

**UNIVERSIDADE NOVE DE JULHO
PROGRAMA DE PÓS-GRADUAÇÃO EM GESTÃO DE PROJETOS - PPGP
DOUTORADO PROFISSIONAL EM ADMINISTRAÇÃO**

FRAMEWORK DE MÉTRICAS ÁGEIS PARA A LIDERANÇA CORPORATIVA

JOSÉ DA SILVA AZANHA NETO

São Paulo

2023

José da Silva Azanha Neto

FRAMEWORK DE MÉTRICAS ÁGEIS PARA A LIDERANÇA CORPORATIVA

AGILE METRICS FRAMEWORK FOR CORPORATE LEADERSHIP

Tese apresentada ao Programa de Pós-Graduação em Gestão de Projetos da Universidade Nove de Julho – UNINOVE, como requisito parcial para obtenção do grau de Doutor em Administração.

Orientador: Prof. Dr. Renato Penha

São Paulo

2023

DEFESA DE TESE DE DOUTORADO

José da Silva Azanha Neto

Tese apresentada ao Programa de Pós-Graduação em Gestão de Projetos da Universidade Nove de Julho – UNINOVE, Doutorado Profissional em Administração, como requisito parcial para obtenção do grau de **Doutor em Administração**, pela Banca Examinadora, formada por:

São Paulo, 12 de dezembro de 2023.



Presidente: Prof. Dr. Renato Penha (ORIENTADOR)



Membro: Prof. Dr. Fernando Antonio Ribeiro Serra (UNINOVE)



Membro: Prof. Dra. Isabel Cristina Scafuto (UNINOVE)



Membro: Prof. Dr. Flávio Santino Bizarrias (ESPM)



Membro: Prof. Dr. Cláudio Fernando André (PUC-SP)

Azanha Neto, José da Silva.

Framework de métricas ágeis para a liderança corporativa. / José da Silva Azanha Neto. 2023.

295 f.

Tese (Doutorado) - Universidade Nove de Julho - UNINOVE, São Paulo, 2023.

Orientador (a): Prof. Dr. Renato Penha.

.Métricas ágeis. 2. Scrum. 3. Kanban. 4. Lean. 5. Portfólio de Projetos. 6. Liderança de projetos.

.Penha, Renato. II. Título.

CDU 658.012.2

AGRADECIMENTOS

Meus especiais agradecimentos ao meu orientador e professor Dr. Renato Penha com apoio incondicional nessa jornada.

Aos professores Dr. Luciano Ferreira da Silva e Dr. Paulo Sérgio Gonçalves de Oliveira que me ajudaram muito durante a etapa de qualificação.

Aos professores Dra. Isabel Cristina Scafuto, Dr. Flávio Santino Bizarrias, Dr. Cláudio Fernando André e Dr. Fernando Antonio Ribeiro Serra que fizeram parte da minha banca de defesa e trouxeram grandes contribuições a este trabalho de tese.

Aos meus amigos e profissionais do setor de Gerenciamento de Projetos Vlamir F. Silva, Gabriel Baptista, Eder Eduardo, Agnaldo Trindade, Bruno M. Domingues e Josué Silva que também contribuíram muito com esse trabalho.

A Uninove e corpo docente do *Strictu Sensu* pelos ensinamentos e por me proporcionar essa conquista importante na minha vida pessoal, acadêmica e profissional.

*“Estamos descobrindo maneiras melhores de desenvolver software,
fazendo-o nós mesmos e ajudando outros a fazerem o mesmo”.*

Manifesto para Desenvolvimento Ágil de Software

RESUMO

Em um ambiente organizacional onde projetos ágeis são medidos pela liderança a partir de métricas tradicionais, abre-se uma oportunidade de pesquisa que pode ser explorada. Assim, o objetivo da tese é de propor um *framework* de métricas ágeis de maneira a orientar a liderança na gestão de projetos ágeis. A pesquisa seguiu os procedimentos metodológicos baseados em quatro estudos. O Estudo 1 é sobre como a liderança aborda a gestão ágil de projetos e utilizou a revisão sistemática da literatura como método. O Estudo 2 é sobre as métricas utilizadas em projetos tradicionais e projetos ágeis, e utilizou a revisão sistemática da literatura como método. O Estudo 3 é sobre o levantamento de publicações sobre métricas e *frameworks* nas bases de patente e utilizou a análise exploratória como método. O Estudo 4 é a realização de entrevistas com a abordagem do *focus group* e a pesquisa qualitativa como método. Como resultados dos quatro estudos, verificou-se que ainda existem obstáculos para implementar a abordagem ágil nas organizações, mas que práticas baseadas na liderança ágil e na liderança corporativa podem auxiliar no desenvolvimento ágil nos projetos. Além disso, percebeu-se que as métricas tradicionais focaram em apresentar a métrica do *Earned Value Management* com a adoção de melhorias na métrica. Nas métricas ágeis, percebeu-se que as métricas como *Lead Time*, *Cycle Time*, Qualidade e *Velocity* são as mais citadas e que se concentraram no *discovery* e no *delivery*. Não foram encontradas soluções relevantes sobre processos ou *frameworks* para métricas ágeis nas bases de patente. O *framework* foi validado com praticantes de gestão de projetos em diferentes organizações. Assim, diversas alterações e contribuições trouxeram maior transparência, visibilidade e robustez ao *Framework* de Métricas Ágeis em diferentes organizações, níveis de maturidade e escala em projetos, além de se aderente às práticas ágeis como *Design Sprint*, *Lean Inception*, *Design Thinking*, *Scrum*, *Kanban*, *Extreme Programming* e *DevOps*. Espera-se contribuir para apresentar às organizações um *framework* capaz de fornecer as métricas ágeis em todas as etapas do projeto para a liderança corporativa. Além de fornecer métricas de projeto, espera-se que as métricas de *feedback* de clientes, *stakeholders* e do mercado possam auxiliar as organizações, a liderança e equipes do projeto na tomada de decisão sobre projetos ágeis. Esta pesquisa está alinhada à linha de pesquisa 2 “Gerenciamento de Projetos”, vinculado ao programa de Pós-Graduação em Gestão de Projetos – PPGP UNINOVE. A presente dissertação também está alinhada com outras pesquisas de seu orientador vinculadas ao projeto-eixo “Projetos Ágeis e Híbridos”.

Palavras-chave: Métricas Ágeis; *Scrum*; *Kanban*; *Lean*; Portfólio de Projetos, Liderança de Projetos.

ABSTRACT

In an organizational environment where agile projects are measured by leadership based on traditional metrics, a research opportunity opens up that can be explored. Thus, the objective of the thesis is to propose an agile metrics framework in order to guide leadership in the management of agile projects. The research followed methodological procedures based on four studies. Study 1 is about how leadership approaches agile project management and used systematic literature review as a method. Study 2 is about the metrics used in traditional projects and agile projects, and used systematic literature review as a method. Study 3 is about surveying publications on metrics and frameworks in patent databases and used exploratory analysis as a method. Study 4 involves conducting interviews using a focus group approach and qualitative research as a method. As a result of the four studies, it was found that there are still obstacles to implementing the agile approach in organizations, but that practices based on agile leadership and corporate leadership can help with agile development in projects. Furthermore, it was noticed that traditional metrics focused on presenting the Earned Value Management metric with the adoption of improvements in the metric. In agile metrics, it was noticed that metrics such as Lead Time, Cycle Time, Quality and Velocity are the most cited and that they focused on discovery and delivery. No relevant solutions on processes or frameworks for agile metrics were found in the patent bases. The framework was validated with project management practitioners in different organizations. Thus, several changes and contributions brought greater transparency, visibility and robustness to the Agile Metrics Framework in different organizations, maturity levels and scale in projects, in addition to adhering to agile practices such as Design Sprint, Lean Inception, Design Thinking, Scrum, Kanban, Extreme Programming and DevOps. It is expected to contribute to presenting organizations with a framework capable of providing agile metrics at all stages of the project for corporate leadership. In addition to providing project metrics, it is expected that feedback metrics from customers, stakeholders and the market can assist organizations, leadership and project teams in making decisions about agile projects. This research is aligned with research line 2 “Project Management”, linked to the Postgraduate Program in Project Management – PPGP UNINOVE. This dissertation is also aligned with other research by its advisor linked to the “Agile and Hybrid Projects” axis project.

Keywords: Agile Metrics; *Scrum*; *Kanban*; *Lean*; Project Portfolio; Project Leadership.

LISTA DE ABREVIATURAS E SIGLAS

AAO	– <i>Agile Architecture Owner</i>
ABM	– <i>Agile Business Manager</i>
ABO	– <i>Agile Business Owner</i>
AC	– <i>Actual Cost</i>
AC	– <i>Agile Coach</i>
ACV	– <i>Actual Cost value</i>
AIG	– <i>Árvores Impulsionadas por Gradiente</i>
AMO	– <i>Agile Metrics Owner</i>
APO1	– <i>Agile Portfolio Owner</i>
APO2	– <i>Agile Program Owner</i>
ASD	– <i>Agile Software Developer</i>
BC	– <i>Budgeted Cost</i>
BDD	– <i>Behavior Driven Development</i>
BDUF	– <i>Big Design Up Front</i>
CBO	– <i>Coupling between Objects</i>
CEAC	– <i>Cost Estimate at Completion</i>
CK	– <i>Chidamber-Kemerer</i>
CPI	– <i>Cost Performance Index</i>
CSF-Live!	– <i>Critical Success Factors</i>
CT	– <i>Cycle Time</i>
CV	– <i>Coefficient of Variation</i>
DevOps	– <i>Desenvolvimento e Operações</i>
EAP	– <i>Estrutura Analítica do Projeto</i>
EDM	– <i>Earned Duration Management</i>
	<i>Enterprise Release Risk-Centric Systems Development and Maintenance Life</i>
ER2C SDMLC	– <i>Cycle</i>
EV	– <i>Earned Value</i>
EVM	– <i>Earned Value Management</i>
EVM/ES	– <i>Earned Value Management/Earned Schedule</i>
FMA	– <i>Framework de Métricas Ágeis</i>
FS	– <i>Framework Scrum</i>
GP	– <i>Gerenciamento de Projetos</i>
GPE	– <i>Gerenciamento de Projetos Empresariais</i>
HU	– <i>Histórias de Usuários</i>
HyEEASe	– <i>Hybrid Effort Estimation in Agile Software Development</i>
ICO	– <i>Initial Coin Offering</i>
INPI	– <i>Instituto Nacional da Propriedade Industrial</i>
KPI	– <i>Key Performance Indicators</i>
LT	– <i>Lead Time</i>
MG	– <i>Métricas Geradas</i>
MK	– <i>Método Kanban</i>
MM	– <i>Matriz Metodológica</i>
MV	– <i>Métricas Visualizadas</i>
MVP	– <i>Minimum Viable Product</i>

NPS	– <i>Net Promoter Score</i>
NTDD	– <i>NON-Test-Driven Development</i>
OKR	– <i>Objective and Key Results.</i>
PDCA	– <i>Plan, Do, Check, Act/Ajust</i>
PF	– <i>Propensão as Falhas</i>
PK	– <i>Planning Poker</i>
PMI	– <i>Project Management Institute</i>
PO	– <i>Product Owner</i>
PV	– <i>Planned Value</i>
RC	– <i>Resource Constrained</i>
RES	– <i>Registro Eletrônico de Saúde</i>
RFC	– <i>Response for a Class</i>
RIC	– <i>Reunião de Integração Contínua</i>
RICT	– <i>Reunião de Integração Contínua dos Times</i>
RM	– <i>Reuniões de Métricas</i>
RSL	– <i>Revisão Sistemática da Literatura</i>
SAC	– <i>Sincronização entre Agile Coaches</i>
SBO	– <i>Sincronização entre Business Owners</i>
SLOC	– <i>Source Lines of Code</i>
SLR	– <i>Systematic Literature Review</i>
SM	– <i>Scrum Master</i>
SOA	– <i>Service-Oriented Architecture</i>
SPI	– <i>Schedule Performance Index</i>
TAL	– <i>The Agile Landscape</i>
TDD	– <i>Test-Driven Development</i>
UCP	– <i>Use Case Point</i>
USRA	– <i>Using Schedule Risk Analysis</i>
VUCA	– <i>Volatility, Uncertainty, Complexity e Ambiguity</i>
WIA	– <i>Work Item Age</i>
WIP	– <i>Work In Progress</i>
WMC	– <i>Weighted Methods per Class</i>
WP	– <i>Work Perform</i>
XP	– <i>Extreming Programming</i>

LISTA DE TABELAS

Tabela 1- Matriz Metodologia da Tese.....	23
Tabela 2- Papéis do Framework de Métricas Ágeis (FMA) na etapa de Continuous Initiate	145
Tabela 3- Papéis do Framework de Métricas Ágeis (FMA) na etapa de Continuous Discovery	146
Tabela 4- Papéis do Framework de Métricas Ágeis (FMA) na etapa de Continuous Delivery	149
Tabela 5- Papéis do Framework de Métricas Ágeis (FMA) na etapa de Continuous Release	150
Tabela 6- Papéis do Framework de Métricas Ágeis (FMA) para a etapa de Continuous Improvement.....	153
Tabela 7- Estratégias e papéis do Framework de Métricas Ágeis (FMA) para a etapa de Continuous Improvement.....	153
Tabela 8- Níveis de adoção do Framework de Métricas Ágeis (FMA) em dois níveis de propósitos organizacionais.....	155
Tabela 9- Níveis de adoção do Framework de Métricas Ágeis (FMA) em diferentes modelos organizacionais do Projeto Ágil.....	155
Tabela 10- Cerimônias do Framework de Métricas Ágeis (FMA).....	159
Tabela 11- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com as Cerimônias do Framework Scrum (FS).....	162
Tabela 12- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com as Cerimônias do Método Kanban (MK).....	162
Tabela 13- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com os processos do PDCA.....	164
Tabela 14- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com as etapas do Objectives Key-Results (OKR).....	164
Tabela 15- Os papéis do <i>Framework</i> de Métricas Ágeis (FMA) e o relacionamento de cada papel nas etapas do FMA.....	165
Tabela 16- Artefatos por Etapas do Framework de Métricas Ágeis (FMA).....	167
Tabela 17- Métricas rastreáveis por Etapas e Categorização como ferramenta gerencial do Framework de Métricas Ágeis (FMA). Medida Gerada (MG); Métrica Visualizada (MV). .	168
Tabela 18- Detalhamento das Métricas do FMA com a referência dos estudos realizados nessa pesquisa, principais interessados, etapas e descrição.....	175
Tabela 19- Relacionamento das Métricas encontradas no Estudo 2, separadas pelas etapas de Discovery e Delivery, objetivo de cada métrica e autores.....	270

LISTA DE FIGURAS

Figura 1 - Desenho da pesquisa da tese por estudos.....	29
Figura 2 - Fluxo da etapa <i>Continuous Initiate</i> para a proposta de <i>Framework</i> de Métricas Ágeis (FMA).....	144
Figura 3 - Fluxo da etapa <i>Continuous Discovery</i> para a proposta de <i>Framework</i> de Métricas Ágeis (FMA).....	146
Figura 4 - Fluxo da etapa <i>Continuous Delivery</i> para a proposta de <i>Framework</i> de Métricas Ágeis (FMA).....	148
Figura 5 - Fluxo da etapa <i>Continuous Release</i> para a proposta de <i>Framework</i> de Métricas Ágeis (FMA).....	150
Figura 6 - Fluxo da etapa <i>Continuous Improvement</i> para a proposta de <i>Framework</i> de Métricas Ágeis (FMA).....	152
Figura 7 - Fluxo de processo do <i>Framework</i> de Métricas Ágeis (FMA).....	157
Figura 8 - Fluxo detalhado do <i>Framework</i> de Métricas Ágeis (FMA).....	158
Figura 9 - Pedido de Patente de invenção.....	187

SUMÁRIO

	LISTA DE ABREVIATURAS E SIGLAS.....	9
	LISTA DE TABELAS.....	11
	LISTA DE FIGURAS.....	12
1	INTRODUÇÃO.....	10
1.1	PROBLEMA DE PESQUISA.....	18
1.2	OBJETIVOS.....	20
1.2.1	OBJETIVO GERAL.....	20
1.2.2	OBJETIVOS ESPECÍFICOS.....	20
1.3	JUSTIFICATIVA.....	21
1.4	ESTRUTURA DA TESE.....	23
2	PROCEDIMENTOS METODOLÓGICOS.....	28
2.1	DESENHO DA PESQUISA.....	28
3	ESTUDO 1: A IMPORTÂNCIA DA LIDERANÇA EM PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA.....	31
4	ESTUDO 2: MAPEAMENTO DE MÉTRICAS PARA DESEMPENHO DE PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA.....	68
5	ESTUDO 3: ESTUDO DE PATENTES SOBRE MÉTRICAS PARA A GESTÃO ÁGIL DE PROJETOS.....	110
6	ESTUDO 4: FMA: <i>FRAMEWORK</i> DE MÉTRICAS ÁGEIS COMO FERRAMENTA DE APOIO A GESTÃO ÁGIL DE PROJETOS.....	123
7	PROCEDIMENTOS METODOLÓGICOS PARA PRODUTO TECNOLÓGICO.....	142
7.1	DESENHO DA PESQUISA.....	142
7.2	CONTINUOUS INITIATE.....	143

7.3	CONTINUOUS DISCOVERY.....	145
7.4	CONTINUOUS DELIVERY.....	147
7.5	CONTINUOUS RELEASE.....	149
7.6	CONTINUOUS IMPROVEMENT.....	151
7.7	FRAMEWORK DE MÉTRICAS.....	154
8	PRODUTO TÉCNICO TECNOLÓGICO.....	185
9	CONCLUSÃO.....	187
10	APÊNDICE.....	192
10.1	THE IMPORTANCE OF LEADERSHIP IN AGILE PROJECTS: SYSTEMATIC LITERATURE REVIEW.....	192
10.2	MAPEAMENTO DE MÉTRICAS PARA DESEMPENHO DE PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA.....	210
10.3	ESTUDO DE PATENTES SOBRE MÉTRICAS PARA A GESTÃO ÁGIL DE PROJETOS.....	256
11	ANEXO I.....	270
11.1	TABELA ADICIONAL DO ESTUDO 2.....	270

1 INTRODUÇÃO

A relação entre a estrutura das organizações e a tecnologia empregada tem sido amplamente discutida, comprovando-se como um fator fundamental que influenciou a evolução da estrutura organizacional ao longo dos anos (Child & Mansfield, 1972). Outros aspectos podem exercer algum tipo de influência nessa relação, como o tamanho e a estrutura organizacional (Hickman, 1998) e a adoção e uso de tecnologia frente a produtividade dos recursos humanos (Daft, 2015). Em relação à tecnologia, as organizações podem sofrer impactos em suas funções relacionadas ao desenvolvimento de atividades internas como, por exemplo, o controle de linha de produção (Child & Mansfield, 1972).

