimoramento das práticas adotadas.

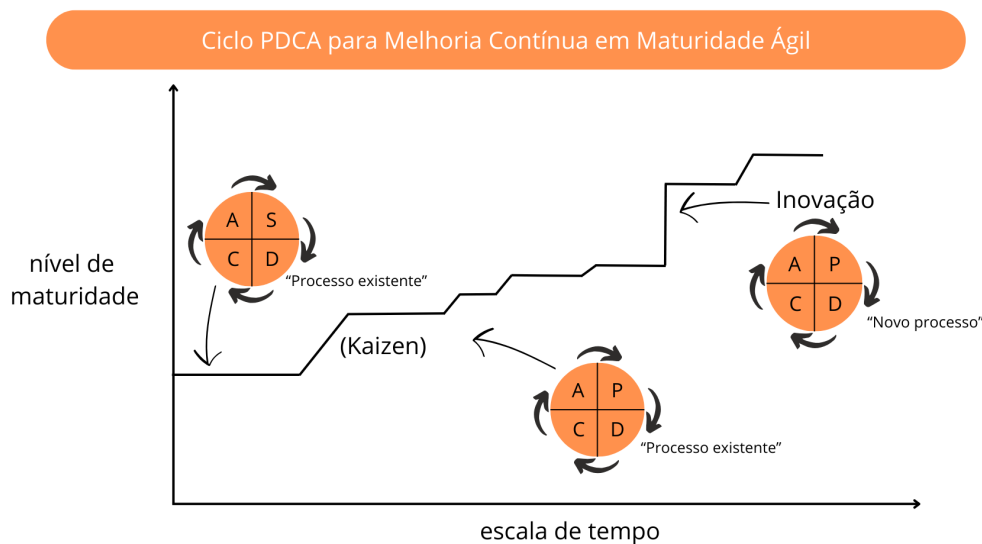


Figura 2. Ciclo PDCA. Adaptado [Werkema 2013]

O Ciclo PDCA é especialmente relevante aqui, pois sua abordagem estruturada permite que as equipes identifiquem, aprimorem e mantenham práticas ágeis e que, assim, elas continuem eficazes.

2.5. Escala Likert

Segundo Feijó et. al. (2020), a escala liker, desenvolvida por Rensis Likert em 1932, é uma ferramenta de pesquisa que mede graus de concordância e é crucial para avaliar a maturidade ágil em ambientes dinâmicos. A classificação dessa escala gera debate quanto à sua natureza, se deve ser tratada como uma escala de dados ordinais ou intervalares. Muitos pesquisadores a utilizam como uma escala de dados intervalares, embora o próprio criador da Escala Likert a tenha concebido como ordinal, particularmente na Psicologia, originando dados qualitativos.

Contudo, a utilização inadequada dessa escala pode comprometer a validade dos resultados da pesquisa, destacando a importância de compreender sua natureza e escolher técnicas estatísticas apropriadas ao trabalhar com dados originados por essa escala [Feijó et al. 2020].

Neste estudo, a Escala Likert desempenha um papel fundamental. Sua abordagem objetiva e quantificável é valiosa para a análise de equipes distribuídas, pois padroniza a avaliação. Ela facilita a observação da evolução da maturidade ágil ao longo do tempo e avalia o impacto das melhorias nas práticas ágeis.

Portanto, a Escala Likert não apenas fornece uma maneira eficaz de medir atitudes e preferências, mas também desempenha um papel crucial na avaliação e melhoria da maturidade ágil em ambientes de trabalho ágeis e dinâmicos.

Normalmente, ela é implementada como uma escala de 5 pontos, indo, por exemplo, de Discordo Totalmente a Concordo Totalmente:

- 1 - Discordo Totalmente
- 2 - Discordo Parcialmente
- 3 - Neutro
- 4 - Concordo Parcialmente
- 5 - Concordo Totalmente

3. Metodologia

Este estudo foi desenvolvido considerando as práticas metodológicas delineadas por Dresch e Antunes Jr. (2015). Considerando que a criação e avaliação de artefatos pode resolver problemas específicos e gerar conhecimento [Dresch et al. 2015], a pesquisa se direcionou para uma metodologia mista, combinando Estudo de Caso e Pesquisa-Ação.