Nesse contexto, Child (1973) afirmou que o tamanho da organização, a tecnologia, a localização e as variáveis ambientais influenciam a complexidade organizacional. A relação entre a complexidade e a formalização pode ser considerada mais direta do que a relação com o tamanho da organização. Child (1973) também destacou que o tamanho da organização é um indicador importante da descentralização, e não pode ser totalmente prevista ou compreendida sem considerar a economia de escala.

Segundo Hickman (1998) e Galbraith (2012), o primeiro fenômeno do processo de concatenação da tecnologia e estrutura organizacional envolve a complexidade e a interdependência que estão cada vez maiores à medida que as empresas adicionam novas ênfases estratégicas e as incorporam em sua estrutura. Galbraith (2012) afirmou que na medida em que aumenta o número e a variedade de entidades relevantes no ambiente dos *stakeholders*, o número e a variedade de unidades dentro da empresa devem aumentar para gerenciar essas entidades.

No início do século XX, como relataram Hickman (1998) e Galbraith (2012), a maioria das grandes empresas eram verticalmente integradas e organizadas em estruturas funcionais, outras tantas dessas empresas se diversificaram e se organizaram na estrutura multidivisional. Sobre a diversificação, Galbraith (2012) afirmou que, na década de 1990, empresas como IBM, Procter e Gamble passaram a focar fortemente nos clientes, acrescentando uma quarta dimensão estratégica que precisava ser incorporada à organização. Galbraith (2012) destacou que em cada região, a estrutura era composta por equipes de unidades de negócios que também se reportam às unidades de negócios globais, onde cada equipe de negócios regional é organizada em torno de funções, que também se reportam às suas unidades funcionais no topo da estrutura de equipe global. Desse modo, Galbraith (2012)

compreendeu que as empresas estavam começando a agregar seus bancos de dados independentes e, usando mecanismos de pesquisa, algoritmos analíticos, para produzir *insights* valiosos a partir de vários dados.

Em sua pesquisa, Galbraith (2012) compreendeu que as questões de coordenação e complexidade estão presentes em todos os tipos de empresas e à medida que o número de diferentes tipos de unidades em uma empresa aumenta e a interdependência entre elas também aumenta, a organização deve processar mais informações. Galbraith (2012) sustentou que uma organização poderia descentralizar a interdependência empregando formas verticais de coordenação que variam de simples relações informais a equipes formais e, finalmente, a processos matriciais complexos.

Para Galbraith (2012), o lado humano da organização foi redesenhado e, em muitas empresas, a ênfase é colocada no desenvolvimento de valores compartilhados que orientam as decisões sem comunicação entre unidades interdependentes e gerentes. Por fim, Galbraith (2012) concluiu que uma cultura de colaboração impulsiona muitos dos processos para gerenciar a interdependência, e as atribuições rotativas são usadas para desenvolver os gerentes que conhecem e se identificam com a organização como um todo.

Em relação às práticas de gestão, Miterev, Mancini e Turner (2017) destacaram a importância da organização baseada em projetos nos processos gerenciais. Os autores observaram que algumas organizações abandonaram completamente a estrutura hierárquica funcional em favor de uma abordagem totalmente orientada a projetos. No entanto, ao fazer essa transição, essas organizações podem perder os benefícios oferecidos pela hierarquia funcional e não consideraram devidamente como substituí-los no contexto do trabalho baseado em projetos.

Diante desse cenário, Miterev, Mancini e Turner (2017) afirmam que é melhor para as organizações manterem a hierarquia funcional e encontrar maneiras de fazer com que a hierarquia funcional e as estruturas de projetos trabalhem juntas. Na mesma linha, Moodley, Sutherland e Pretorius (2016) destacam que nas últimas quatro décadas houve uma ampla adoção do desenho organizacional matricial, especialmente em organizações baseadas em projetos. Um dos principais desafios nesse sentido está relacionado à ambiguidade de autoridade resultante da dupla cadeia de comando, mesmo dentro da hierarquia funcional. De acordo com Miterev, Mancini e Turner (2017), as organizações efetuam o processo de tomada de decisão estratégica respeitando a cadeia hierárquica ao se adotar o trabalho baseado em projetos.

Organizações baseadas em projetos referem-se a uma variedade de formas organizacionais que envolvem a criação de sistemas temporários estruturados para realizar tarefas planejadas do projeto (Thiry & Deguire, 2007). Thiry e Deguire (2007) ainda destacam que organizações baseadas em projetos têm recebido cada vez mais atenção nos últimos anos como uma forma emergente de organização e é necessário adotar abordagens que integrem e permitam estruturas alinhadas com a estratégia organizacional. No entanto, organizações que adotam o trabalho baseado em projetos reconhecem a necessidade de fornecer funções de integração e sugerem o uso de estruturas orientadas a projetos, como portfólios de projetos e escritórios de gerenciamento de projetos (Miterev, Mancini & Turner, 2017). Por fim, de acordo com Miterev, Mancini e Turner (2017), as organizações baseadas em projetos empregam o gerenciamento de organizações temporárias na forma de projetos, portfólio e programas como seus principais processos de negócio, apresentando-se como uma nova forma de organização caracterizada por mudanças constantes no *design* organizacional.

Em relação à estratégia organizacional, Miterev, Mancini e Turner (2017) apresentaram um modelo de quatro elementos composto pela (i) estrutura, (ii) processo, (iii) pessoas e (iv) produtos. A estrutura refere-se a estrutura organizacional que compõem departamentos e cargos. O processo corresponde aos procedimentos que determinam como a organização funciona. As pessoas são responsáveis por fazer a máquina da estrutura e processos girar e, por fim, os produtos estão vinculados ao valor que a organização pretende fornecer aos seus *stakeholders* externos e a sua relação com a estratégia (Miterev, Mancini & Turner 2017).

Além disso, Miterev, Mancini e Turner (2017) afirmaram que na literatura de gerenciamento de projetos, existe um suporte significativo para a proposição de que os cinco elementos significativos no *design* da organização baseada em projetos são estratégia, estrutura, processo, comportamento e pessoas. Segundo Tlhomelang e Barry (2008), grandes empresas em economias desenvolvidas têm aplicado o gerenciamento de projetos empresariais (GPE) e este artigo apresentou os resultados de um estudo realizado para investigar os benefícios potenciais da implementação de GPE em organizações de médio a grande porte. Assim, Tlhomelang e Barry (2008) compreenderam que exigiria da organização um programa acelerado de desenvolvimento das capacidades da força de trabalho e da competência organizacional necessária para gerenciar os negócios. No entanto, Miterev, Mancini e Turner (2017) mostraram que em muitas organizações que entregavam produtos ou serviços sob medida para clientes e a hierarquia funcional permaneceu dominante. Para

contribuir no processo gerencial, Miterev, Mancini e Turner (2017) sugeriram que a estrutura de governança deveria estar alinhada com o projeto, ou seja, na organização baseada em projetos, o gerenciamento de projetos, programas e portfólios são os processos de negócios adotados para entregar produtos e serviços aos clientes. Thiry e Deguire (2007) afirmaram que organizações maduras baseadas em projetos precisam adotar abordagens integradoras que permitam estruturas consistentes, entrega de estratégia e uniformização do conhecimento.

A proximidade do cliente automaticamente contribuiu para que as organizações adaptassem suas práticas gerenciais (Beck *et al.*, 2021). A adaptabilidade, segundo Kristensen e Shafiee (2019), é focar no futuro, na capacidade de responder às mudanças, ser ágil e progredir. Em se tratando das práticas gerenciais, Kristensen e Shafiee (2019) definiram que ser verdadeiramente ágil requer diferentes formas de gestão do conhecimento, liderança, comunicação, tomada de decisão, governança, organização de equipes, alocação de papéis, alinhamento e mentalidade organizacional. Para Kristensen e Shafiee (2019), as organizações podem achar a transformação ágil altamente disruptiva e uma remodelação, ou um ajuste natural ao seu *design* atual, e um retorno aos seus valores fundamentais com uma vantagem competitiva significativa.

Nesse sentido, Kristensen e Shafiee (2019) e Adkins (2020) definem que alcançar a agilidade é um fator-chave para as organizações que desejam reduzir o tempo de lançamento no mercado, promover a inovação e lidar com a complexidade. Entretanto, Kristensen e Shafiee (2019) e Adkins (2020) ainda afirmaram que existem organizações que aplicam práticas organizacionais que não são compatíveis com o objetivo de lidar com um ambiente organizacional dinâmico e um mercado em constante mudança. Internamente nas organizações, Kristensen e Shafiee (2019) e Adkins (2020) destacaram que muitas empresas fizeram uma transição dos modelos tradicionais para o *framework Scrum* com o objetivo de se tornarem organizações ágeis. Porém, Sutherland (2019) e Rubin (2017) afirmaram que o *Scrum* é uma estrutura de desenvolvimento de produtos e a sua implementação não é suficiente para se tornar verdadeiramente ágil, pois as organizações simplesmente ignoraram as mudanças necessárias na estrutura organizacional, em processos e governança, e na atuação de gestão da liderança organizacional.

No contexto das características organizacionais, Kristensen e Shafiee (2019) trouxeram que a investigação da relação entre as características do ambiente e das organizações determinou dois tipos principais de *design* organizacional: o mecanicista e o orgânico. Os resultados apontados por Kristensen e Shafiee (2019), mostraram que em

ambientes relativamente estáveis e previsíveis, as organizações tendem a ter um *design* mecanicista, como Miterev, Mancini e Turner (2017) como um gerenciamento funcional, hierárquico ou de linha. Este tipo de organização, segundo Kristensen e Shafiee (2019), tem uma estrutura altamente hierárquica e operação de gestão formal com autoridade centralizada, muitas regras e procedimentos formais, divisão precisa do trabalho, estreita amplitude de controle e meios formais de coordenação. As organizações que operam no ambiente *Volatility, Uncertainty, Complexity e Ambiguity* (VUCA), conforme Kristensen e Shafiee (2019), devem ter um *design* orgânico, menos formal, menos hierárquico e menos mecanicista. Assim, do ponto de vista de Kristensen e Shafiee (2019), vivemos uma mudança de paradigma nas formas com que as organizações alcançam para equilibrar estabilidade e o dinamismo, pois é uma mudança clara na direção da metáfora da máquina para a metáfora de um organismo vivo. Nesse sentido, Kristensen e Shafiee (2019) afirmaram que as organizações ágeis como um organismo vivo se mobilizam rapidamente, capacitadas para agir e facilitam a ação, pois evoluíram para prosperar em um ambiente imprevisível e em rápida mudança.

A nova organização ágil, conforme Kristensen e Shafiee (2019), foi projetada para (a) cocriar valor para todas as partes interessadas, reconhecendo a abundância de oportunidades e recursos disponíveis; (b) permitir que os funcionários sejam altamente engajados, cuidem uns dos outros, encontrem soluções engenhosas e forneçam resultados excepcionais por meio de responsabilidades claras; (c) abraçar a incerteza e seja o mais rápido e produtivo ao tentar coisas novas para minimizar os riscos; (d) evoluir continuamente e abraçar a incerteza e a ambiguidade; e (e) capacitar os funcionários a assumirem total responsabilidade, confiantes de que conduzirão a organização a cumprir seus propósitos e visão.

No entendimento de Kristensen e Shafiee (2019), as organizações ágeis possuem estrutura não complexa, transparente e formalizada, pois em vez de uma organização matricial complexa, comumente em organizações baseadas em projetos (Moodley, Sutherland & Pretorius, 2016), uma organização ágil tem uma configuração fácil de entender com uma lógica semelhante entre departamentos e escritórios regionais. Segundo Kristensen e Shafiee (2019), isso permite que líderes, equipes e toda a organização antecipem mudanças na organização em resposta ao ambiente e cria um roteiro claro no qual as funções evoluem para se tornarem comunidades robustas de conhecimento e prática, permitindo que as organizações desenvolvam profundidade, especialização e proporcionando estabilidade e continuidade ao

longo do tempo à medida que as pessoas alternam entre diferentes equipes operacionais do projeto.

Por fim, Kristensen e Shafiee (2019) e Adkins (2020) destacaram que outra alavanca para aumentar a agilidade é a boa governança prática como um processo contínuo que acontece em nível de equipe por equipe e cria um ponto de interação. Kristensen e Shafiee (2019) e Adkins (2020) complementaram que as equipes relevantes obtêm os direitos à decisão e tomam as decisões rapidamente em fóruns de coordenação altamente produtivos. Rachman e Ratnayake (2018) definiram quatro elementos principais para a produtividade como a liderança e comprometimento da gestão, envolvimento dos funcionários, cooperação e confiança com fornecedores e gerenciamento enxuto de projetos com a utilização de práticas *Lean*. Acrescentam Kristensen e Shafiee (2019) que a liderança compartilhada e servidora é um aspecto crítico da cultura organizacional mais relacionado à agilidade, assim o ambiente cultural precisa ser fomentado por um impulso empreendedor com pessoas motivadas para aprender e inovar. Nesse sentido, segundo Kristensen e Shafiee (2019), isso exige que essas empresas evoluam contínua e rapidamente seus processos operacionais, considerando sua arquitetura, interfaces e ferramentas de tecnologia a partir de projetos de desenvolvimento de tecnologia para inovar seus produtos e serviços aos seus clientes e *stakeholders*.

Especificamente nas organizações baseadas em projetos de desenvolvimento de *software*, Sommerville (2007) e Junior e Junior (2019) destacaram que, inicialmente, os projetos são baseados nos tradicionais modelos oriundos da Engenharia Civil, onde o plano contendo a maior parte do planejamento do projeto envolvendo, principalmente, escopo, custo, prazo, qualidade, aquisições e recursos humanos. Para Sommerville (2007) e Junior e Junior (2019), essa prática de gestão de projetos foi denominada como tradicional.

Para Sutherland (2019), o grande desafio em projetos de desenvolvimento de *software* utilizando o modelo tradicional está relacionado ao gerenciamento das mudanças de escopo durante o ciclo de vida de um projeto. A justificativa, conforme Sutherland (2019), é que clientes e *stakeholders* não conseguiam prever todo o escopo nas fases de levantamento de requisitos do projeto. Como agravante, Junior e Junior (2019) e Van Casteren (2017) destacaram que a velocidade com que as organizações mudavam o escopo de seus projetos passou de anos para semanas ou até dias. Nesse sentido, Ries (2011) afirmou que, em um ambiente dos negócios ambíguo e complexo como o de negócios baseados em Tecnologia da Informação, o modelo tradicional deixou de fazer sentido em projetos de desenvolvimento de

software devido às incertezas e constantes mudanças de requisitos que ocorrem nesses ambientes.

Diante desse contexto marcado pela necessidade de adaptação nas práticas gerenciais em projetos de desenvolvimento de *software* que, segundo o *Agile Manifesto* (2001), alguns engenheiros de *software* se reuniram na década de 1990 para discutir sobre a maneira como os projetos desse segmento eram conduzidos. O resultado foi a criação de um evento conhecimento mundialmente por Manifesto Ágil para o Desenvolvimento de *Software*. A partir desse evento, relataram Zafar, Nazir e Abbas (2017) e Nawaz, Aftab e Anwer (2017) que muitos modelos de gestão de projetos de desenvolvimento de *software* se popularizaram, tais como *Scrum*, *Extreming Programming (XP)*, *Feature-Driven Development* e o *Dynamic System Development Method*.

Com o passar nos anos e, mesmo com a adoção das principais propostas ágeis como o *Scrum*, *Lean Information Technology*, *XP*, *Kanban*, Sutherland (2019) e Ries (2011) relataram que o pensamento de gerentes de projeto e da alta gestão sobre a entrega de valor ainda se concentra na entrega de um projeto considerando o tripé de escopo, prazo e custo. Assim, Sutherland (2019) e Ries (2011) descreveram o pensamento sobre o que representa ser o “valor” em projetos ágeis, que está no desenvolvimento de produtos de *software* certos e na hora certa (*just in time*). Os autores ainda afirmaram que se faz necessário determinar hipóteses que precisam ser validadas, obter o *feedback* de clientes e *stakeholders*, aprender e adaptar o produto com o valor esperado no menor tempo possível (Sutherland, 2019; Ries, 2011). Além disso, para Sutherland (2019) e Ries (2011), muitas organizações medem o valor por meio das métricas de vaidade, tais como a pontuação de um aplicativo por usuários que nunca utilizaram e são consideradas no contexto de qualidade do produto. Isto é, tais métricas não se sustentam na validação das hipóteses e na decisão de pivotar ou perseverar sobre a estratégia do projeto de um produto de desenvolvimento de *software* (Sutherland, 2019; Ries, 2011). Para elucidar esse contexto, Pinochet, Pachelli e Da Rocha (2018) definiram que uma métrica de vaidade é considerada uma forma superficial de aumentar a autoestima de uma publicação ou *site*, sem oferecer *insights* significativos para melhorar os resultados reais de um produto, serviço ou mesmo sobre a organização.

Segundo Sutherland (2019) e Ries (2011), o grande desafio das organizações tradicionais está em como mudar o *mindset* tradicional de controle e monitoramento de projetos em times de desenvolvimento de *software* que estejam utilizando de, ao menos, um *framework* ágil. Esse *mindset* de gestão tradicional, conforme Sutherland (2019) e Ries

(2011), quando aplicado aos projetos ágeis traz disfunções nas tomadas de decisão gerando impactos financeiros em projetos de software. Isto é, segundo Sutherland (2019) e Ries (2011), os prazos de entrega informados aos líderes devem ser baseados em dados históricos de entrega do time e realizar os lançamentos do produto mínimo viável (*Minimum Viable Product* – MVP). Conforme Ries (2011) e Caroli (2017), o MVP é uma pequena versão funcional e sem erros críticos de um *software* lançado em produção ou no mercado para obter o *feedback* de clientes com o foco de construir, medir e aprender com os dados. Uma alternativa de mudança de *mindset* organizacional é indicado por Genari e Ferrari (2016) como o *Kaizen* que envolve em seu processo os principais agentes da estrutura organizacional e da organização de projetos, e está relacionado com a evolução incremental de processos organizacionais e de produtos.

Segundo o *Project Management Institute* (PMI, 2017), as métricas tradicionais são orientadas ao planejamento, com destaque ao *Earned Value Management* (EVM). Para o PMI, o EVM realiza a análise do valor agregado comparando o que foi planejado *versus* o que foi realizado para avaliar o desempenho do projeto. Na visão de Jyothi e Rao (2017) e dos Santos *et al.* (2018), na gestão ágil de projetos o contexto muda para o valor entregue ao cliente e as métricas se concentram, em sua maioria, no *delivery* do projeto. Conforme Jyothi e Rao (2017), o *delivery* é uma etapa de construção e entrega de um incremento de *software* e as métricas calculadas apenas na etapa de *delivery* não oferecem a visão como instrumento gerencial que a liderança corporativa necessita para a tomada de decisão na gestão de projetos. Para Toh, Sahibuddin e Mahrin (2019), a prática *DevOps*, junção das palavras “desenvolvimento” (*development*) e “operação” (*operation*), considera as etapas de *discovery*, *delivery* e *release*, e são executadas com objetivo da entrega contínua de *software* de valor. Além disso, *DevOps* através de suas técnicas, práticas arquitetônicas e culturais resultam numa convergência de muitos conceitos e movimentos filosóficos e gerenciais em torno da organização (Kim *et al.*, 2018).

Nesse sentido que Budacu e Pocatilu (2018) reafirmaram o conceito de *delivery* com o *Continuous Delivery as a Service* e *DevOps* como sistemas produtivos que utilizam métricas como o LT e o CT que são úteis quando o que é produzido permanece o mesmo, como um produto ou item específico. Ao considerar o trabalho do conhecimento, Budacu e Pocatilu (2018) afirmaram que a saída desse tipo de trabalho é complexa e pode variar muito e as métricas *Kanban* tendem a ser mais úteis em atividades de projeto, manutenção de *software*, sustentação de sistemas produtivos e suporte técnico de TI.

Sutherland (2019) e Ries (2011) argumentam que as organizações estão migrando para estruturas orientadas a projetos, especialmente na gestão de projetos de desenvolvimento de *software*. Sutherland (2019) e Ries (2011) enfatizaram a necessidade de as organizações adaptarem suas práticas de gestão e métricas de avaliação de seus projetos de acordo com a gestão adotada, nesse caso, gestão tradicional ou ágil de projetos. As métricas em projetos ágeis se concentram no trabalho de *delivery* do projeto, produzindo uma lacuna entre a gestão ágil de projetos, a gestão de projetos tradicionais, as métricas de organizacionais e a gestão de portfólio de projetos (Huijgens *et al.*, 2017; Budacu & Pocatilu, 2018). Como consequência, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as ágeis em gerenciamento de projetos de desenvolvimento de *software* (Huijgens *et al.*, 2017; Budacu & Pocatilu, 2018). Em relação ao prazo do projeto, as práticas ágeis preconizam que o prazo das atividades informados aos líderes devem ser baseados em dados históricos de entrega da equipe de projeto ao invés do longo planejamento realizado após a fase de iniciação do projeto tradicional (Sutherland, 2019; Ries, 2011).

Tal cenário cria uma lacuna entre a gestão ágil e tradicional de projetos, a gestão programas de projetos, as métricas e estruturas organizacionais, e a gestão tradicional de portfólio e orçamentária em projetos (Budacu & Pocatilu, 2018; Huijgens *et al.*, 2017; Jyothi & Rao, 2017). Conforme Azanha Neto, *et al.* (2022), a gestão e liderança em seus diferentes níveis de atuação dentro da organização e nos projetos, possuem a uma percepção confusa entre as práticas tradicionais e ágeis no gerenciamento de projetos de desenvolvimento de *software*. Ainda Azanha Neto, *et al.* (2022) definiram que as práticas ágeis sugerem que os prazos das atividades informados aos líderes se baseiem em dados históricos de entrega, em vez de depender de um planejamento e controle das atividades após a fase de iniciação do projeto.

1.1 PROBLEMA DE PESQUISA

Diante do ambiente organizacional orientado a projetos, Jyothi e Rao (2017) e dos Santos *et al.* (2018) relatam que os projetos ágeis são medidos por métricas como o *Cycle Time* (CT) e o *Lead Time* (LT), ambas com a finalidade de avaliar o desempenho do projeto e do produto gerado para os *stakeholders*. O CT permite medir o tempo em pequenas etapas de desenvolvimento de *software* como o ciclo de programação e testes. O LT permite medir o tempo de desenvolvimento de *software* desde o momento de comprometimento com o início

do desenvolvimento até a efetiva entrega do *software* ao cliente. Entretanto, a forma que se mede o desempenho nos projetos de desenvolvimento de *software* ainda é muito fragmentada em questões operacionais do projeto. Isto é, medir o desempenho apenas durante a concepção e o desenvolvimento do projeto não garante que clientes, executivos da organização e *stakeholders* estejam olhando para a mesma perspectiva quando se trata de métricas de desempenho e monitoramento de projetos e as metas importantes da organização. Como consequência da ausência desse olhar para a mesma perspectiva, a utilização das métricas em propostas de gestão diferentes (tradicional e ágil), podem causar uma percepção difusa entre as práticas tradicionais e as ágeis em gestão de projetos de desenvolvimento de *software*.

Assim, tal forma de medir o desempenho de tais projetos, não é adequada para que a organização e *stakeholders* internos ou externos, tenham uma visão unificada do projeto dentro de um ambiente sob as práticas ágeis de gerenciamento de projetos. Além disso, faltam métricas relacionadas as etapas de *discovery*, *delivery* e *release* que venham a direcionar as estratégias de projeto, de produto e organizacionais para entrega de valor aos clientes e *stakeholders* de maneira mais rápida e confiável, por meio do emprego, por exemplo, das práticas *DevOps*. O *DevOps* enfatiza a automação de tarefas repetitivas, a utilização de práticas de integração contínua e implantação contínua (*Continuous Integration* – CI e *Continuous Delivery* - CD), a monitorização contínua do desempenho do *software* em produção (Dorninger & Ziebermayr, 2021). Por fim, a utilização de ambientes de desenvolvimento e operações com foco na adoção de práticas ágeis e na busca pela melhoria contínua (Gokarna & Singh, 2021), entrega e implementação dos projetos de software e na transformação digital (Süß, Swift & Escott, 2022). Assim, a ausência de tais métricas e/ou a utilização de métricas de projetos tradicionais, dificultam a visão e o trabalho da liderança corporativa para tomar decisões fundamentais em projetos ágeis.

A partir dessa problemática, a proposta de um *framework* de métricas aplicado em todas as etapas do *The Agile Landscape* (TAL) que correspondem as etapas de *Initiate*, *Discovery*, *Delivey* e *Release* visa medir todos os aspectos do projeto desde a concepção de uma ideia ou hipótese até a avaliação e *feedback* do produto de *software* contemplado pelo cliente. Assim, espera-se que esse *framework* de métrica possa suportar com uma visão consolidada de métricas ágeis para apoiar a gestão em diversos níveis de avaliação de hipóteses, escopo, projeto e produto. Além disso, espera-se que esse trabalho de tese possa contribuir com a evolução de estudos acadêmicos no desenvolvimento da agilidade na gestão

de projetos nas diferentes organizações, na agilidade de negócios (*business agility*) e na medição em diferentes etapas do projeto ágil.

Assim, as análises tradicionais baseadas no triângulo de ferro ou nas técnicas de valor agregado com o EVM (PMI, 2017) utilizadas para apoio aos líderes podem ser consideradas inadequadas se aplicadas fora do contexto de gerenciamento ágil de projetos de desenvolvimento de *software* (Budacu & Pocatilu, 2018), podendo causar impactos no controle e gerenciamento de tais projetos, como Misra e Omorodion (2011) e Oza e Korkala (2012) afirmaram.

Dessa forma, para responder estes problemas, propõe-se a utilização de um *framework* para uso como instrumento gerencial de apoio ao controle e monitoramento de projetos ágeis. Como resultado, espera-se que o *framework* proposto apresente métricas técnicas, de fluxo, de ambiente e de negócio que são capazes de apoiar na gestão de desempenho de projeto e produto nos níveis de portfólio, programa e equipes de projeto para que líderes, *stakeholders* e clientes possam compartilhar resultados na evolução do projeto de desenvolvimento de *software*.

Com base na contextualização e problemática apresentada sobre como as empresas possam repensar sua estrutura organizacional para serem ágeis e, ao mesmo tempo, em saber conduzir projetos de forma ágil, é proposta a seguinte questão de pesquisa: **Como as métricas ágeis podem orientar a liderança corporativa na gestão de projetos ágeis?**

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

Propor um *framework* de métricas ágeis de maneira a orientar a liderança na gestão de projetos ágeis.

1.2.2 OBJETIVOS ESPECÍFICOS

- i. Identificar qual é o papel da liderança corporativa em relação aos projetos ágeis;
- ii. Identificar às métricas de gerenciamento em projetos tradicionais e ágeis;

- iii. Identificar o estado da base de patentes sobre propostas de *framework* de métricas em projetos ágeis;
- iv. Apresentar um *framework* de métricas em projetos ágeis de maneira a orientar a liderança na gestão de projetos ágeis.

1.3 JUSTIFICATIVA

Uma organização fiel ao conceito das práticas ágeis, conforme Kristensen e Shafiee (2019), requer diferentes formas de gestão do conhecimento, liderança, comunicação, tomada de decisão, governança, organização de equipes, alocação de papéis, alinhamento e mentalidade organizacional. Assim, Kristensen e Shafiee (2019) afirmaram que as organizações entendem que a transformação ágil altamente disruptiva e uma remodelação ou um ajuste natural ao seu *design* atual é necessário para uma vantagem competitiva significativa.

A justificativa desse trabalho, conforme Kristensen e Shafiee (2019) destacaram, está na necessidade das organizações de alcançarem a agilidade em seus projetos para reduzirem o tempo de lançamento de seus produtos no mercado, para promover a inovação e para lidar com a complexidade de seus projetos e no mercado onde atuam. Outra questão que justifica esse trabalho é que, conforme Kristensen e Shafiee (2019), a implementação de uma técnica ou *framework* ágil não é suficiente para a organização se tornar verdadeiramente ágil, pois as organizações simplesmente ignoraram as mudanças necessárias na estrutura organizacional, em processos, governança e na liderança.

Diante das fases do desenvolvimento ágil de software que envolvem as etapas de *Initiate*, *Discovery*, *Delivery* e *Release* conforme *The Agile Landscape – Deloitte* (TAL) (Paragano, 2021), as métricas em projetos ágeis podem mensurar as etapas do TAL para dar visibilidade sobre o estado real do desempenho das hipóteses, dos resultados obtidos e do *feedback* do cliente. Nesse contexto, a criação de um *framework* de métricas poderá auxiliar as lideranças presentes em uma estrutura de gestão de projetos no processo de remodelagem estrutural da organização visando a agilidade, quanto na gestão ágil de projetos a fim de lançar produtos rapidamente ao mercado e inovar.

A liderança corporativa não possui a visão de todas as métricas necessárias para ter a visibilidade real dos resultados dos projetos ágeis. Isto é, os resultados das iterações são incrementais aos resultados gerais de escopo, prazo e custo esperados pela liderança. Assim,

conforme Jyothi e Rao (2017) e dos Santos *et al.* (2018), as métricas utilizadas atualmente em projetos ágeis oferecem uma visão em cada etapa do projeto, tais como as métricas de *Cycle Time*, *Lead Time* e *Work In Progress* (WIP).

Diante dos cenários apresentados acima, a realização desta pesquisa se justifica por mapear as métricas necessárias para compor um *framework* que ofereça uma visão sobre todas as etapas do gerenciamento de projetos ágeis, dos resultados apresentados e do impacto de valor desses resultados sobre os *stakeholders* e clientes. Assim, espera-se que esse *framework* de métrica proposto por esse trabalho possa suportar com uma visão consolidada de métricas ágeis para apoiar a gestão em diversos níveis de avaliação de hipóteses, escopo, projeto e produto nas etapas do TAL. Além disso, espera-se que essa tese possa contribuir na evolução de estudos acadêmicos no desenvolvimento da agilidade na gestão de projetos nas organizações.

Por fim, espera-se que o *framework* de métricas ofereça aos praticantes na gestão de projetos ágeis uma ferramenta de caráter gerencial de métricas que possa auxiliar na gestão de desempenho e *feedback* de projeto e produto nos níveis de portfólio, programa e times de projeto. Isto é, poderá contribuir academicamente com o desenvolvimento de novas métricas de projetos ágeis e um avanço na forma como a liderança faz a gestão de projetos nos níveis de programa e portfólio.

1.4 ESTRUTURA DA TESE

Para a estrutura da tese, a Tabela 1 descreve a Matriz Metodológica (MM) proposta por Costa, Ramos e Pedron (2019) e tem como objetivo demonstrar como a tese está estruturada com base nos estudos que a compõem. A MM está estruturada com a questão de pesquisa que rege todo o estudo e a sua questão de pesquisa. Além disso, é demonstrada a Justificativa de distinção dos estudos com o título, questão de pesquisa e objetivo geral de cada estudo. Em complemento, cada um dos estudos apresenta a justificativa de interdependência entre os estudos da tese, com as seguintes informações: Tipo de cada estudo, método de pesquisa adotado, procedimentos de coleta de dados, procedimentos de análise de dados, *status* de publicação e o produto tecnológico proposto.

Dessa maneira, os estudos que foram (até a conclusão da versão final da tese) publicados em revistas científicas serão exibidos na coluna Estudos. Assim, os Estudos 1, 2 e 3 foram publicados em revistas científicas e o Estudo 4 está em processo de submissão.

Tabela 1- Matriz Metodologia da Tese

Questão central da tese:	Como as métricas ágeis podem orientar a liderança corporativa na gestão de projetos ágeis?
Objetivo geral da tese:	Propor um <i>framework</i> de métricas ágeis de maneira a orientar a liderança na gestão de projetos ágeis.
Justificativa geral da tese:	As organizações e a liderança que realizam o processo de transição dos modelos tradicionais para serem mais ágeis e as organizações que nasceram ágeis, devem compreender que a implementação de um processo, técnicas, ferramentas e <i>frameworks</i> ágeis não são suficientes para se tornar verdadeiramente ágil, pois as organizações simplesmente ignoraram as mudanças necessárias na estrutura organizacional, tais como em seus processos, governança, na cultura e na gestão por meio de sua liderança (Estudo 1). Para apoiar a liderança na gestão, as métricas de projetos podem auxiliar nesse processo de gerenciamento ágil de projetos (Estudo 2). Diante desse cenário, as métricas em projetos ágeis se concentram no trabalho de <i>delivery</i> do projeto, produzindo uma lacuna entre a gestão ágil de projetos, a gestão de projetos tradicionais, as métricas de organizacionais e a gestão de custos e portfólio de projetos. Como consequência, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as ágeis em gerenciamento de projetos de desenvolvimento de <i>software</i> . A realização desta pesquisa se justifica por abordar as métricas de desempenho de projetos ágeis, contribuindo com a teoria e a prática por consolidar as métricas para uso da liderança durante

todo o ciclo de iterações. Ademais, os resultados desta pesquisa permitirão sugerir um <i>framework</i> de métricas ágeis de maneira a orientar a liderança na gestão de projetos ágeis.							
Justificativa de distinção dos estudos				Justificativa de interdependência dos estudos			
O primeiro estudo teve como objetivo estudos sobre as práticas de gerenciamento de projetos abordam a liderança de projetos ágeis. O segundo estudo teve o objetivo de mapear as métricas de desempenho de projetos ágeis na literatura acadêmica. O terceiro estudo teve como objetivo de realizar uma pesquisa em base de dados de patentes para identificar possíveis patentes sobre o desenvolvimento de um processo orientado a métricas ágeis e ao modelo TAL. O quarto estudo objetiva a validação através do <i>focus group</i> com praticantes em gestão ágil de projetos de um <i>framework</i> de métricas ágeis de maneira a orientar a liderança na gestão de projetos ágeis.				O primeiro estudo identificou a importância da liderança na gestão de projetos ágeis, que motiva a realização do segundo estudo. O segundo estudo realizou o mapeamento de métricas para desempenho de projetos ágeis para identificar as métricas segundo o modelo TAL. Para garantir o caráter de ineditismo do <i>framework</i> a ser proposto, o terceiro estudo realizou um levantamento patentário para identificar as possíveis patentes sobre o desenvolvimento de um processo orientado a métricas ágeis e ao modelo TAL. O quarto estudo permite validar o <i>framework</i> de métricas construído com base na avaliação de praticantes em gestão de projetos ágeis em diferentes organizações. Ao final da realização dos estudos será possível o depósito de uma patente de invenção.			
Estudos	Título	Questão de Pesquisa	Objetivo Geral	Tipo de Estudo	Método de pesquisa	Procedimentos de coleta de dados	Procedimentos de análise de dados
Estudo 1 (Publicado na <i>Research, Society and Development</i>, v. 11, n. 5, disponível no Apêndice e em https://rsdjournal.org/index.php/rsd/article/view/28117)	A importância da liderança em projetos ágeis: Revisão Sistemática da Literatura	Como os estudos sobre as práticas ágeis de gestão de projetos abordam a liderança?	Identificar qual é o papel da liderança corporativa em relação aos projetos ágeis.	Teórico.	Revisão Sistemática da Literatura	Coleta realizada na bases de dados Scopus e Web of Science;String de pesquisa: (("fora*" or "sprint" or "scrum" or "kanban") AND ("leader" or "leadership")). Não foi aplicado nenhum filtro com o objetivo de alcançar o maior número publicações no tema. Aplicado critérios de inclusão e exclusão.	Leitura crítica e reflexiva, com a classificação do conteúdo de acordo com a aderência aos temas da pesquisa. Categorização e construção de matriz de amarração para apresentação dos resultados, de acordo com o protocolo de Pollock e Berge (2018). Pollock, A., e Berge, E. (2018). How to do a systematic review. <i>International Journal of Stroke</i> , 13(2), 138-156.

							https://doi.org/10.1177/1747493017743796
Estudo 2 (Publicado na Journal on Innovation and Sustainability, v. 14, n. 4 (2023), disponível no Apêndice e em https://revistas.pucsp.br/index.php/risus/article/view/63363/44256)	Mapeamento de métricas para desempenho de projetos ágeis: Revisão Sistemática da Literatura	Quais são as métricas utilizadas para medir o desempenho de projetos nas práticas tradicionais e ágeis?	Identificar às métricas de gerenciamento em projetos tradicionais e ágeis	Teórico.	Revisão Sistemática da Literatura	Coleta realizada nas bases de dados Scopus e Web of Science; String de pesquisa: ("metric*" and ("project managem*") or "metric*" and ("agil*" or "SCRUM" or "kanban" or "lean*" or "Devops" or "NPS" or "Net promoter score" or "EBM" or "Evidence Based Management" or "OKR" or "Objective Key Result*")). Não foi aplicado filtro com o objetivo de alcançar o maior número publicações. Aplicado critérios de inclusão e exclusão.	Leitura crítica e reflexiva, com a classificação do conteúdo de acordo com a aderência aos temas da pesquisa. Categorização e construção de matriz de amarração para apresentação dos resultados, de acordo com o protocolo de Pollock e Berge (2018). Análise de conteúdo buscando pontos convergentes e divergentes da base montada nos grupos focais. Pollock, A., e Berge, E. (2018). How to do a systematic review. <i>International Journal of Stroke</i> , 13(2), 138-156. https://doi.org/10.1177/1747493017743796
Estudo 3 (Publicado na South American Development Society Journal, v. 9 n. 27 (2023), disponível no	Estudo de Patentes sobre Métricas para a Gestão Ágil de Projetos.	Qual é o estado da base de patentes está em relação aos modelos propostos para métricas ágeis?	Identificar o estado da base de patentes sobre propostas de framework de métricas em projetos ágeis.	Teórico.	Revisão Sistemática da Literatura	Procedimento de Identificação - String de Pesquisa nas bases ("agile metric*" and ("software development" or "business agility"))	Abordagem qualitativa e exploratória proposto por Creswell, J. W. (2014). <i>Qualitative, quantitative and mixed methods approaches</i> . Sage. e mapeamento de

<p>Apêndice e em https://sadsj.org/index.php/revista/article/view/663/521)</p>							<p>patentes conforme sugerido por Paranhos, R. D. C. S., & Ribeiro, N. M. (2018). Importância da prospecção tecnológica em base em patentes e seus objetivos da busca. Cadernos de Prospecção, 11(5), 1274.</p>
<p>Estudo 4</p>	<p>FMA: Framework de Métricas Ágeis como Ferramenta de Apoio A Gestão Ágil de Projetos.</p>	<p>Como as métricas de desempenho podem contribuir com a liderança no gerenciamento de projetos ágeis?</p>	<p>Apresentar um framework de métricas em projetos ágeis de maneira a orientar a liderança na gestão de projetos ágeis.</p>	<p>Empírico.</p>	<p>Qualitativo</p>	<p>Entrevistas e Grupos Focais.</p>	<p>Análise de conteúdo.</p>
	<p>Nome e tipo de produto</p>	<p>Descrição</p>	<p>Aderência</p>	<p>Impacto</p>	<p>Aplicabilidade</p>	<p>Inovação</p>	<p>Complexidade</p>
<p>Produto Tecnológico</p>	<p>Patente de Invenção.</p>	<p><i>Framework</i> de métricas de desempenho para apoio à liderança no gerenciamento de projetos ágeis.</p>	<p>Alta - O produto tecnológico desta tese está altamente alinhado com a linha de pesquisa 'Gerenciamento de Projetos' do PPGP UNINOVE e projetos de 'Projetos Ágeis e Híbridos', o que confirma sua alta aderência e relevância</p>	<p>Alta - O produto desta tese, útil para todo o setor de TI, apoia a a gestão de projetos ágeis, promovendo mudanças e elevando seus padrões de controle e</p>	<p>Média - Esta solução pode ser aplicada em qualquer contexto de negócio, setor ou complexidade do projeto em ambientes ágeis.</p>	<p>Média - Inovação incremental pela combinação de conhecimentos prévios e adaptativo devido às mudanças de contexto.</p>	<p>Alta. complexidade, ligada à variedade de participantes e conhecimentos, é alta para o produto desta tese. Ele é o único que abrange todas as etapas do Agile Landscape – Deloitte (TAL), desde a concepção até avaliação do <i>software</i>.</p>

			para as ciências sociais aplicadas	monitoramento por meio da adoção de métricas ágeis.			
--	--	--	---------------------------------------	--	--	--	--

Fonte: elaborado pelos autores, 2022.