O Estudo de Caso foi aplicado em uma empresa da área de desenvolvimento de software para plataforma de inteligência de dados, que adota trabalho remoto com equipes distribuídas. Isso foi possível pois um grupo de colaboradores aceitou em responder o questionário de avaliação, gerando uma análise empírica e contextual das práticas ágeis em um ambiente natural.

Junto à Pesquisa-Ação, essa abordagem permitiu não apenas a observação, mas também uma participação ativa dos pesquisadores na análise em questão. Através de sugestões de melhoria, será possível gerar um avanço em relação às práticas ágeis consideradas no nível 2 do AMM.

Isto posto, um conjunto de passos foi elaborado para guiar a análise de maneira consistente, simplificada e sequencial:

Passo 1: Desenvolver questões baseadas nas KPA's de nível 2 do AMM, a fim de se avaliar áreas de processo específicas, e relacionadas aos riscos e desafios das 5 categorias do DAD citadas por Shrivastava e Rathod (2017): Ciclo de Vida do Desenvolvimento de Software, Gestão de Projetos e Consciência de Grupo. Elas deverão ser elaboradas de maneira que seja possível, ao entrevistado, respondê-las através opções em escala likert.

Passo 2: Organizar e catalogar todas as respostas pelas KPA's e categorias de risco que cada uma delas atendem.

Passo 3: Estimar, por meio de cálculos, o quão gerenciada a equipe analisada se encontra e apontar as áreas de melhoria específicas identificadas.

Passo 4: Demonstrar a aplicação do ciclo PDCA, limitando-o à sugestões de melhorias práticas para cada área de processo-chave crítica identificada.

Em resumo, aderindo à metodologia estabelecida, a condução do estudo adotou uma abordagem iterativa e reflexiva. Um questionário foi respondido por um grupo de participantes, possibilitando uma compreensão aprofundada do contexto atual. A partir dessa análise, sugestões de melhorias foram elaboradas, alinhadas ao ciclo PDCA[Werkema 2013].

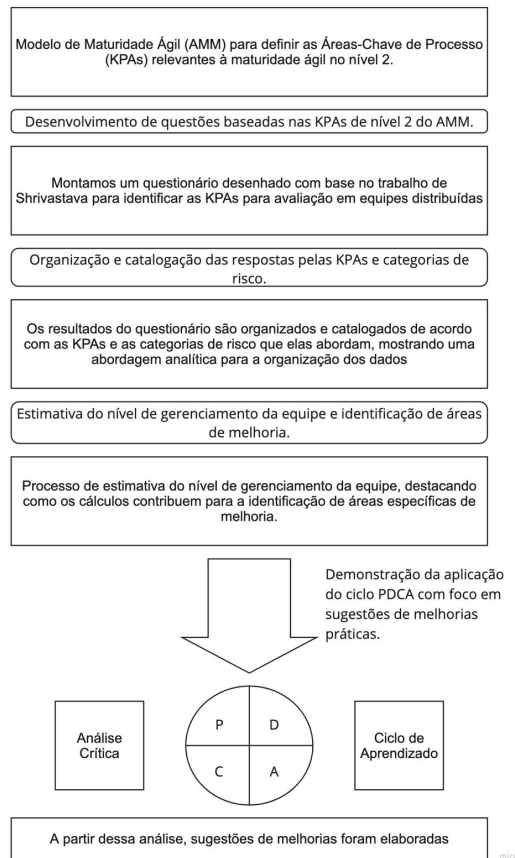


Figura 3. Processo do PDCA. Adaptado [Pagliuso et al. 2010]

4. Resultados e Discussões

4.1. Desenvolvimento do Questionário

Inicialmente, a fim de garantir uma medição estatística precisa e estabelecer um padrão de comparação consistente, o questionário foi pensado para ser respondido através da escala likert de 5 pontos apresentada no tópico 2.5. Para isso, cada questão foi formatada como uma afirmação de maneira que o entrevistado pudesse optar por respondê-la entre Discordo a Concordo Totalmente.

12. A comunicação eficaz é incentivada durante o processo de desenvolvimento, garantindo *
que as equipes entendam claramente os requisitos dos story cards e possam fazer
perguntas para esclarecimentos quando necessário.

1 2 3 4 5

Discordo Totalmente Concordo Totalmente

Figura 4. Questão 12 Disponível no Google Forms

E como abordado na seção de Metodologia, cada afirmativa foi baseada em alguma área de processo-chave (KPA's) de nível 2 específica do AMM. Cada uma delas visa avaliar o respondente considerando as 18 áreas de risco do DAD, englobadas dentro de suas respectivas categorias, conforme ilustrado na Tabela 1.