Nos próximos três capítulos (dois, três e quatro) serão apresentados os artigos que formam, estruturam e norteiam e sustentam esta pesquisa de tese.

2 PROCEDIMENTOS METODOLÓGICOS

Para este projeto de tese, por ser uma peça única, se faz necessário apresentar a metodologia geral, bem como os procedimentos metodológicos detalhados dos estudos e como eles se conectam.

2.1 DESENHO DA PESQUISA

A estrutura desta pesquisa de tese é orientada pela estratégia de publicação de artigos onde serão considerados quatro estudos dentro de um processo de pesquisa que será detalhado na Figura 1 (Desenho da pesquisa da tese por estudos) a seguir. Além disso, o tópico a seguir irá apresentar a matriz metodológica da tese (Tabela 1) que possui detalhes sobre os objetivos e justificativas da pesquisa, além do detalhamento estruturado dos quatro estudos previstos nesse projeto de pesquisa.

Considerando a relação entre os estudos verifica-se que os achados de ambos os estudos se complementam, como pode ser visto na Figura 1. Isto é asseverado ao identificar que os estudos consideram em seus achados aspectos importantes para se atingir os resultados esperados por essa pesquisa.

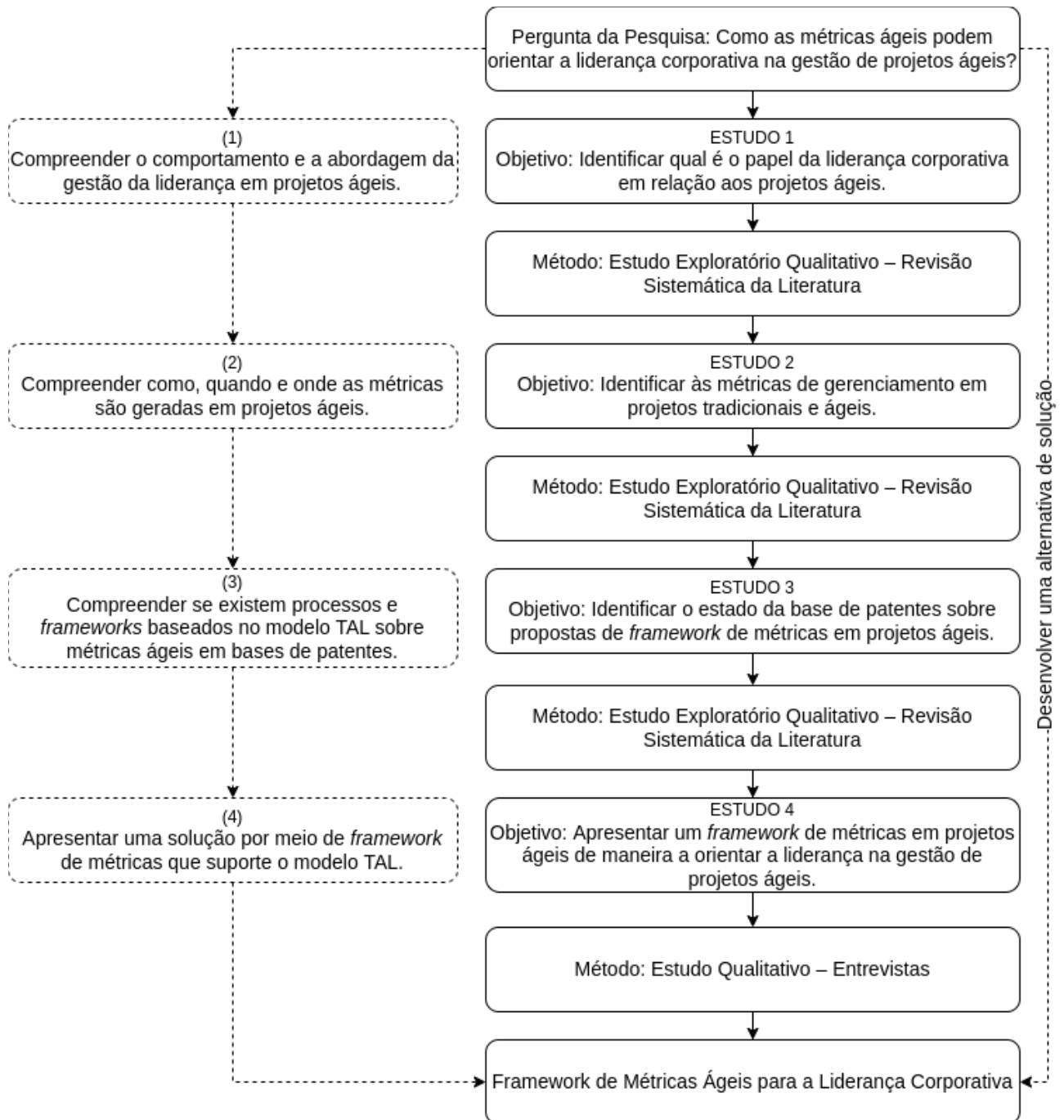


Figura 1 - Desenho da pesquisa da tese por estudos

Fonte: elaborado pelo autor

O Estudo 1 concentra-se em analisar como a liderança é abordada na gestão ágil de projetos. Por meio dessa análise, foi possível compreender como os líderes de projeto implementam e mantêm práticas ágeis, bem como lidam com a dinâmica específica da liderança servidora e os desafios desses projetos ágeis.

Após estabelecer uma compreensão das práticas de liderança ágil, iniciou-se o Estudo 2, que compara as métricas utilizadas para avaliar o sucesso em projetos tradicionais e em projetos ágeis. Isto nos permite avaliar como os indicadores de desempenho diferem entre as abordagens e que tipos de métricas são mais eficazes para avaliar a implementação ágil.

Usando as informações coletadas no Estudo 2, passa-se para o Estudo 3, onde foi realizado um estudo para identificar publicações e patentes sobre métricas e possíveis *frameworks* usados na gestão ágil de projetos. Este passo nos permite estabelecer um corpo de conhecimento sobre o assunto e identificar lacunas na literatura existente que nossas pesquisas podem preencher. Nesse estágio é apresentada a versão inicial do *framework* que é objetivo desta pesquisa de tese de doutorado.

Finalmente, no Estudo 4, foram realizadas entrevistas utilizando a abordagem de grupos focais com gestores de projetos praticantes da abordagem ágil para obter suas opiniões e experiências relativas às métricas ágeis em todas as fases da gestão ágil. Após essa etapa, vale ressaltar que o *framework* proposto deu origem ao produto tecnológico desta pesquisa. O produto tecnológico proposto teve como objetivo o desenvolvimento de um *framework* para orientar a liderança na gestão de projetos ágeis, considerando que as métricas necessárias para uma visão completa em diversos níveis da liderança e das etapas na gestão de projetos ágeis. O produto tecnológico foi submetido aos órgãos oficiais para realizar o depósito de patente e desenvolvimento de científico e tecnológico, beneficiando as necessidades de mercado e da academia por meio de inovação tecnológica.

A seguir é apresentada a matriz metodológica da tese, com a síntese dos estudos e respectivo encadeamento até a proposição do produto tecnológico.

3 ESTUDO 1: A IMPORTÂNCIA DA LIDERANÇA EM PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA¹

RESUMO

Em um contexto competitivo onde as organizações precisam entregar produtos ou serviços por meio da gestão ágil de projetos, as métricas tornam-se insumos importantes para o controle e monitoramento de projetos. A grande questão é que existem organizações que, apesar de conduzirem projetos ágeis, podem controlar os projetos por meio do uso de métricas tradicionais. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis. Como estratégia metodológica foi adotado uma Revisão Sistemática de Literatura para auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento. Para a coleta de dados foram utilizadas as bases *Web of Science* e *Scopus*. Foram encontrados 83 artigos e os resultados apontaram que os projetos tradicionais seguem controlando os projetos com métricas tradicionais, tal como *Earned Value Management* (EVM). Em projetos ágeis, as métricas mais comuns são métricas associadas *backlog* do produto, *delivery* e qualidade do produto. Este estudo contribui com outras pesquisas que desejam identificar as métricas utilizadas entre as diferentes abordagens de gestão de projetos e orientar novas pesquisas para trabalhos futuros.

Palavras-chave: Desenvolvimento Software. Métricas Ágeis. Métricas Tradicionais. Desempenho de Projeto.

¹ Estudo publicado na *Research, Society and Development*, v. 11, n. 5, disponível no Apêndice e em <https://rsdjournal.org/index.php/rsd/article/view/28117>.

1. INTRODUÇÃO

O modelo em Cascata foi criado com o objetivo de representar a execução de fases de um projeto de desenvolvimento de *software* (Royce, 1970). Nesse modelo, cada fase possui dependência com a fase anterior, onde a próxima fase somente poderá ser iniciada quando a fase atual for concluída (Sommerville, 2007). O modelo em Cascata é composto pelas fases de Levantamento de Requisitos, Análise de Requisitos, Projeto, Implementação ou Desenvolvimento, Testes e Implantação e Manutenção (Pressman, 2021). Dentre elas, as fases de Levantamento de Requisitos e Análise de Requisitos e Projeto podem exigir grandes esforços durante o planejamento e concepção sobre o escopo que será desenvolvido no projeto (Junior e Junior, 2019). Em relação à execução das atividades um dos desafios é alinhar a entrega final do projeto ao desejado pelo cliente (Van Casteren, 2017).

No contexto das práticas gerenciais, os projetos de desenvolvimento de *software* em Cascata sob as práticas tradicionais de gerenciamento de projetos possuem diversas incertezas no início do projeto que podem mudar o escopo inicial e o direcionamento do produto durante o ciclo de vida do projeto (Sutherland, 2019). Para minimizar tais incertezas, Sutherland (2019) atualizou o planejamento em projetos de desenvolvimento de *software* com a adoção do modelo de *Big Design Up Front* (BDUF), adicionando a fase inicial de planejamento. O BDUF é caracterizado pode contemplar todas as atividades previstas que o projeto deve entregar ao longo do seu ciclo de vida (Sutherland, 2019).

Em se tratando das práticas gerenciais e o desempenho de projetos, o uso de métricas tornou-se um fator estratégico para que o controle e o monitoramento do projeto se tornem eficazes (Glenwright, 2007). No tocante às práticas tradicionais, as métricas mais utilizadas pelas organizações é o gerenciamento do valor agregado (Junior e Junior, 2019). O gerenciamento do valor agregado - *Earned Value Management* (EVM) foi proposto pelo *Project Management Institute* (PMI, 2021). O EVM é considerado uma técnica de gerenciamento de projetos para medir o desempenho e o progresso de projetos, com a capacidade de combinar medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido (Glenwright, 2007; PMI, 2021).

Porém, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as práticas ágeis em gerenciamento de projetos de *softwares*. Em relação ao prazo do projeto, as práticas ágeis preconizam que os prazos das atividades informados aos líderes devem ser baseados em dados históricos de entrega da equipe de projeto, ao invés do

planejado (Sutherland, 2019). No caso das práticas em projetos tradicionais de desenvolvimento de *software* ou como visto no modelo BDUF, os prazos são baseados as estimativas definidas na fase de planejamento do projeto e dando origem ao cronograma de atividades (Sutherland, 2019), refletindo em uma distorção na utilização das métricas em ambas as práticas.

No contexto das práticas ágeis, as métricas não são orientadas ao planejamento, mas a entrega de valor entregue ao cliente (Gren, Goldman & Jacobsson, 2020). De acordo com Jyothi e Rao (2017) e Budacu e Pocatilu (2018), as métricas ágeis mais utilizadas são (i) *velocity*; (ii) *Work Item Age*; (iii) *Throughput* e (iv) *Lead Time*. O *velocity* é uma métrica para avaliação utilizada no desempenho dos desenvolvedores por meio dos pontos de histórias do usuário (escopo). O *Work Item Age* é uma métrica para medir o tempo em dias que um requisito está em uma determinada fase do desenvolvimento. Já o *Throughput* representa uma métrica para avaliação da capacidade de entrega da equipe de projeto ao final de cada iteração enquanto o *Lead Time* é considerado uma medida que avalia o tempo em que um requisito é puxado no *Kanban* para início do desenvolvimento até ser entregue ao cliente (Budacu & Pocatilu, 2018).

Já as práticas tradicionais utilizam o EVM para medir o desempenho e o progresso de projetos, utilizando métricas baseadas em medidas comparativas entre o esforço planejado em relação ao esforço realizado (PMI, 2021). Glenwright (2007) apresenta as métricas mais utilizadas pelas organizações sob as práticas tradicionais de gerenciamento de projetos são (i) índice de desempenho de cronograma e (ii) índice de desempenho de custo. O índice de desempenho do cronograma (*Schedule Performance Index – SPI*) analisa o progresso real ou o valor agregado do trabalho realizado com o progresso do trabalho planejado até o momento da medição. Já o índice de desempenho de custo (*Cost Performance Index – CPI*) é uma medida da eficiência de custos realizados com o progresso do custo planejado até o momento da medição.

As métricas possuem relações distintas de acompanhamento durante o ciclo de vida dos projetos. Em relação às entregas dos projetos, nas práticas tradicionais, o produto gerado é entregue somente no final do projeto ou entregue em fases dentro de uma estratégia de cronograma em fases (PMI, 2021). Conseqüentemente, o produto é considerado como entregue aos *stakeholders* somente no final do projeto. Já nas práticas ágeis o produto é entregue em até quatro semanas (*sprint*) em forma de incrementos funcionais. A soma dos

incrementos funcionais pode viabilizar os lançamentos do *Minimum Viable Product* (MVP) (Ries, 2011). O MVP é caracterizado por ser uma versão funcional e sem erros críticos de um *software* para possível implementação em ambiente produtivo ou no mercado, com o objetivo de receber o *feedback* de clientes mais rápido possível (Sutherland, 2019).

Ao se observar a utilização das métricas para análise do desempenho de projetos de desenvolvimento de *software*, é possível notar divergências entre as práticas ágeis e tradicionais de gerenciamento de projetos. Essa divergência consiste em um cenário em que organizações que conduzem projetos sob as práticas tradicionais decidem implementar as práticas ágeis de gerenciamento de projetos. Porém, a liderança permanece utilizando métricas tradicionais, isto é, medidas baseadas nas abordagens de custo, prazo e escopo diferentes entre as práticas, podendo causar impactos no monitoramento e desempenho dos projetos ágeis.

O problema que tange e justifica nessa pesquisa é que muitas organizações que conduzem projetos, implementaram o modelo ágil de desenvolvimento de produtos em equipes de projeto isoladas e com as métricas ágeis comumente na operação. Além disso, a liderança realiza a medição do desempenho do projeto por meio de métricas tradicionais, o que podem causar impactos no gerenciamento de projetos ágeis, dado que a gestão tradicional e a ágil de projetos operam com abordagens de custo, prazo e escopo de maneiras diferentes.

Assim e com base no apresentado sobre a divergência da utilização de métricas para o controle e monitoramento de projetos tradicionais e ágeis, abre-se uma lacuna para a realização desta pesquisa, procurando responder a seguinte questão: **“Quais são as métricas utilizadas para medir o desempenho de projetos nas práticas ágeis?”**. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis. Para responder tal indagação, foi adotado como estratégia metodológica uma Revisão Sistemática de Literatura (RSL) verificando as métricas tradicionais e ágeis utilizadas para análise de desempenho de projetos. A justificativa para tal escolha se dá pelo papel da RSL auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento (Tranfield, Denyer & Smart, 2003).

Na seção seguinte serão apresentados os procedimentos metodológicos. Posteriormente são apresentados os resultados e, por fim, serão apresentadas as conclusões, discutidas as limitações desta pesquisa e indicações para estudos futuros.

2. MATERIAIS E MÉTODOS

Esta pesquisa adotou uma RSL como método a fim de compreender a convergência de duas temáticas relevantes no contexto de gestão de projetos, as métricas, tanto nas práticas tradicionais como nas práticas ágeis. A RSL se difere das tradicionais revisões narrativas por adotar um processo científico sistemático que é replicável e transparente (Tranfield, Denyer & Smart, 2003). Nesse sentido, justifica-se também a sua utilização por uma RSL minimizar o viés na construção de um *corpus* teórico, bem como a possibilidade de construir uma trilha de auditoria das decisões e procedimentos aplicados (Cook, Mulrow & Haynes, 1997).

Os procedimentos para realização desta RSL seguiram seis fases prescritas por Pollock e Berge (2018): (i) esclarecer metas e objetivos de pesquisa; (ii) buscar pesquisas relevantes; (iii) coletar dados; (iv) avaliar a qualidade dos estudos; (v) sintetizar as evidências; (vi) interpretar os achados. O objetivo das fases e atividades apresentadas é garantir o rigor e a robustez que se objetivam neste tipo de pesquisa.

A primeira etapa foi motivada pela questão que norteia esta pesquisa, que foi “quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis?” Para tanto, foram utilizadas como fontes de pesquisa as bases de dados acadêmicas *Web of Science* e *Scopus*, por serem umas das principais bases de dados para acesso às pesquisas publicadas na área de ciências sociais.

A *string* utilizada para realização das buscas foi ("*metric**" and ("*project managem**") or "*metric**" and ("*agil**" or "*SCRUM*" or "*kanban*" or "*lean**" or "*Devops*" or "*NPS*" or "*Net promoter score*" or "*EBM*" or "*Evidence Based Management*" or "*OKR*" or "*Objective Key Result**"). A pesquisa foi realizada em 01 de maio de 2022. A utilização dos operadores booleanos “and” e “or”, além do uso do símbolo “*” permitem uma maior abrangência e controle na construção da base de pesquisa. Os operadores são aplicados levando em consideração a intersecção das duas áreas de pesquisa estudadas. O uso do asterisco incorpora todas as variações da palavra na posição posterior a que ela se encontra. Destaca-se que não foi aplicado filtro temporal para que fosse possível mapear toda a produção sobre as temáticas estudadas.

Portanto, após a primeira fase que compreende o estabelecimento dos objetivos, os pesquisadores passaram para a aplicação da *string* de busca. Os resultados encontrados na primeira rodada passaram pela análise e triagem, apresentado na Figura 1, respeitando o

proposto por Pollock e Berge (2018), que apresentam quatro etapas para a elaboração do *corpus* de análise.

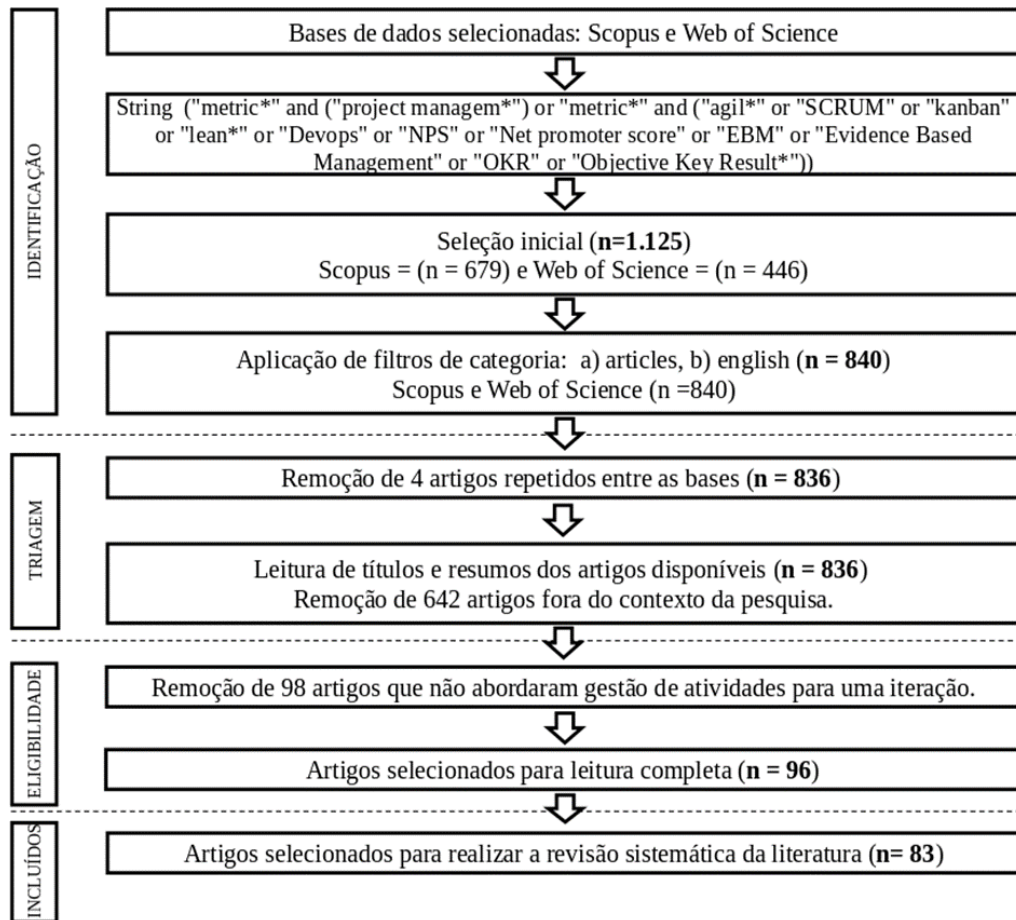


Figura 1 - Etapas da RSL

Fonte: adaptado de Pollock e Berge (2018).

A Figura 1 apresenta as etapas para a elaboração do *corpus* de análise. A primeira etapa destaca a *string* utilizada para a pesquisa nas bases de dados. A segunda etapa é a triagem dos resultados com a finalidade de balizar os artigos correspondentes à proposta de pesquisa. Já na terceira etapa foram aplicados os critérios de elegibilidade, onde os artigos foram avaliados segundo os critérios de inclusão e exclusão. A quarta etapa constituiu o *corpus* de análise, resultando na base de dados amostral de artigos para a pesquisa.

Para a composição da base de dados, alguns filtros foram aplicados durante a pesquisa inicial, como o critério de somente artigos em periódicos, excluindo da base os artigos em congresso, livros, entre outros. Essa exclusão foi definida, devido esses estudos não passarem por uma revisão por pares em alguns casos, ou mesmo pela maturidade apresentada em *working papers*, além de mitigar a redundância de obras apresentadas em congressos e

publicadas em periódicos. As áreas delimitadas para esta pesquisa foram: “*Management*” e “*Business*” para *Web of Science*, “*Business*” para a base *Scopus*.

Com a consolidação das bases de dados, foram removidos os artigos repetidos com o objetivo de eliminar a redundância. Na sequência, a base de dados foi tratada com auxílio de planilhas eletrônicas *Excel*. O *Excel* permitiu realizar as análises dos dados e apresentar os resultados a partir da combinação de informações quantitativas pela análise de frequência, bem como informações qualitativas pela categorização dos conteúdos dos artigos. Esta fase da pesquisa permitiu também apresentar uma análise descritiva relevante sobre o estudo realizado.

A partir da leitura dos títulos e resumos na fase de triagem para verificar a elegibilidade dos artigos foram estabelecidos alguns critérios de inclusão e exclusão, destacados na Tabela 1. Após verificação e leitura dos resumos e introduções dos 840 artigos selecionados previamente, a base final foi composta por 83 artigos aceitos que formaram o *corpus* de pesquisa.

Tabela 1 - Critérios de Inclusão e Exclusão

Critérios de Inclusão	Razão para Inclusão
Artigos que conceituam os construtos estudados	Permitir abordar finalidades da pesquisa: compreender os construtos estudados conforme as obras.
Artigos que abordam as relações entre os construtos estudados	Permitir contemplar alinhamento das obras: compreender as interdependências e relações entre os construtos.
Artigos publicados	Oferecer maior rigor nos argumentos e contribuições teóricas estudadas.
Critérios de Exclusão	Razão para Exclusão
Artigos com foco em Qualidade, <i>Marketing</i> , Saúde, Finanças ou outros fins fora dos construtos determinados.	Excluir artigos que não estejam focados nas questões que oferecerão <i>insights</i> para atender os objetivos da pesquisa.
Artigos sem fundamentação teórica relevante ou de baixa relação com os construtos.	Uma das finalidades do estudo é obter perspectivas futuras de pesquisa, por meio de conhecimento teórico existente em uma estrutura, para a qual os pressupostos teóricos são pré-requisitos.

Fonte: Elaborado pelos autores, 2021.

O próximo passo foi realizar a leitura dos 83 artigos contidos no *corpus* de análise, categorizando os conteúdos em planilhas *Excel* com o objetivo de apresentar um agrupamento dos achados e comparação das categorias. As atividades aplicadas nessa fase estão alinhadas com as prescrições de Pollock e Berge (2018) nas fases (v) sintetizar as evidências e (vi) interpretar os achados. Embora tenham sido aplicados alguns tratamentos quantitativos. Nessa pesquisa, priorizou-se a análise qualitativa dos artigos a fim de constituir uma matriz que pudesse representar os achados deste estudo.

3. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Nessa seção são apresentados os resultados após a realização dos processos de análise dos artigos. Na sequência, são apresentadas as categorias evidenciadas após a análise aprofundada dos conteúdos dos artigos.

1. Evolução da publicação dos artigos

Os artigos selecionados nas bases de dados *Web of Science e Scopus* foram triados e o *corpus* de análise foi constituído por 83 artigos. Os artigos passaram por uma análise mais aprofundada, o que permitiu compreender os estudos sobre a relação entre as métricas ágeis e as métricas tradicionais. Os 83 artigos analisados estão situados temporalmente entre os anos de 2011 até maio de 2021, apresentados na Figura 2.

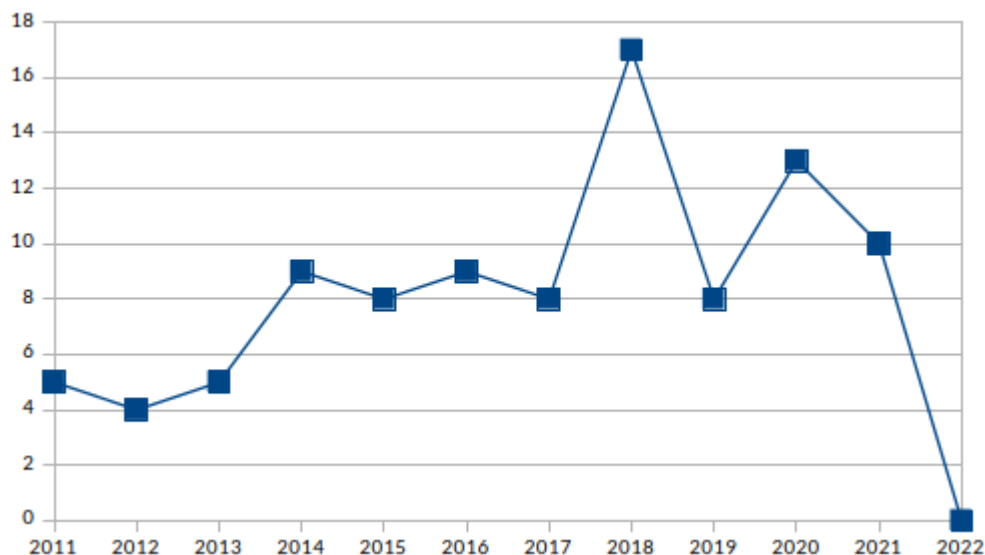


Figura 2 - Análise temporal dos artigos

Fonte: dados da pesquisa, 2022.

Como pode ser observado na análise da Figura 2, de acordo com esta base de dados, a publicação de artigos que tratam da relação entre as métricas ágeis e métricas tradicionais ficou entre 4 e 5 artigos de 2011 até 2013. Entre os anos de 2014 até 2017 que as publicações subiram para 8 e 9 artigos publicados. Somente no ano de 2018 que 17 artigos foram publicados e retornando para 8 artigos em 2019, 13 em 2020 e 10 em 2021. Um aspecto relevante é que no ano de 2018 teve o número maior de publicações e correspondeu com 18% de publicações nos últimos 10 anos ou de 2011 a 2021. Após a análise da distribuição

temporal dos artigos também se buscou compreender a frequência de artigos por periódico. A Tabela 2 demonstra os primeiros 10 periódicos e a respectiva quantidade de artigos publicados no tema, ordenados pelo maior número de publicações e pela ordem alfabética do nome dos periódicos.

Tabela 2 - Periódicos e número de publicações no tema

#	Periódicos	Publicações
1	IEEE Access	15
2	Journal Of Systems And Software	14
3	International Journal Of Project Management	10
4	International Journal Of Lean Six Sigma	10
5	Computer Networks	7
6	Computers & Industrial Engineering	7
7	Journal Of Modelling In Management	6
8	Journal Of Software-Evolution And Process	6
9	Journal Of Manufacturing Technology Management	6
10	International Journal Of Software Engineering And Knowledge Engineering	6

Fonte: dados da pesquisa, 2022.

Com relação ao número de artigos publicados por periódico, os periódicos considerados mais profícuos, de acordo com dados da pesquisa, são: ‘IEEE Access’ com 15 artigos; ‘*Journal Of Systems And Software*’ com 14 artigos e ‘*International Journal Of Project Management*’ com 10 artigos. Nota-se que a grande maioria dos periódicos configura com apenas 1 e 2 artigos publicados, sumarizando um total de 60 periódicos nesse contexto.

Apesar da disciplina de gestão de projetos estar associada a Administração, as três revistas com mais publicações são associadas a engenharia e tecnologia (*IEEE Access* e *Journal Of Systems And Software*) e uma de gestão de projetos (*International Journal Of Project Management*). Na próxima seção serão discutidos artigos mapeados no corpus dessa pesquisa, a categorização que os artigos foram submetidos e a discussão dos achados dos artigos que compõem as categorias.

2. Análise e discussão das categorias encontradas

Após o mapeamento dos artigos que constituíram o *corpus* de análise, foi realizada uma análise aprofundada dos conteúdos publicados. A leitura e categorização dos conteúdos levou a classificação dos artigos em 04 categorias, apresentados na Tabela 3.

Tabela 3. Classificação dos artigos em 4 categorias

Categoria	Descrição	Qtde	Autores
Métricas de Projetos Tradicionais	Apresentam os artigos com ênfase nas métricas de desempenho de projetos tradicionais.	40	Vanhoucke (2011); Chen (2021); Aljedaibi e Khamis (2019); Colin <i>et al.</i> , (2015); Rajagopalan e Srivastava (2018); Chen, Chen e Lin (2016); Fleche <i>et al.</i> (2017); Sharma, Caldas e Mulva (2021); Batselier e Vanhoucke (2015); Chen (2018); Pajares e Lopez-Paredes (2011); Vanhoucke (2019); Colin e Vanhoucke (2014); Song, Marten e Vanhoucke (2021); Kerkhove e Vanhoucke (2017); Geng <i>et al.</i> (2018); Sato e Hirao (2013); Staron e Meding (2016); Spijkman <i>et al.</i> (2021); Ibrahim, Hanna e Kievet (2020); Chen <i>et al.</i> (2020); Ko e Cho (2020); Yousefi <i>et al.</i> (2019); Orgut <i>et al.</i> (2018); Kristiansen e Ritala (2018); Rajablu <i>et al.</i> (2017); Han, Choi e O'Connor (2017); Hanna (2016); Kim, Kim e Cho (2016); Hazır, (2015); Wauters e Vanhoucke (2015); Grau e Back (2015); Chen, H. L. (2014); Narbaev e De Marco (2014); Jethani (2013); El Asmar, Hanna e Loh (2013); Hanna, Tadt e Whited (2012); Iqbal, Naeem e Khan (2012); Pozzana <i>et al.</i> (2021); Colin e Vanhoucke (2015).
Métricas Ágeis de Fluxo	Apresentam os artigos com ênfase nas métricas ágeis e com o foco em Fluxo.	9	Grimaldi <i>et al.</i> (2016); Dennehy e Conboy (2018); Mas, Mesquida e Pacheco (2020); Ahmed <i>et al.</i> (2017); Sajedi-Badashian e Stroulia (2020); Petersen e Wohlin (2011); Choraś <i>et al.</i> (2020); Meidan <i>et al.</i> (2018); Sadler (2020).
Métricas Ágeis de Backlog do Produto	Apresentam os artigos com ênfase nas métricas ágeis e com o foco no backlog do produto.	9	Azzeh e Nassif (2016); Wallace e Sheetz (2014); Parrend <i>et al.</i> (2014); Kayes, Sarker e Chakareski (2016); Zheng <i>et al.</i> (2021); Pavlova <i>et al.</i> (2021); Antinyan (2014); Prakash e Viswanathan (2018); Asha e Mani (2018).
Métricas Ágeis de Produto	Apresentam os artigos com ênfase nas métricas ágeis e com o foco no produto e em sua qualidade.	25	Nidagundi e Novickis (2016); Mäkiäho, Vartiainen, e Poranen (2022); Chang (2015); Mascarenhas Hornos <i>et al.</i> (2014); Singh, Singh e Singh (2019); Simpson <i>et al.</i> (2021); Day <i>et al.</i> (2019); Üsfekes, <i>et al.</i> (2019); Abdelrahman Aljemabi, Wang e Saleh (2020); Costa, Vasconcelos, e Fragoso (2020); Ibba <i>et al.</i> (2018); Concas <i>et al.</i> (2012); Shawky e Abd-El-Hafiz (2016); Tahir, Rasool e Noman (2018); Isong e Obeten (2013); Tanveer <i>et al.</i> (2019); Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018); Yamashita e Counsell (2013); Bakhtiary, Gandomani e Salajegheh (2020); Hernández <i>et al.</i> (2020); Athanasiou <i>et al.</i> (2014); Gao <i>et al.</i> (2011); Perkusich <i>et al.</i> (2017); Savola, Frühwirth e Pietikäinen (2012).
	Total	83	

Fonte: dados da pesquisa, 2022.

As quatro categorias encontradas compreendem uma abstração observada nas leituras dos artigos. Cabe explicitar que o processo de análise inicial dos artigos foi individual e, depois, para seleção das categorias passou-se pela análise dos pesquisadores desta pesquisa. O

processo de categorização foi realizado pela discussão e estabelecimento de um consenso sobre qual era a maior aderência de todos os artigos em cada uma das categorias constituídas.

Na próxima seção, serão discutidas as categorias para explicitar os achados desta pesquisa e as categorias aqui apresentadas representam uma forma de explicação e organização dos conteúdos estudados.

3.2.1 Métricas de Projetos Tradicionais

Como visto, as práticas de gerenciamento de projetos tradicionais se baseiam, para medir o desempenho dos projetos, em um conjunto de métricas orientados frente a um planejamento prévio. Desse modo, as métricas são caracterizadas, em sua essência, utilizando as medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido. Tais medidas são comparadas com a posição atual do projeto (trabalho realizado) em relação ao planejado até o momento da medição (trabalho realizado).

Frente ao uso das ferramentas tradicionais de gestão de projetos, Vanhoucke (2011) apresentou como métrica, no Monte-Carlo para simular o progresso do projeto sob a estrutura analítica do projeto (EAP), utilizando dados do gerenciamento de valor agregado – (EVM) como possíveis sinais de alertas e acionadores para a necessidade de ações corretivas. Ainda sob o olhar da EAP, Colin e Vanhoucke (2015) apresentaram um modelo multivariado adotando como métrica o EVM/ES, resultando no *Earned Value Management/Earned Schedule*. A métrica é utilizada pelo gerente caso o projeto apresente algum risco de maneira preventiva. Kerkhove e Vanhoucke (2017) adotaram a técnica *Earned Incentive Metric*, empregando o EVM/ES com incentivo de incentivos de custo e/ou tempo. Como resultado, os autores destacaram que *Earned Incentive Metric* apresentou em papel mais significativo que o EVM/ES em uma amostra de 4.200 projetos com tamanhos e estruturas de contrato diferentes, ou parametrizando o tamanho de um projeto grande (Chen *et al.*, 2020).