Atendendo ao passo 1 definido na Metodologia, o processo de elaboração de cada questão da enquete segue exemplificado na imagem abaixo:

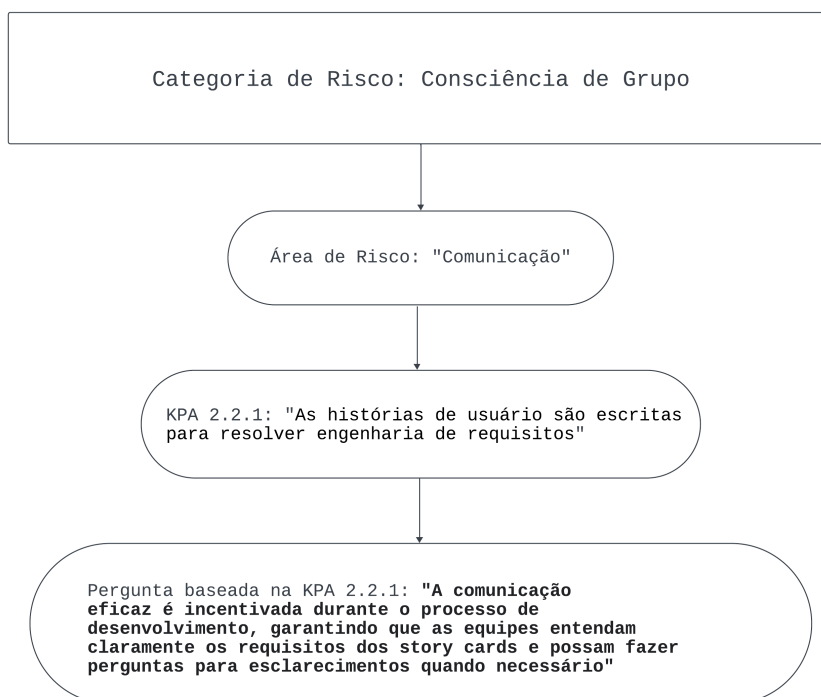


Figura 5. Elaboração da Questão 12

Na figura 5 é apresentada a questão 12 do questionário. Como é possível observar, ela foi criada com base na **KPA 2.2.1** e visa avaliar o entrevistado sob o fator de **Comunicação** relacionado à categoria de **Conscientização de Grupo**.

4.2. Coleta de Dados

Ao todo, foram desenvolvidas 18 questões obrigatórias e 3 opcionais. Duas delas perguntavam ao respondente o nome e o setor da organização em que atuava. Uma última queria saber o grau hierárquico do entrevistado, isto é, seu cargo. O questionário se encontra no seguinte link: <https://docs.google.com/forms/d/e/1FAIpQLSfXDIKXz7yMo38Jupn3A4HB9dnycgd9REwCgk0WyYednNWjyQ/viewform>

Entretanto, a organização em questão não permitiu a divulgação de seu nome para esta pesquisa e, por isso, aqui ela será referida como **empresa x**.

Como plataforma selecionada para a elaboração do questionário, optou-se pelo Google Forms, uma ferramenta de pesquisa online e gratuita. Sua integração eficiente com o Google Sheets simplifica a análise de dados, proporcionando amplas opções para organização, visualização de gráficos e elaboração de fórmulas.

A coleta de dados centrou-se na aplicação do questionário junto a uma amostra de desenvolvedores que atuam dentro de uma equipe distribuída, fornecendo, assim, um contexto sólido.

Essa equipe, composta por aproximadamente 12 colaboradores dedicados à área de tecnologia e desenvolvimento, foi selecionada como o foco da pesquisa. Destaca-se que o questionário foi encaminhado para cada um dos membros por meio do envio do link do questionário via e-mail. Este método de distribuição resultou em uma participação ativa, culminando em 9 respostas completas recebidas. Ou seja, obteve-se taxa de 75% de retorno.



Figura 6. Nível Hierárquico dos Respondentes

Na figura 6, é possível notar que a maioria dos respondentes são, de fato, desenvolvedores. No entanto, três colaboradores fugiram do padrão: um reconheceu-se como *Product Owner*, outro como Gerente e, por fim, um último respondente não se identificou com nenhuma das alternativas e marcou a opção **Outro**.