Sobre a adoção do EVM como métrica, Chen, Chen e Lin (2016) concluíram que a melhoria no poder preditivo do *Planned Value* (PV), calculado pelo erro percentual médio absoluto (*Mean Absolute Percentage Error* - MAPE), antes da execução do projeto fornece aos gestores informações preditivas confiáveis no desempenho do *Earned Value* (EV) e *Actual Cost value* (AC), permitindo uma ação proativa efetiva para garantir resultados de desempenho favoráveis. Chen (2021) e Hazır (2015) adotaram como métrica variação de

desempenho na fase de planejamento que explica uma parte substancial da dificuldade do projeto na conclusão, mediante a um modelo de classificação de problemas de projetos, com o objetivo de identificar problemas nos projetos que provavelmente falharão antes de eles começarem. No mesmo sentido, Sharma, Caldas e Mulva (2021) utilizaram o EVM nas áreas de custo, cronograma, segurança, retrabalho e mudanças no projeto com o propósito de apoiar à tomada de decisão, facilitar o orçamento de capital, o impacto das condições de mercado e na melhoria contínua e melhores práticas de gestão.

Nessa mesma linha de métrica, Yousefi *et al.* (2019) apresentam o controle estatístico em índices do *Earned Duration Management* (EDM) para melhor investigar as variações do desempenho do cronograma do projeto ou para controlar projetos de inovação radical (Orgut *et al.*, 2018). Ainda em relação ao cronograma, Han *et al.* (2017) destacam o uso de métricas de qualidade de cronogramas baseados no limite de duração máxima (30 dias); Proporção de tarefas detalhadas para marcos; Porcentagem de tarefas no caminho crítico; Número de defasagens; Tipo de relacionamento (terminar para começar); e Marcos sem predecessor ou sucessor. Já Wauters e Vanhoucke (2015) criaram um experimento computacional para avaliar a estabilidade usando a nova métrica de defasagem média proposta usando o *coefficient of variation* (CV) do EVM. Grau e Back (2015), no mesmo contexto, destacam que as métricas de controle do cronograma devem apoiar o gestor a ser proativo, em vez de reativo, no reconhecimento de tendências e eventos melhora significativamente a capacidade da equipe de minimizar a diferença entre os resultados previstos e reais. Chen (2014) combinou séries temporais e análise de regressão para desenvolver PV em um modelo de resposta EV para prever *work perform* (wp) de EV. Usando a relação entre *budgeted cost* (bc) e o wp estimado representado no modelo de resposta EV, Chen (2014) desenvolveu ainda EV em um modelo AC para prever *actual cost* (ac) de AC.

Em relação a duração do projeto, Batselier e Vanhoucke (2015) adotaram duas técnicas baseadas no EVM que integram, respectivamente, retrabalho e sensibilidade à atividade no tempo de EVM prevendo as extensões do projeto. Lopez-Paredes (2011) introduziram duas novas métricas denominadas por Índice de Controle de Custos e Índice de Controle de Cronograma. Ambos os índices compararam as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o risco, custo e cronograma do projeto, o que pode ser aplicado nas fases de iniciação, planejamento, execução, monitoramento e controle e encerramento

(Chen, 2018). Sato e Hirao (2013) contribuíram para estabelecer o conceito de orçamento ótimo e esclarecer essa condição e a teoria permite a análise do equilíbrio custo/risco não apenas no nível do projeto, mas também no nível da rede de atividades. Kim, Kim e Cho (2016) estudaram um processo com relação linear entre atividades em que as saídas de um negócio precedente se tornam diretamente os insumos de um sucessor. Narbaev e De Marco (2014) calcularam o *cost estimate at completion* (CEAC) que é um número considerado como novidade de pesquisa e que contribui para a extensão do corpo de conhecimento EVM. Narbaev e De Marco (2014) apresentaram uma nova abordagem metodológica que foi desenvolvida para fornecer previsões CEAC mais precisas, confiáveis e eficazes para os estágios iniciais e intermediários da execução do projeto de construção.

Em relação ao resultado dos projetos, Aljedaibi e Khamis (2019) desenvolveram o *Critical Success Factors* (CSF-Live!), é um método para medir, monitorar e controlar fatores críticos de sucesso de sistemas de *software* de grande, por meio de escala de metas, custos, infraestrutura de TI, gestão de projetos, entre outros. Já Rajagopalan e Srivastava (2018) adotaram um método único usando uma variável composta recém-definida chamado de *Project Health Index*, capaz de medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base no *Corrective And Preventive Actions*. Ibrahim, Hanna e Kievet (2020) examinaram as diferenças de desempenho tempo de processamento de pedido de mudança, aumento no prazo do cronograma, aumento de custo, mudança percentual do projeto, qualidade geral dos sistemas do projeto, mudança percentual relacionada à qualidade/valor, mudança percentual relacionada ao *design*.

Como a fase de planejamento prevê o uso de recursos humanos, algumas métricas da abordagem tradicional foram introduzidas no gerenciamento de projetos. Song, Martens e Vanhoucke (2021) a métrica de *Using Schedule Risk Analysis* (SRA) com restrição de recursos (*Resource Constrained* - RC) para melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos. Pozzana *et al.* (2021) apresentaram um conjunto de dados composto por projetos de engenharia descritos por suas redes de atividades, registros do desempenho de cada atividade, medindo se a atividade foi entregue no prazo ou atrasada.

Em se tratando do envolvimento das métricas com os envolvidos nos projetos, Rajablu *et al.* (2017) desenvolveram o *Stakeholder Metrics-integrated Management Model* que integra as partes interessadas que permite gerenciar as partes interessadas identificando,

comunicando, engajando, capacitando e atendendo as partes interessadas com métricas e ferramentas. Além disso, Hanna, Tadt e Whited (2012) utilizaram análise *bootstrap* avaliar o desempenho de um projeto de infraestrutura. Hanna (2016) avaliaram com métrica o desempenho geral mais alto no que se refere à implementação de técnicas de construção enxuta, equipe de liderança de projeto engajada e capacitada e um alto de envolvimento das partes interessadas em todo o projeto.

Algumas métricas possuem foco no desenvolvimento do produto. Fleche *et al.* (2017) implantaram métricas quantitativas e não intrusivas para os *designers*, onde cada métrica utiliza variáveis do *design* do produto planejado. Spijkman *et al.* (2021) propuseram o modelo *Requirements Engineering for Software Architecture* (RE4SA) que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. Além disso, Spijkman *et al.* (2021) forneceram métricas para quantificar o alinhamento entre requisitos e arquitetura para detectar sinais de granularidade, concentrando-se em valores discrepantes em um conjunto de dados. Jethani (2013) apresentou o esforço no nível da tarefa e monitoramento do progresso; Monitoramento de marcos e cronogramas; e Monitoramento de defeitos, com base nas atividades regulares de planejamento e monitoramento de projetos de desenvolvimento de *software*. A visão é estendida por Iqbal, Naeem e Khan (2012), que completaram que a rastreabilidade avalia os vínculos entre os requisitos dentro de um documento e métricas de integridade de requisitos verificam se os requisitos especificados estão completos ou não para o desenvolvimento do produto.

Em ambiente de escritório de projetos, Geng *et al.* (2018) integraram o aprendizado organizacional e o desenvolvimento do conhecimento com um processo de seleção de projetos, propondo uma estrutura para avaliar essa integração baseada no *Analytic Hierarchy Process*, ou por meio de uma avaliação de eficiência precisa ser realizada por meio da atualização da lista de análise de projetos (Ko e Cho, 2020).

Os artigos dessa categoria apresentaram discussões importantes sobre as principais métricas utilizadas pelas organizações para o controle e monitoramento de projetos tradicionais. A partir dessa perspectiva, as métricas como EVM, SPI, EV, AC, PV e ES são as mais citadas na literatura e nota-se a introdução de novas métricas como EDM e métricas com *benchmarks* associados para o processo de RFI.

3.2.2 Métricas Ágeis de Fluxo

As métricas ágeis de fluxo visam medir o desempenho de projetos ágeis durante o fluxo de desenvolvimento de software ou de produto. Assim, as métricas ágeis de fluxo procuram medir desde o ponto de comprometimento (*Sprint Backlog*) até a entrega da funcionalidade para os clientes/*stakeholder* e comumente utilizada o *Lead Time* e a *Cycle Time* que procura medir etapas de desenvolvimento.

Inserido no Scrum, Parrend *et al.* (2014) avaliaram o impacto das práticas de gerenciamento de projetos nos dois primeiros *Sprints*, medindo as práticas de elegibilidade do incremento com impacto mensurável no desempenho das equipes e nível de exigência que levam ao fracasso se não forem implementadas. Parrend, *et al.* (2014) destacaram a importância dos princípios do *Scrum* e as propriedades *swarming* são práticas relevantes de sucesso e um sintoma de gerenciamento de desempenho. O *swarming* é uma técnica que consiste em empregar a inteligência coletiva da equipe do projeto na execução de uma atividade crítica do projeto (Parrend, *et al.* 2014).

Dennehy e Conboy (2018) realizaram a avaliação por meio de um estudo empírico do uso de fluxo de trabalho com *Kanban*, obtendo 2 resultados. Primeiro, um conjunto de desafios empíricos, tais como a mudança organizacional e a configuração de quadros *Kanban*. Segundo identificaram que as métricas de *Lead Time* e *Velocity* são as mais utilizadas para o planejamento das demandas e nos processos de melhoria contínua no projeto (Dennehy & Conboy, 2018).

Mas, Mesquida e Pacheco (2020) realizaram um estudo para saber se empresa é ágil e madura e se um conjunto de métricas de desempenho estiver em conformidade com as melhores práticas dos processos de gerenciamento de projetos. Um conjunto de métricas foram identificadas nessas empresas ágeis, como *efficiency flow*, *lead time*, *cycle time*, *reopens* e *tasks finished flow* (Mesquida & Pacheco, 2020). A métrica *efficiency flow* mede quantos itens do *backlog* são entregues por ciclo (Mesquida & Pacheco, 2020). O *lead time* mede o tempo em que um item do *backlog* é iniciado até o momento que é entregue ao cliente (Mesquida & Pacheco, 2020). A métrica *reopens* mede o número de tarefas ou incidentes reabertos e a métrica *tasks finished flow* mede as tarefas finalizadas dentro do fluxo de desenvolvimento (Mesquida & Pacheco, 2020).

Grimaldi *et al.* (2016) apresentou como uma estrutura ágil escalável, baseada em SAFe 3.0 e mostrou como implementar e medir o desempenho onde as funções e responsabilidades são divididas entre a organização e o fornecedor. Grimaldi *et al.* (2016) utilizam métricas como eficácia e eficiência, custo-hora real e planejada, impedimentos e custo do atraso durante as rodadas de *Sprint*.

Ahmed *et al.* (2017) discutiram sobre as métricas *Sprint Burn Down*, *Epic Burn Down Velocity Chart*, *Control Chart* e muitos dos cálculos de métricas são fortemente acoplados aos pontos de história para itens no épico ou *Sprint*. Segundo Ahmed *et al.* (2017), as métricas desempenham um papel fundamental no desenvolvimento e ajudam a acompanhar o progresso das equipes. Além disso, Ahmed *et al.* (2017) destacaram que ainda restam muitas métricas relacionadas ao *Scrum* e estimativas que precisavam ser pesquisadas para apresentá-las ao cenário de desenvolvimento atual para impulsionar a equipe e o processo ágil.

O trabalho de Sajedi-Badashian e Stroulia (2020) forneceu um método *Multisource* que permitiu o uso de dados de diferentes contextos além de fornecer uma maneira de investigar o valor da informação de diferentes fontes e veja se elas podem ser úteis na recomendação do responsável. Assim, Sajedi-Badashian e Stroulia (2020) mostraram que é possível melhorar a eficácia do processo de atribuição de *bugs* com a métrica de atribuição de maneira a atribuir a pessoa mais indicada para a resolução do problema e reduzindo o *time to market* do produto.

Do ponto de vista de Petersen e Wohlin (2011), o *Lean* fornece ferramentas de análise e melhoria com foco no ciclo de vida do desenvolvimento, enquanto o ágil se concentra em soluções e prescreve conjuntos de práticas para obter agilidade. A principal contribuição de Petersen e Wohlin (2011) foi a definição de novas medidas conectadas aos diagramas para atingir os seguintes objetivos: (1) reduzir o *lead time* para obter alta capacidade de resposta às necessidades dos clientes e (2) fornecer um sistema de rastreamento que mostre o status do desenvolvimento de produtos de *software*.

Choraś *et al.* (2020) relataram experiências práticas no uso de métricas relacionadas a cada fase do processo de desenvolvimento de software ágil. Contribuíram Choraś *et al.* (2020) que dado que o *software* precisa ser customizado para cada cliente, enquanto os domínios de negócios dos clientes variam significativamente. As métricas de processo propostas por Choraś *et al.* (2020) foram utilizadas continuamente para ajudar a garantir a qualidade e estabilidade do software como a estimativa de tarefas e rastreamento tarefas em tempo real.

Meidan *et al.* (2018) identificaram que o desenvolvimento ágil e empresa de pequeno/médio porte foram os contextos de pesquisa mais frequentemente identificados. Os autores demonstraram que a capacidade, desempenho e maturidade foram os atributos de processo mais medidos, enquanto esforço, desempenho e risco foram os atributos de projeto mais medidos.

O modelo de ciclo de vida de desenvolvimento e manutenção de sistemas proposto por Sadler (2020) aplicou efetivamente diferentes níveis de rigor e métricas de qualidade mais rígidas, dependendo do risco associado aos lançamentos. O modelo de Sadler (2020) foi baseado nas normas relacionadas ao gerenciamento de riscos denominado ER²C SDMLC (*Enterprise Release Risk-Centric Systems Development and Maintenance Life Cycle*). Dentre as métricas avaliadas por Sadler (2020), temos a Clareza do Artefato e Envolvimento e Propriedade do Cliente receberam as classificações médias mais baixas, enquanto Aplicabilidade e Rastreabilidade receberam as mais altas. Apesar Sadler (2020) ter acompanhado os modelos de artefatos, a abertura do modelo de ciclo de vida significa que os profissionais são livres para adaptar os artefatos enquanto, em alguns casos, os profissionais gostam de usar modelos de artefatos finalizados, o que pode explicar por que a Clareza do Artefato é entre as métricas com classificação mais baixa para ER²C SDMLC.

Asha e Mani (2018) apresentaram uma metodologia de teste para aplicar técnicas de Desenvolvimento Orientado a Testes de Aceitação (DOTA) denominada como Teste Simples de Teste de Aceitação (TSTA) que podem ser executados como casos de teste de software no contexto do DevOps. Além disso, Asha e Mani (2018) utilizaram os algoritmos Tarjan e Breadth-First Search que renderizam 100% de cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste aplicando as métricas de componentes funcionais, aumentando a frequência de entrega contínua e reduzindo o *time to market*.

Os artigos dos autores apresentaram as métricas ágeis relacionadas com o fluxo operacional de projetos utilizando o *framework Scrum* e o *Kanban* são as métricas de *Efficiency Flow*, *Lead Time*, *Velocity*, *Sprint Burn Down*, *Epic Burn Down*, *Velocity Chart*, *Control Chart* e muitos dos cálculos de métricas são fortemente acoplados aos pontos de história. Além disso, métricas ligadas os itens de *backlog*, como *reopens*, *tasks finished flow* e *time spent extremes*, e outras métricas baseadas no *framework SAFe 3.0* como eficácia, eficiência, custo-hora real e planejada, resíduos, impedimentos e custo do atraso.

3.2.3 Métricas Ágeis de Backlog do Produto

As métricas ágeis de *Backlog do Produto* visam medir o desempenho de projetos ágeis durante a etapa de definição do escopo do projeto que, em projetos ágeis, são organizados no *backlog* do produto. Assim, as métricas ágeis de *Backlog do Produto* procuram medir o esforço de desenvolvimento da funcionalidade, nível de priorização e complexidade por *Use Case Points*, e estimativas por pontos de história.

Kayes, Sarker e Chakareski (2016) propuseram uma métrica no *Product Backlog Rating* (PBR) para avaliar o processo de teste no *Scrum* que forneceu uma estrutura adicional para o gerenciamento da qualidade e uma visão abrangente do processo de teste ao longo do ciclo de desenvolvimento do produto. O PBR de Kayes, Sarker e Chakareski (2016) considera a complexidade dos recursos a serem desenvolvidos em uma iteração do *Scrum*, avalia as classificações dos testes e oferece uma pontuação numérica do processo de teste.

A pesquisa de Zheng *et al.* (2021) mapeou as métricas para o processo *Scrum* e métricas de *Service-Oriented Architecture* (SOA) para auxiliar nas decisões de planejamento e *design* no planejamento e gerenciamento durante a *Sprint*. Zheng *et al.* (2021) sugeriram os *Key Performance Indicators* (KPIs) para ajudar os *stakeholders* a entender os imperativos da técnica proposta e os ajudarão no planejamento de recursos, bem como nos principais marcos para testar o desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos baseados em *Scrum* e SOA.

Pavlova *et al.* (2021) determinaram os parâmetros do projeto como as características da entrega futura como o custo, prazo, possíveis desvios de prazo e custo, e tais parâmetros resultam em métricas para avaliar o grau de singularidade do projeto e o grau de complexidade do projeto. Para a métrica de complexidade, Pavlova *et al.* (2021) consideraram o número de variantes de soluções tecnológicas, o número de operações para cada variante e o número de alternativas para execução da operação para cada variante da tecnologia do produto do projeto.

Wallace e Sheetz (2014) determinaram as propriedades de uma medida de software levam à sua adoção por engenheiros e gerentes de software e destacaram as medidas de esforço de desenvolvimento de software. Wallace e Sheetz (2014) destacaram que a utilidade percebida de uma medida de software pode ser medida por uma avaliação da aplicabilidade da medida ao longo do ciclo de vida, dependência de uma linguagem de programação específica,

capacidade de prescrever soluções e a validade da medida.

Azzeh e Nassif (2016) propuseram um modelo híbrido voltado para esforço e produtividade para apoiar a estimativa de esforço inicial baseada no tamanho do *Use Case Point* (UCP) e utilizou um número significativo de projetos coletados que utilizaram como base o UCP.

A ontologia foi desenvolvida por Prakash e Viswanathan (2018) acumulou conhecimento sobre os critérios de avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos. Prakash e Viswanathan (2018) incluíram axiomas que determinam a qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.