Destaca-se que o instrumento de pesquisa foi concebido com uma fase preliminar que incorporou um Termo de Consentimento Livre e Esclarecido (TCLE). Durante esta etapa, os participantes foram convidados a revisar os termos e manifestar sua concordância em participar da pesquisa. Aqueles que concordaram tiveram a oportunidade de acessar e preencher a versão completa do instrumento. No entanto, caso expressassem discordância, o questionário era encerrado, assegurando, dessa maneira, a voluntariedade e a conformidade com os princípios éticos da pesquisa.

4.3. Análise de Dados

Atendendo ao passo 2 da metodologia deste estudo, a análise de dados incluiu a categorização das respostas dos participantes em relação às cinco categorias de risco do Modelo de Desenvolvimento Ágil Distribuído (DAD). Para tanto, foi necessário organizar as respostas de cada um dos 9 respondentes para as 18 questões obrigatórias do questionário.

A tabela 2 apresenta todas as respostas obtidas classificadas por respondente, categoria e KPA utilizada como base para se criar todas as questões. Sua imagem foi cortada afim de garantir uma melhor visualização. A ela foi atribuída uma formatação condicional de cores de maneira que tornasse possível indentificar rapidamente os pontos fracos da equipe analisada. Dessa forma, as células em tom de vermelho indicam as repostas mais baixas na escala de 1 a 5. Já as áreas de processo-chave também foram agrupadas em cores para distinguirem-se uma das outras.

Ciclo de Vida do Desenvolvimento de Software						
	QUESTÃO 1	QUESTÃO 2	QUESTÃO 3	QUESTÃO 4	QUESTÃO 5	QUESTÃO 6
Respondente	KPA 2.2.1	KPA 2.1.3	KPA 2.4.2	KPA 2.2.2	KPA 2.2.3	KPA 2.1.8
1	4	3	1	4	4	4
2	4	3	3	3	5	4
3	3	4	4	4	3	4
4	4	2	1	2	4	5
5	3	2	1	2	4	3
6	4	4	3	5	4	4
7	2	2	3	4	2	3
8	3	4	4	4	5	5
9	2	2	3	3	3	3

Tabela 2. Dados por Respondente, KPA e Categoria de Risco

4.4. Estimando o Nível de Maturidade

Após a organização de todos os dados, chegou a hora de calcular os níveis de desempenho e maturidade. Para tal, foram realizados os seguintes cálculos:

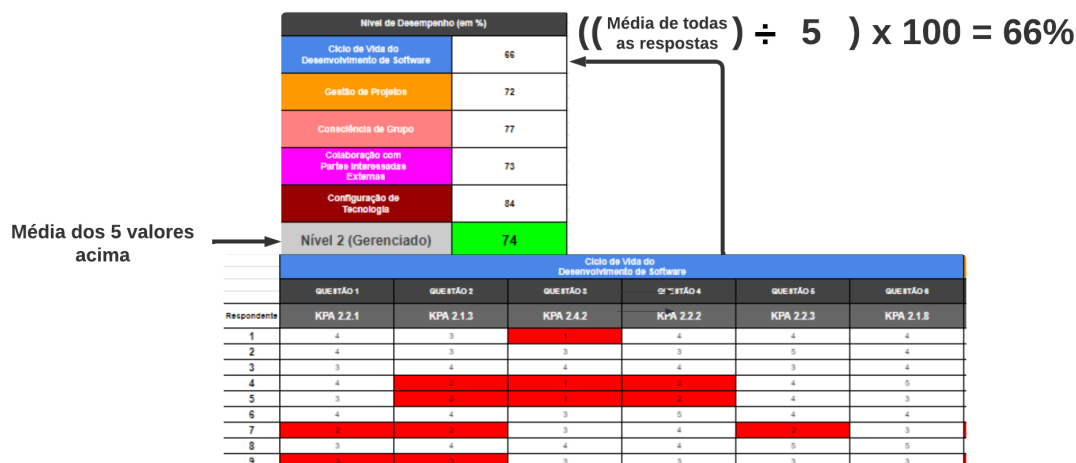


Figura 7. Cálculo dos Valores: Categoria de Ciclo de Vida do Desenvolvimento de Software

Como é possível observar na figura 7, o indicador final de maturidade da equipe analisada resultou em 74%. Para calculá-lo, bastaram apenas alguns cálculos simples para encontrar a média.