As métricas de *Backlog* de Produto apresentadas nos artigos, trazem métricas importantes para a gestão de requisitos em projetos ágeis. Assim, destacam-se as métricas como o *Product Backlog Rating*, qualidade e valor dos negócios, complexidade de projetos baseados em *Scrum* e SOA, complexidade do requisito, complexidade da arquitetura de solução e complexidade da mudança. Outros achados tivemos com a estimativa de esforço inicial baseada no tamanho do *Use Case Point* (UCP), avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos e a medição do tamanho do requisito a partir do comportamento criado pelo BDD usando a notação de Gherkin.

3.2.4 Métricas Ágeis de Produto

As métricas ágeis de produto visam medir a qualidade de construção e de ações para produzir um produto de melhor qualidade e para reduzir o seu *time to market*. Dessa forma, as métricas ágeis de produto nessa sessão medem tanto atividades para melhorar a qualidade do código-fonte, melhoria do processo de testes de software, a gestão e rastreabilidade de bugs e o processo de desenvolvimento orientado a testes. Além disso, a categoria buscou apresentar as métricas para medir o nível de satisfação do cliente/*stakeholder* com o produto entregue.

Mäkiahho, Vartiainen e Poranen (2022) apresentaram uma ferramenta de monitoramento de métricas que foi desenvolvida por alunos de graduação e pós-graduação para apoiar os membros do projeto, gerentes de projeto e gestores. Segundo Mäkiahho, Vartiainen e Poranen (2022) a ferramenta permitiu acompanhar a quantidade de *commits*,

horas trabalhadas e se os requisitos permanecem muito tempo em determinados estados no processo de desenvolvimento.

Os experimentos de Chang (2015) que os modelos de risco de software obtidos foram usados para detectar itens de risco de software a partir do *cluster* de dados oriundos de projetos. Tais dados foram apresentados por Chang (2015) no formato de métricas e os riscos foram mitigados e fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de *software*.

Dentre as métricas apresentadas por Mascarenhas *et al.* (2014), temos o crescimento nas vendas de novos produtos, o orçamento dedicado à análise ou verificação do cliente, projetos orientados para o cliente, projetos/programas respeitando custos e orçamento, marcos respeitados e componentes reutilizáveis.

A pesquisa de Singh *et al.* (2019) propuseram análises de complexidade de programas orientados a objetos em Java através da métrica Chidamber-Kemerer (CK) e desempenho são realizadas usando a métrica *Mean Time to Execution*. Como resultado indicado por Singh *et al.* (2019), realizar reengenharia usando práticas ágeis é benéfico em termos de implementação de requisitos, estimativa de custo e melhoria de desempenho. Além disso, Singh *et al.* (2019) sugerem que as estimativas de custos são realistas e envolvem um consenso de todas as partes interessadas. Isong e Obeten (2013) realizaram uma validação empírica existente e publicada das métricas *Chidamber-Kemerer* (CK) e *Source Lines of Code* (SLOC) e se elas têm ou não influência sobre PF de classes OO. A validação empírica de Isong e Obeten (2013) das métricas CK e SLOC e os modelos que as utilizaram em três aspectos: (1) nível de significância e insignificância com propensão a falhas (PF), (2) estado de validação e (3) utilidade na qualidade de software. Segundo Isong e Obeten (2013), SLOC, *Coupling between Objects* (CBO), *Response for a Class* (RFC), *Weighted Methods per Class* (WMC) são métricas com forte relação com PF e são consideradas os melhores preditores de PF. Concas *et al.* (2012) também estudou diversas iniciativas em projetos ágeis que utilizaram as métricas CK, SLOC, CBO, RFC, WMC, entre outras.

A experiência de Day *et al.* (2019) utilizaram o *Net Promoter Score* (NPS) relataram que é viável ter um sistema de apoio à decisão no Registro Eletrônico de Saúde (RES) para testes de usabilidade e a implementação em larga escala pode ter um efeito positivo no paciente e melhorar a tomada de decisão compartilhada. Simpson *et al.* (2021) procurou reduzir a carga do RES com um *Sprint* de otimização de 2 semanas que trabalhou com 19

provedores de prática avançada em 1 unidade de especialidade. Simpson *et al.* (2021) mediram o NPS o tempo gasto no EHR com base nos dados de log do usuário, onde o NPS aumentou de 6 para 60 e para o *Sprint* o NPS foi de 93.

Üsfekes *et al.* (2019) investigou através da revisão da literatura a utilidade de mecanismos de incentivo para rastreamento eficiente de *bugs* no *Application Lifecycle Management*. Para Üsfekes *et al.* (2019), isso colabora positivamente para os resultados de métricas que olham para a quantidade de *bugs* resolvidos por unidade de tempo. Abdelrahman Aljemabi *et al.* (2020) mostraram que a *Bug Tracking System-based*, exibem três diferentes níveis de padrões de coordenação (Plano, Consciente e Reflexivo) com base em atividades de colaboração e a métrica de tempo médio para reparo para avaliar o tempo de resolução de *bugs*.

Costa, Vasconcelos e Fragoso, B. (2020) destacaram as etapas de identificar as métricas aplicáveis à arquitetura dos projetos e quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos. As métricas foram propostas por Costa, Vasconcelos e Fragoso, B. (2020) para estimar as características de qualidade dos Sistemas de Informação, sendo organizadas em métricas de Funcionalidade, Confiabilidade, Eficiência, Manutenibilidade, Portabilidade e Alinhamento.

Ibban *et al.* (2018) entenderam as atividades de engenharia de software relacionadas a Ofertas Iniciais de Moedas (*Initial Coin Offering – ICO*) que é uma forma inovadora de financiar projetos baseados em *blockchain*. Além disso, Ibban *et al.* (2018) buscaram fazer uma comparação entre ICOs e ICOs ágeis e realizaram uma análise das ICOs ágeis em relação ao planejamento de projetos, desenvolvimento de software e recursos de código. Cerca de 5% das ICOs examinadas por Ibban *et al.* (2018) aplicam práticas ágeis e analisaram as ICOs ágeis em termos de métricas de código-fonte, versões de idioma e uso de ferramentas de teste. Ibban *et al.* (2018) concluíram que as práticas ágeis são adequadas para desenvolver ICOs que são projetos inovadores e requisitos não são totalmente compreendidos ou tendem a mudar. Nidagundi e Novickis (2016) descobriram possibilidades de adoção dos modelos e métricas de transformação *Lean* no plano de teste de software para simplificar o processo de teste para uso posterior dessas métricas de teste em Canvas. Com o Canvas, Nidagundi e Novickis (2016) afirmaram que as métricas de teste podem nos dar *feedback* para o processo de teste de software e dividiram o projeto completo em pequenos subcomponentes e testá-los, obteremos novas métricas de teste.

Shawky e Abd-El-Hafiz (2016) apresentaram uma abordagem que estuda o efeito do processo de desenvolvimento na qualidade dos sistemas de software e verificaram se existe relação entre o tipo de método de desenvolvimento utilizado e os valores de algumas métricas indicadoras de qualidade. A pesquisa dos autores afirmou que as métricas usadas incluem métricas de complexidade, acoplamento e coesão, sendo as duas últimas sendo utilizadas para medir a complexidade.

Tahir, Rasool e Noman (2018) identificaram 29 modelos de medição de software e 4 ferramentas projetadas para PMs em PMEs, sendo os modelos de medição (51%) referentes às melhorias de processo de software. A maioria dos estudos primários identificados por Tahir, Rasool e Noman (2018), discutiu o uso de especialistas em medição e experiência (60%), seguido pelo uso de padrões de medição (40%) e o uso de ferramentas automatizadas (22%) para seleção de métricas.

Tanveer *et al.* (2019) desenvolveram um método híbrido que utiliza informações de análise de impacto de mudança para melhorar a estimativa de esforço e modelo de estimação baseado em árvores impulsionadas por gradiente (AIG). A avaliação subjetiva de Tanveer *et al.* (2019) sobre o *Hybrid Effort Estimation in Agile Software Development* (HyEEASE) e do *Planning Poker* (PK) pelos alunos em facilidade de uso percebida, utilidade e aprendizagem indicam resultados positivos. Tanveer *et al.* (2019) avaliaram que o PK obteve classificações superiores ao HyEEASE em facilidade de uso percebida e capacidade de aprendizado, entretanto, os alunos consideraram que HyEEASE requer menos esforço mental. Além disso, em relação à precisão da estimativa, Tanveer *et al.* (2019) concluíram uma estimativa puramente baseada no modelo AIG não teve um desempenho melhor do que uma estimativa baseada em especialistas suportado pelo HyEEASE.

Pradhan e Nanniyur (2021) apresentaram um estudo de caso sobre a transformação em larga escala de um sistema de gestão de qualidade legado para um sistema moderno com aprendizado de máquina, dados conectados, operações integradas e *Big Data*. Pradhan e Nanniyur (2021) identificaram que as métricas de qualidade precisam abranger a Sprint/Projeto, Release, Solução, Portfólio e Visão Empresarial na Cisco Systems e os resultados validaram a eficácia da transformação nos perfis de defeitos recebidos, taxas de falhas e cronograma de maturidade de lançamento.

Kamulegeya, Mugwanya e Hebig (2018) buscaram entender o uso e os benefícios percebidos da medição em *startups* de software na África Oriental e identificaram que as

start-ups estão usando uma série de métricas orientadas a negócios e produtos e que as *startups* mais antigas demonstraram também desejar métricas orientadas a projetos e as métricas organizacionais são as métricas menos utilizadas.

A pesquisa de Yamashita e Counsell (2013) investigou o potencial de *code smells* (potenciais problemas no código) para refletir indicadores de manutenibilidade em nível de software. Ao comparar os *code smells* com outras abordagens, Yamashita e Counsell (2013) descobriram que as medidas estruturais forneceram mais informações sobre qual sistema tinha o *design* equilibrado, mas essa medida ignorou quando as tarefas de manutenção eram de tamanho pequeno ou médio.

Bakhtiary, Gandomani e Salajegheh (2020) mediram a aplicação das abordagens de *Test-Driven Development* (TDD) e Não-TDD (NTDD) para compará-los, dada a qualidade externa e interna do resultado do trabalho e a produtividade dos desenvolvedores em uma empresa de software. Bakhtiary, Gandomani e Salajegheh (2020) mostraram que ao aumentar a escala do projeto, o número de defeitos descobertos antes e após o lançamento aumentou mais na abordagem NTDD em relação ao TDD. Bakhtiary, Gandomani e Salajegheh (2020) observaram que houve uma redução na satisfação do cliente no NTDD em comparação com a abordagem TDD em cada ciclo, e teve desempenho melhor. Outra análise de Bakhtiary, Gandomani e Salajegheh (2020) e Antinyan (2014) trouxeram que medir a complexidade ciclomática mostraram que a abordagem TDD reduziu a complexidade no código em comparação com a abordagem NON-TDD.

Hernández, *et al.* (2020) estabeleceu o nível de relevância das métricas de produtividade de equipes *Agile Software Development*. Para atingir o objetivo, o nível de relevância das métricas de produtividade da equipe identificado por Hernández *et al.* (2020) em três aspectos fundamentais: entrega antecipada e frequente de software, agregação de valor e aspectos relacionados ao esforço. Hernández, *et al.* (2020) sugeriram que é relevante para os profissionais medir a produtividade entregando software com antecedência e frequência com alta relevância o esforço de implantação de software que resolva os problemas de clientes. Hernández, *et al.* (2020) afirmaram que o cálculo sistemático do retorno do investimento é visto como uma forma relevante de medir a agregação de valor.

O estudo de Athanasiou *et al.* (2014) apresentou um modelo que avalia a qualidade do código de teste combinando métricas de código-fonte que refletem três aspectos principais da qualidade do código de teste: integridade, eficácia e manutenibilidade. Além disso,

Athanasidou *et al.* (2014) encontraram uma correlação positiva e estatisticamente significativa entre a qualidade do código de teste e o rendimento e a produtividade. As contribuições do estudo de Athanasidou *et al.* (2014) iniciaram a partir do modelo que combina métricas para fornecer uma medida da qualidade do código de teste e a investigação empírica que demonstrou uma correlação positiva significativa entre a qualidade do código de teste e no tratamento de *bugs*.

Com o enfoque no problema de seleção de atributos no contexto da estimativa de qualidade de software, o estudo de Gao *et al.* (2011) investigou comparativa é apresentada para avaliar nossa proposta de abordagem híbrida de seleção de atributos. Dado esse contexto, Gao *et al.* (2011) destacaram que uma seleção inteligente de métricas de software antes de treinar um modelo de previsão de defeitos provavelmente melhorará o resultado, removendo recursos redundantes e menos importantes. Uma inferência final feita por Gao *et al.* (2011) é que, mesmo após a remoção de 85% do número disponível de métricas de software, os modelos de previsão de defeitos não foram afetados negativamente.

Perkusich *et al.* (2017) sugeriram que o modelo abordasse a avaliação da qualidade de processo e de produto a partir de métricas e Redes Bayesianas. Os resultados dos nós da rede e métricas de qualidade do incremento, dos eventos do Scrum e do código-fonte, Perkusich *et al.* (2017) identificaram nos nós uma oportunidade para analisar a qualidade do trabalho da equipe do projeto e avaliar as chances de sucesso do projeto.

Savola, Frühwirth e Pietikäinen (2012) demonstraram o potencial prático das métricas de segurança orientadas a riscos, particularmente ao oferecer alguma visibilidade inicial da eficácia e eficiência da segurança. A principal contribuição do estudo de Savola, Frühwirth e Pietikäinen (2012) está na avaliação do potencial e significado de uma metodologia de desenvolvimento de métricas de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de software.

Os artigos apresentaram métricas que avaliaram diferentes aspectos na avaliação no desenvolvimento de produto. Assim, estudos apresentaram métricas baseadas em modelos de risco que podem ser usados para entender as características dos riscos de software, crescimento nas vendas de novos produtos, medição de projetos orientados para o cliente, projetos/programas respeitando custos e orçamento, marcos respeitados e componentes reutilizáveis. No tocante a complexidade de programas orientados a objetos, autores propuseram as métricas de *Chidamber-Kemerer* e *Coupling between Objects* para avaliar o

nível de significância e insignificância com propensão a falhas, estado de validação e utilidade na qualidade de software como *bugs* resolvidos por unidade de tempo. Outras métricas aplicáveis à arquitetura dos projetos e para quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos, como métricas de funcionalidade, confiabilidade, eficiência, manutenibilidade, portabilidade e alinhamento de negócio. Para rastrear as abordagens ágeis, outros estudos apresentaram as métricas de complexidade, acoplamento e coesão. Mediram a satisfação de clientes em projetos que usam TDD, onde os autores mostraram que a utilização do TDD gerou aumento de desempenho nos resultados de satisfação de clientes. Outros *autores* sugeriram, para avaliar o nível de satisfação de clientes em relação ao produto, identificou-se uma grande adoção da medida do NPS.

A seção seguinte apresenta, para cada categoria, a classificação das métricas utilizadas para cada prática, tanto no ágil como o tradicional.

4.5. CLASSIFICAÇÃO DAS MÉTRICAS POR CATEGORIA

A Tabela 4 destaca a classificação das métricas encontradas e suas respectivas categorias mapeadas durante o processo de RSL, além das subcategorias e processos de medição mapeados após a análise dos resultados.

Tabela 4. Classificação das métricas por categoria e subcategoria.

Categoria	Subcategoria	Métrica	Processo de Medição
Métricas Ágeis de Backlog do Produto	Complexidade	- Grau de complexidade do projeto - KPI de Imperativos da técnica proposta	<ul style="list-style-type: none"> Com o número de variantes de soluções tecnológicas, auxilia no planejamento de recursos, testar o desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos.
	Prioridade	- Product Backlog Rating - Avaliação da qualidade das histórias de usuários (HU)	<ul style="list-style-type: none"> Mede o nível de prioridade do item do backlog e ajuda a determinar a qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.
	Monitoramento do Projeto	- Esforço de desenvolvimento de software - Esforço e produtividade - Grau de singularidade do projeto	<ul style="list-style-type: none"> Avaliação da aplicabilidade ao longo do ciclo de vida, dependência de uma linguagem de programação específica e capacidade de prescrever soluções. Determina as características da entrega futura (backlog) como o custo, prazo, possíveis desvios de prazo e custo.
Métricas Ágeis de Fluxo	Fluxo ou Processo	- Reduzir Lead Time - Lead Time - Cycle Time - Efficiency flow - Tasks finished flow - Control Chart - Capacidade do processo - Desempenho do processo - Maturidade do Processo	<ul style="list-style-type: none"> Mede a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente e permite medir quantos itens do backlog são entregues por ciclo. Permite reduzir o lead time para obter alta capacidade de resposta às necessidades dos clientes.

	Iteração ou Sprint	<ul style="list-style-type: none"> - Eficácia e eficiência - Sprint Burn Down - impedimentos - Custo do atraso - Custo hora real e planejada - Previsão por Planning Poker 	a) Realiza a medição por Sprint ou iteração considerando os pontos de história, nível de assertividade das estimativas, impedimentos, custos para medir a eficiência e eficácia do projeto.
	Incremento ou Artefato	<ul style="list-style-type: none"> - Clareza do Artefato - Envolvimento e Propriedade do Cliente - Aplicabilidade e Rastreabilidade 	b) Medir a clareza, rastreabilidade e aplicabilidade do artefato produzido pelo projeto, os níveis de envolvimento e propriedade do cliente em relação aos entregáveis do projeto.
	Estimativa e Esforço	<ul style="list-style-type: none"> - Epic Burn Down - Velocity - Estimativa de tarefas - Esforço no projeto - Status Desenvolvimento 	<ul style="list-style-type: none"> c) Métrica sobre pontos de história no Épico (abrangendo uma ou muitas Sprints), mede o desempenho (velocity) através dos pontos de história feitos por Sprint. d) Realiza a estimativa de tarefas para ajudar a garantir a qualidade e estabilidade do software.
	Monitoramento do Projeto	<ul style="list-style-type: none"> - Componentes funcionais - Desempenho no projeto - Risco no projeto 	e) Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste e medir o desempenho do projeto. Medir os riscos do projeto ágil para dar visibilidade ao backlog do produto e sua priorização.
	Qualidade do Produto	<ul style="list-style-type: none"> - Rastreamento de tarefas - Atribuição de bugs - Reopens 	f) Rastreamento de tarefas para ajudar a garantir a qualidade e estabilidade do software. Através da atribuição da pessoa mais indicada para a resolução do problema aberto ou reaberto e reduzindo o time to market do produto.
Métricas Ágeis do Produto	Monitoramento do Projeto	<ul style="list-style-type: none"> - Horas trabalhadas - Tempo nos estágios de desenvolvimento - Aspectos relacionados ao esforço - Características dos riscos de software - Segurança orientada a riscos - Projetos/programas respeitando custos e orçamento - Marcos respeitados 	<ul style="list-style-type: none"> g) Analisa a quantidade de horas trabalhadas durante um período (Sprint ou ciclo), o tempo do requisito nos estágios de desenvolvimento, marcos e esforço para entregar ao cliente. h) Fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de software.