Em primeiro lugar, com as respostas do questionário separadas por categorias, calculou-se a média de todos os valores relacionadas à categoria de risco de Ciclo de Vida do Desenvolvimento de Software. Como visto, o valor resultante encontrado foi de 66% de desempenho. Adiante, o mesmo procedimento foi feito para as outras 4 categorias de risco.

E para calcular o indicador final de maturidade, foi calculada a média dos 5 valores encontrados anteriormente.

1. **Média por categoria:** média aritmética dos valores de todas as respostas correspondentes à categoria do DAD em questão. A divisão por 5 é baseada no valor

total da escala de 5 pontos adotada. Por fim, é feita uma multiplicação por 100 a fim de deixar a resposta em porcentagem.

$$\text{Média} = \left(\frac{\text{valores das respostas}}{5} \right) \times 100$$

Figura 8. Média por Categoria

- Indicador Final de Maturidade:** média aritmética dos 5 valores encontrados anteriormente.

Adotando **50%** pontuação mínima, para considerar se a equipe é gerenciada ou não, pode-se afirmar que, com base na Pontuação Final de **74%**, a equipe da **empresa x** analisada apresentou um desempenho alto e está, portanto, enquadrada como gerenciada.

4.5. Aplicação do Ciclo PDCA

Considerando o nível de maturidade estimado da equipe analisada, a aplicação do Ciclo PDCA será direcionada para aprimorar medidas de desempenho específicas.

Entretanto, mesmo que as categorias Ciclo de Vida do Desenvolvimento de Software, Gestão de Projetos e Consciência de Grupo tenham registrado as porcentagens mais baixas de desempenho, esses dados não são cruciais para a identificação das áreas específicas que necessitam de melhoria.

Áreas de Melhoria Específicas					
Ciclo de Vida do Desenvolvimento de Software	KPA 2.2.1	KPA 2.1.3	KPA 2.4.2	KPA 2.2.2	KPA 2.2.3
Gestão de Projetos	KPA 2.1.1	KPA 2.1.2	KPA 2.1.3	KPA 2.1.5	
Consciência de Grupo	KPA 2.3.2	KPA 2.3.3			

Tabela 4. Áreas de Melhoria Identificadas

A abordagem consiste em identificar as perguntas que receberam valores inferiores a 3 (neutro) na escala de 1 a 5, destacadas em vermelho na tabela 2.

Em outras palavras, ao localizar as questões com as maiores taxas de respostas indicando Discordo Totalmente e Discordo Parcialmente, será possível encontrar as áreas de processo críticas que evidenciam os pontos a serem aprimoradas na equipe distribuída analisada. Assim, tendo em mãos essas áreas (KPA's), será possível dar início ao ciclo PDCA, aqui, focado em oferecer sugestões de melhorias práticas:

- **Planejar:**

No estágio de planejamento, são estabelecidas metas para melhorar esses indicadores.

Para a equipe analisada, as metas consistem em medidas de melhoria nas áreas de Ciclo de Vida do Desenvolvimento de Software, Gestão de Projetos e Consciência de Grupo.

- **Executar:**

Na fase de execução, serão implementadas as ações corretivas para melhorar as áreas de processo específicas identificadas, conforme as KPA's. O objetivo é alinhá-las com as práticas ágeis, considerando as particularidades do contexto distribuído.

Sendo assim, seguem sugestões de melhoria para cada uma das áreas de processo críticas identificadas:

- KPA 2.2.1 - As histórias de usuários são escritas para resolver a engenharia de requisitos:

Melhoria Prática:

1. Ferramenta: *Wiki* de Requisitos;
2. Implementar uma *Wiki* de Requisitos: Utilizar uma ferramenta colaborativa, como Confluence, para criar e manter uma *wiki* de requisitos. Isso proporciona transparência e facilita a colaboração entre membros da equipe e *stakeholders* na definição de histórias de usuário.

- KPA 2.1.3 - O planejamento de liberação é usado para criar cronogramas:

Melhoria Prática:

1. Ferramenta: JIRA com Roadmaps;
2. Adotar o JIRA com *Roadmaps*: Utilizar o JIRA para o planejamento de release, aproveitando recursos visuais como *Roadmaps* para facilitar a criação e visualização de cronogramas. Isso permite uma gestão mais eficaz do tempo e recursos.

- KPA 2.4.2 - Os testes unitários são identificados nos cartões de tarefas:

Melhoria Prática:

1. Ferramenta: Trello com Integração de Testes;
2. Integrar Testes ao Trello: Utilizar o Trello para gerenciar tarefas, incorporando campos específicos para identificação de testes de unidade. Integrar com ferramentas de automação de testes, como Selenium, para garantir que os testes estejam alinhados com as tarefas.

- KPA 2.2.2 - A estrutura única do cartão de história é definida:

Melhoria Prática:

1. Ferramenta: Template Padronizado;
2. Desenvolver um Template de Cartão de História: Criar um modelo padronizado no Trello ou JIRA para estruturar cartões de história, incluindo seções específicas para requisitos, critérios de aceitação e informações essenciais. Isso facilita a compreensão e execução consistentes.

- KPA 2.2.3 - Há um plano de saída para gerenciar os cartões de histórias:

Melhoria Prática:

1. Ferramenta: *Checklist* de Critérios de Aceitação no JIRA;

2. Implementar *Checklists* no JIRA: Adicionar *checklists* de critérios de aceitação ao final de cada história no JIRA, proporcionando um plano de saída claro e garantindo que todos os requisitos sejam atendidos antes de avançar.
- KPA 2.1.1 - O *planning game* é usado para criar planos de projeto:

Melhoria Prática:

1. Ferramenta: *Planning Poker Online*;
 2. Adotar o *Planning Poker Online*: Utilizar ferramentas online, como o *Planning Poker* no navegador, para facilitar o jogo de planejamento. Isso permite a participação remota da equipe, promovendo estimativas mais rápidas e precisas.
- KPA 2.1.2 - Estime o escopo do projeto:

Melhoria Prática:

1. Ferramenta: *Planning Poker* Integrado ao JIRA;
 2. Integrar *Planning Poker* ao JIRA: Incorporar plugins ou extensões ao JIRA que ofereçam funcionalidades de *Planning Poker*. Isso simplifica o processo de estimativa e integra as estimativas diretamente às tarefas.
- KPA 2.1.3 - O planejamento de liberação é usado para criar cronogramas:

Melhoria Prática:

1. Ferramenta: Relatórios de Desempenho no JIRA;
 2. Utilizar Relatórios de Desempenho: Criar relatórios específicos no JIRA para monitorar o desempenho do planejamento de release ao longo do tempo. Isso permite uma verificação regular do progresso e ajustes conforme necessário.
- KPA 2.1.5 - O projeto é dividido em iterações:

Melhoria Prática:

1. Ferramenta: Quadro Scrum no Trello;
 2. Adotar o Quadro Scrum no Trello: Utilizar o quadro Scrum no Trello para gerenciar iterações, simplificando a visualização das tarefas e promovendo uma abordagem ágil. Isso facilita a verificação contínua do progresso.
- KPA 2.3.2 - As decisões de negócios são tomadas pelo representante do cliente:

Melhoria Prática:

1. Ferramenta: Videoconferências Estruturadas;
 2. Implementar Videoconferências Estruturadas: Utilizar ferramentas de videoconferência, como Zoom ou Microsoft Teams, para reuniões estruturadas de tomada de decisão. Garantir a participação do cliente de forma proativa em discussões críticas.
- KPA 2.3.3 - Obtenha o compromisso com os cartões de história do cliente no local:

Melhoria Prática:

1. Ferramenta: Plataforma de Compartilhamento de Protótipos;

2. Utilizar Plataforma de Compartilhamento de Protótipos: Adotar ferramentas como Figma ou InVision para criar protótipos visuais de histórias de usuário. Facilitar sessões de comprometimento ao compartilhar protótipos com clientes, garantindo um entendimento claro e comprometimento explícito.

- **Verificar:**

A etapa de verificação envolverá a reaplicação do questionário para avaliar o impacto das mudanças implementadas. Medidas de desempenho serão reavaliadas para verificar a eficácia das intervenções realizadas.

- **Agir:**

Com base na análise dos resultados, a etapa final do PDCA consistirá em agir de maneira decisiva para consolidar as melhorias identificadas. Isso envolverá a consolidação das práticas aprimoradas no cotidiano da equipe, proporcionando treinamento adicional quando necessário e monitorando continuamente o progresso.

A aplicação iterativa do Ciclo PDCA permitirá uma evolução controlada e sustentável no nível de maturidade ágil da equipe, destacando áreas de foco específicas para otimização do desempenho.

5. Considerações Finais

A integração da abordagem quantitativa, aliada à análise qualitativa do Estudo de Caso, proporcionou uma visão abrangente das práticas ágeis em equipes distribuídas. A metodologia empregada identificou áreas específicas para aprimoramento, destacando a efetividade do Ciclo PDCA como estratégia para impulsionar melhorias contínuas no contexto analisado.

Ao analisar os dados obtidos por meio do questionário e do Estudo de Caso, observa-se que os objetivos estabelecidos foram alcançados. A combinação das duas abordagens ofereceu uma compreensão abrangente das dinâmicas ágeis em equipes distribuídas.

As teorias, especialmente aquelas relacionadas ao Ciclo PDCA, desempenharam um papel crucial ao orientar e fundamentar a abordagem metodológica. A eficácia desse ciclo como ferramenta para implementar melhorias foi evidenciada, reforçando a importância de uma abordagem cíclica e adaptativa nas práticas ágeis.

O estudo alcançou seu propósito ao fornecer uma visão aprofundada das práticas ágeis em contextos distribuídos, identificando áreas específicas para aprimoramento e validando a aplicabilidade do Ciclo PDCA nesse cenário. A metodologia adotada, que incluiu a combinação de Estudo de Caso e Pesquisa-Ação, permitiu não apenas a observação, mas também uma participação ativa dos pesquisadores na análise. A abordagem iterativa e reflexiva, junto com a coleta de dados por meio de um questionário respondido por um grupo de participantes, proporcionou uma compreensão aprofundada do contexto atual. A partir dessa análise, foram elaboradas sugestões de melhorias alinhadas ao ciclo PDCA, contribuindo assim para o avanço das práticas ágeis consideradas no nível 2 do AMM.

Reconhecem-se algumas limitações deste estudo, como as restrições temporais que impediram a realização da etapa do ciclo PDCA de avaliação pós-implementação.

Para pesquisas subsequentes, sugere-se a continuidade da investigação nas áreas identificadas como oportunidades de melhoria. Explorar novos métodos de coleta de dados, ampliar a amostra ou aprofundar análises específicas podem enriquecer ainda mais a compreensão das práticas ágeis em ambientes distribuídos.

Em resumo, esta pesquisa não apenas expande o entendimento acadêmico sobre práticas ágeis em equipes distribuídas, mas também oferece insights práticos e sugestões valiosas para futuros trabalhos nesta área. Considerando o foco específico no nível 2 do AMM, destaca-se a relevância deste estudo para avaliar o nível de maturidade ágil em equipes distribuídas e determinar se atendem ao critério de serem consideradas Gerenciadas.

Referências

- Brasil (2011). Lei nº 12.551, de 15 de dezembro de 2011. *Diário Oficial da República Federativa do Brasil*.
- Dresch, A. e Lacerda, D. P. e Antunes Jr, J. A. V. e Dresch, A. e Lacerda, D. P., e Antunes, J. A. V. (2015). *Design science research*. Springer.
- Feijó, A. M. e Vicente, E. F. R., e Petri, S. M. (2020). O uso das escalas likert nas pesquisas de contabilidade. *Revista Gestão Organizacional*, 13(1):27–41.
- Khalid, M. A. B. e Farooq, A., e Mahmood, W. (2021). Communication challenges for distributed teams. *International Journal of Engineering and Manufacturing (IJEM)*, 11(1):19–28.
- OLIVEIRA, M. T. d. e LUZ, L. P. d., et al. (2021). Um novo normal? a adoção do estilo híbrido pelas empresas de tecnologia pós pandemia do covid-19.
- Pacheco, A. P. R. e Salles, B. W. e Garcia, M. A., e Possamai, O. (2012). O ciclo pdca na gestão do conhecimento: uma abordagem sistêmica. *PPGEGC–Universidade Federal de Santa Catarina–Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento–apostila*, 2.
- Pagliuso, A. T. e Cardoso, R., e Spiegel, T. (2010). Gestão organizacional: o desafio da construção do modelo de gestão.
- Patel, C. e Ramachandran, M. (2009). Agile maturity model (amm): a software process improvement framework for agile software development practices. *International Journal of Software Engineering, IJSE*, 2(1):3–28.
- Shrivastava, S. V. e Rathod, U. (2017). A risk management framework for distributed agile projects. *Information and software technology*, 85:1–15.
- Werkema, C. (2013). *Métodos PDCA e Demaic e Suas Ferramentas Analíticas*, volume 1. Elsevier Brasil.