	Qualidade do Código-Fonte	<ul style="list-style-type: none"> - Chidamber-Kemerer - Source Lines of Code - Coupling between Objects - Response for a Class - Weighted Methods per Class - Mean Time to Execution - Componentes reutilizáveis - Complexidade por acoplamento - Complexidade por coesão - Code smells - Qualidade do código-fonte - Complexidade ciclomática - Usou Test-Driven Development - Não usou Test-Driven Development - Quantidade de commits 	<p>i) Análises de complexidade de programas orientados a objetos, códigos-fonte publicados, de número de linhas de código, reuso de componentes e a avaliação da qualidade de implementação por melhores práticas orientadas a objetos com baixo acoplamento e alta coesão.</p>
	Qualidade de Software	<ul style="list-style-type: none"> - Bugs resolvidos por unidade de tempo - Nível de Confiabilidade - Nível de Eficiência/Eficácia - Nível de Eficácia - Nível de Integridade - Nível de Manutenibilidade - Nível de Portabilidade - Teste em Canvas - Qualidade por Sprint - Qualidade por Release/Incremento - Qualidade por Solução - Qualidade por Portfólio - Qualidade por Visão Empresarial - Qualidade dos Evento do Scrum - Modelo de previsão de defeitos 	<p>j) Realiza a medição da quantidade de bugs resolvidos por unidade de tempo e mede o nível de confiabilidade, eficiência, eficácia, integridade, manutenibilidade e portabilidade em termos de qualidade de software.</p> <p>k) Utiliza práticas Lean no plano de teste de software para simplificar o processo de teste.</p>
	Relação Cliente/Stakeholders	<ul style="list-style-type: none"> - Crescimento nas vendas de novos produtos 	<p>l) Mede os orçamentos com dedicação de análise e verificação do cliente e o número de projetos criados e orientados para o cliente.</p>

		<ul style="list-style-type: none"> - Projetos orientados para o cliente - Orçamento dedicado à análise ou verificação do cliente - Nível de Alinhamento da Funcionalidade - Agregação de valor - Net Promoter Score 	<p>m) Mede o nível de satisfação dos clientes em relação do produto recebido, sua experiência e agregação de valor.</p>
	Entrega Contínua	<ul style="list-style-type: none"> - Melhoria de processo de software - Entrega antecipada e frequente de software 	<p>n) Utilizou métricas de produto para avaliar melhorias no processo de desenvolvimento de software e a relevância da produtividade entregando software com antecedência e frequência.</p>
Métricas de Projetos Tradicionais	Análise do Valor Agregado	<ul style="list-style-type: none"> - EVM (Earned Value Management) - Work Perform 	<p>o) Mede o desempenho e o progresso de projetos, com a capacidade de combinar medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido.</p> <p>p) Combina séries temporais e análise de regressão para desenvolver PV (planned value) em um modelo de resposta EV (earned value) para prever work perform.</p>
	Qualidade do Produto	<ul style="list-style-type: none"> - Monitoramento de defeitos - Qualidade geral dos sistemas do projeto - Mudança percentual relacionada à qualidade/valor 	<p>q) Monitora os defeitos identificados durante o desenvolvimento do projeto e monitora a mudança do percentual relacionado à qualidade/valor do produto.</p>
	Monitoramento do Projeto	<ul style="list-style-type: none"> - Variação de Desempenho - Número de defasagens - EDM (Earned Duration Management) - Índice de Controle de Cronograma - Desempenho de cada atividade - Monitoramento de marcos e cronogramas - Análise bootstrap - Project Health Index - Esforço da tarefa e monitoramento - Índice de Controle de Custos - Análise do equilíbrio custo/risco - Análise de Risco do Cronograma com restrição de recursos - Critical Success Factors 	<p>r) Análise de inicialização para avaliar o desempenho de um projeto e a classificação de problemas de projetos para identificar problemas em projetos que provavelmente falharão antes de começarem.</p> <p>s) Calcula o número de defasagens (atrasos) no cronograma e mede os custos do projeto.</p> <p>t) Utiliza o conceito de orçamento ótimo no nível do projeto e no nível da rede de atividades.</p>

	Alinhamento Organizacional e Conhecimento Adquirido	- Seleção de Projetos	u) Integraram o aprendizado organizacional e o desenvolvimento do conhecimento através do processo de seleção de projetos.
	Requisitos do Produto	- Modelo RE4SA - Integridade de requisitos	v) Modelo Requirements Engineering for Software Architecture (RE4SA) que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. w) Avalia se os requisitos especificados estão completos ou não para o desenvolvimento.
	Análise da Qualidade do Cronograma	- Proporção de tarefas detalhadas para marcos - Porcentagem de tarefas no caminho crítico - Tipo de relacionamento (terminar para começar) - Marcos sem predecessor ou sucessor - Impacto Predecessor no Sucessor	x) Calcula o percentual de tarefas do projeto no caminho crítico do cronograma e investiga a proporção de tarefas do projeto que foram detalhadas para marcos específicos. y) Avalia os marcos do cronograma sem definição de predecessor e sucessor e calcula o tipo de relacionamento de tarefas predecessoras.
	Gestão de Mudanças	- Tempo de processamento de pedido de mudança - Mudança percentual relacionada ao design	z) Calcula o tempo de processamento de pedido de mudança no projeto e avalia a mudança percentual relacionada ao design.
	Relação Cliente/Stakeholders	- <i>Stakeholder Metrics-integrated Management Model</i>	aa) Integra as partes interessadas com métricas de identificação, comunicação, engajamento, capacitação e atendimento.

Fonte: dados da pesquisa, 2022.

Assim, um contraponto com objetivo de obter novas discussões para trabalhos futuros é necessário. Esse contraponto sobre as métricas apresentadas pela Tabela 4, precisa ser dividido em dois cenários dentro do contexto das práticas de gestão ágil de projetos: 1) faltam propostas de métricas que venham medir o *discovery*, *delivery* e a entrega contínua (*release*), mesmo que isoladamente, no desenvolvimento de um produto e 2) faltam propostas de um processo de medição das etapas de *discovery*, *delivery* e entrega contínua para que assim seja possível medir e monitorar o projeto desde a iniciação até a entrega em partes (Sprints) e/ou no encerramento do projeto.

Com base na Tabela 4 e adicionalmente aos cenários 1 e 2 acima apresentados, verifica-se ainda a lacuna de processos de gestão ágil de projetos que contribuam como um sistema ou *framework* capaz de ser um instrumento gerencial para tomada de decisão a partir de métricas estratégicas para a organização e para o desenvolvimento ágil de produtos. Os projetos ágeis que desenvolvem produtos precisam testar hipóteses para resolver problemas organizacionais ou de clientes e precisam ser medidas para a tomada de decisão de refutar ou seguir adiante com a estratégia.

4.6 CONSIDERAÇÕES FINAIS

A partir da pesquisa realizada nas bases *Web of Science* e *Scopus*, a RSL encontrou-se 83 artigos que foram selecionados, dos quais apenas 42 abordavam aspectos relacionados a métricas de projetos tradicionais, 10 sobre as métricas ágeis de fluxo, 7 sobre as métricas ágeis de *backlog* do produto e 24 sobre as métricas ágeis de produto.

Apesar do histórico da adoção dos projetos tradicionais existir há décadas, o total dos artigos sobre as métricas em projetos tradicionais (40) encontradas nesse estudo não apresentaram um número tão maior que os resultados das métricas ágeis. Nas métricas ágeis, temos 43 artigos identificadas nesse estudo. Assim, percebeu-se que as métricas tradicionais focaram em apresentar a métrica do EVM com a adoção de melhorias como melhorar a previsibilidade de riscos e de tomada de decisões para alcançar o melhor valor agregado e o sucesso do projeto ou adaptações inspiradas no EVM como o EDM. Outra métrica muito citada é a utilização do SPI buscando medir o desempenho do cronograma (prazo).

Nas métricas ágeis, percebeu-se que as métricas se concentraram no *discovery* e no *delivery*, ou seja, em relação a medir o *backlog* do produto (9 artigos) no *discovery* e no fluxo e produto, tivemos 34 artigos no *delivery*. As métricas de *backlog* do produto se concentraram em medir a qualidade do requisito, enquanto as de produto, verificou-se uma grande concentração de métricas com preocupações com a qualidade do produto, ou seja, no tocante

a quantidade de *bugs* resolvidos, tempo de resolução de problemas, qualidade do código-fonte e uma menor parcela sobre a satisfação do cliente. Sobre as métricas de fluxo, identificou-se métricas como o *Lead Time*, *Velocity*, *Efficiency Flow*, *Cycle Time*, *Sprint Burn Down*, *Epic Burn Down*, *Control Chart*, entre outras.

Ainda existe uma lacuna de processos de gestão ágil de projetos que contribuam como um *framework* capaz de ser um instrumento gerencial com métricas estratégicas para a organização e para o desenvolvimento ágil de produtos. Assim, projetos ágeis precisam testar hipóteses para resolver problemas organizacionais ou de clientes e precisam ser medidas para a tomada de decisão do que fazer a seguir.

Espera-se que este estudo possa contribuir com outras pesquisas que desejam identificar as métricas utilizadas entre as abordagens ágil e tradicional de gestão de projetos e orientar novas pesquisas para trabalhos futuros no sentido de analisar uma possível lacuna entre na utilização de métricas tradicionais em projetos ágeis, que se entende também ser uma limitação dessa pesquisa.

REFERÊNCIAS

- Abdelrahman Aljemabi, M., Wang, Z., & Saleh, M. A. (2020). Mining social collaboration patterns in developer social networks. *IET Software*, 14(7), 839-849.
- Ahmed, A. R., Tayyab, M., Bhatti, S. N., Alzahrani, A. J., & Babar, M. I. (2017). Impact of story point estimation on product using metrics in scrum development process. *International Journal of Advanced Computer Science and Applications*, 8(4).
- Aljedaibi, W., & Khamis, S. (2019). Towards Measuring the Project Management Process During Large Scale Software System Implementation Phase. *The ISC International Journal of Information Security*, 11(3), 161-172.
- Antinyan, V. (2014). Monitoring evolution of code complexity and magnitude of changes. *Acta Cybernetica*, 21(3), 367-382.
- Asha, N., & Mani, P. (2018). Knowledge-based Acceptance Test driven agile Approach for Quality Software Development. *International Journal of Resent Technology and Engineering*, 7.
- Athanasidou, D., Nugroho, A., Visser, J., & Zaidman, A. (2014). Test code quality and its relation to issue handling performance. *IEEE Transactions on Software Engineering*, 40(11), 1100-1125.
- Azzeh, M., & Nassif, A. B. (2016). A hybrid model for estimating software project effort from Use Case Points. *Applied Soft Computing*, 49, 981-989.
- Bakhtiary, V., Gandomani, T. J., & Salajegheh, A. (2020). The effectiveness of test-driven development approach on software projects: A multi-case study. *Bulletin of Electrical Engineering and Informatics*, 9(5), 2030-2037.
- Batselier, J., & Vanhoucke, M. (2015). Evaluation of deterministic state-of-the-art forecasting approaches for project duration based on earned value management. *International Journal of Project Management*, 33(7), 1588-1596.
- Budacu, E. N., & Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, 22(4), 70-79.

- Chang, C. P. (2015). Software risk modeling by clustering project metrics. *International Journal of Software Engineering and Knowledge Engineering*, 25(06), 1053-1076.
- Chen, H. L. (2018). Early prediction of project duration: A longitudinal study. *Engineering Management Journal*, 30(3), 191-202.
- Chen, H. L. (2021). Early identification of distressed capital projects: A longitudinal approach. *International Journal of Managing Projects in Business*.
- Chen, H. L. (2014). Improving forecasting accuracy of project earned value metrics: Linear modeling approach. *Journal of Management in Engineering*, 30(2), 135-145.
- Chen, H. L., Chen, W. T., & Lin, Y. L. (2016). Earned value project management: Improving the predictive power of planned value. *International Journal of Project Management*, 34(1), 22-29.
- Chen, Z., Demeulemeester, E., Bai, S., & Guo, Y. (2020). A Bayesian approach to set the tolerance limits for a statistical project control method. *International Journal of Production Research*, 58(10), 3150-3163.
- Choraś, M., Springer, T., Kozik, R., López, L., Martínez-Fernández, S., Ram, P., ... & Franch, X. (2020). Measuring and improving agile processes in a small-size software development company. *IEEE access*, 8, 78452-78466.
- Costa, J., Vasconcelos, A., & Fragoso, B. (2020). An Enterprise Architecture Approach for Assessing the Alignment Between Projects and Goals. *International Journal of Information Technology Project Management (IJITPM)*, 11(3), 55-76.
- Colin, J., Martens, A., Vanhoucke, M., & Wauters, M. (2015). A multivariate approach for top-down project control using earned value management. *Decision Support Systems*, 79, 65-76.
- Colin, J., & Vanhoucke, M. (2014). Setting tolerance limits for statistical project control using earned value management. *Omega*, 49, 107-122.
- Colin, J., & Vanhoucke, M. (2015). A comparison of the performance of various project control methods using earned value management systems. *Expert Systems with Applications*, 42(6), 3159-3175.
- Concas, G., Marchesi, M., Destefanis, G., & Tonelli, R. (2012). An empirical study of software metrics for assessing the phases of an agile project. *International Journal of Software Engineering and Knowledge Engineering*, 22(04), 525-548.
- Day, F. C., Pourhomayoun, M., Keeves, D., Lees, A. F., Sarrafzadeh, M., Bell, D., & Pfeffer, M. A. (2019). Feasibility study of an EHR-integrated mobile shared decision making application. *International journal of medical informatics*, 124, 24-30.
- Dennehy, D., & Conboy, K. (2018). Identifying challenges and a research agenda for flow in software project management. *Project Management Journal*, 49(6), 103-118.
- El Asmar, M., Hanna, A. S., & Loh, W. Y. (2013). Quantifying performance for the integrated project delivery system as compared to established delivery systems. *Journal of construction engineering and management*, 139(11), 04013012.
- Fleche, D., Bluntzer, J. B., Al Khatib, A., Mahdjoub, M., & Sagot, J. C. (2017). Collaborative project: Evolution of computer-aided design data completeness as management information. *Concurrent Engineering*, 25(3), 212-228.
- Gao, K., Khoshgoftaar, T. M., Wang, H., & Seliya, N. (2011). Choosing software metrics for defect prediction: an investigation on feature selection techniques. *Software: Practice and Experience*, 41(5), 579-606.
- Geng, S., Chuah, K. B., Law, K. M., Cheung, C. K., Chau, Y. C., & Rui, C. (2018). Knowledge contribution as a factor in project selection. *Project Management Journal*, 49(1), 25-41.
- Glenwright, E. (2007). A survey of the 30 most serious flaws in scheduling. In *PMICoS 2007 Annual Conference*.

- Grau, D., & Back, W. E. (2015). Predictability index: Novel metric to assess cost and schedule performance. *Journal of Construction Engineering and Management*, 141(12), 04015043.
- Gren, L., Goldman, A., & Jacobsson, C. (2020). Agile ways of working: a team maturity perspective. *Journal of Software: Evolution and Process*, 32(6), e2244.
- Grimaldi, P., Perrotta, L., Corvello, V., & Verteramo, S. (2016). An agile, measurable and scalable approach to deliver software applications in a large enterprise. *International Journal of Agile Systems and Management*, 9(4), 326-339.
- Han, S., Choi, J. O., & O'Connor, J. T. (2017). Quality of baseline schedules: Lessons from higher education capital facility projects. *Journal of Professional Issues in Engineering Education and Practice*, 143(1), 04016017.
- Hanna, A. S. (2016). Benchmark performance metrics for integrated project delivery. *Journal of Construction Engineering and Management*, 142(9), 04016040.
- Hanna, A. S., Tadt, E. J., & Whited, G. C. (2012). Request for information: benchmarks and metrics for major highway projects. *Journal of construction engineering and management*, 138(12), 1347-1352.
- Hazır, Ö. (2015). A review of analytical models, approaches and decision support tools in project monitoring and control. *International Journal of Project Management*, 33(4), 808-815.
- Hernández, G., Martínez, Á., Jiménez, R., & Jiménez, F. (2019). Productivity Metrics for an Agile Software Development Team: A Systematic Review. *TecnoLógicas*, 22(SPE), 63-81.
- Ibba, S., Pinna, A., Lunesu, M. I., Marchesi, M., & Tonelli, R. (2018). Initial coin offerings and agile practices. *Future Internet*, 10(11), 103.
- Ibrahim, M. W., Hanna, A., & Kievet, D. (2020). Quantitative comparison of project performance between project delivery systems. *Journal of management in engineering*, 36(6), 04020082.
- Iqbal, S., Naeem, M., & Khan, A. (2012). Yet another set of requirement metrics for software projects.
- Isong, B., & Obeten, E. (2013). A systematic review of the empirical validation of object-oriented metrics towards fault-proneness prediction. *International Journal of Software Engineering and Knowledge Engineering*, 23(10), 1513-1540.
- Jethani, K. (2013). Software metrics for effective project management. *International Journal of System Assurance Engineering and Management*, 4(4), 335-340.
- Junior, V. B., & Junior, S. D. (2019). *Elaboração de um Plano de Gerenciamento de Projeto para Implantação de Laboratório de Metrologia Utilizando o Guia PMBOK® 5º Edição*. *Cippus*, 7(2), 127-145.
- Jyothi, V. E., & Rao, K. N. (2017). Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Engineering Research and Science*, 4(3), 237085.
- Kamulegeya, G., Mugwanya, R., & Hebig, R. (2018). Measurements in the early stage software start-ups: a multiple case study in a nascent ecosystem. *Foundations of Computing and Decision Sciences*, 43(4), 251-280.
- Kayes, I., Sarker, M., & Chakareski, J. (2016). Product backlog rating: a case study on measuring test quality in scrum. *Innovations in Systems and Software Engineering*, 12(4), 303-317.
- Kerkhove, L. P., & Vanhoucke, M. (2017). Extensions of earned value management: Using the earned incentive metric to improve signal quality. *International Journal of Project Management*, 35(2), 148-168.
- Kristiansen, J. N., & Ritala, P. (2018). Measuring radical innovation project success: typical metrics don't work. *Journal of Business Strategy*.

- Kim, T., Kim, Y. W., & Cho, H. (2016). Customer earned value: performance indicator from flow and value generation view. *Journal of management in engineering*, 32(1), 04015017.
- Ko, B. S., & Cho, M. S. (2020). Evaluating the relative efficiency of defense R&D projects in a multi-project environment using EVM and CAIV measures. *ICIC Express Letters, Part B: Applications*, 11(7), 675-682.
- Yamashita, A., & Counsell, S. (2013). Code smells as system-level indicators of maintainability: An empirical study. *Journal of Systems and Software*, 86(10), 2639-2653.
- Yousefi, N., Sobhani, A., Naeni, L. M., & Currie, K. R. (2019). Using statistical control charts to monitor duration-based performance of project. *arXiv preprint arXiv:1902.02270*.
- Mas, A., Mesquida, A. L., & Pacheco, M. (2020). Supporting the deployment of ISO-based project management processes with agile metrics. *Computer Standards & Interfaces*, 70, 103405.
- Mascarenhas Hornos da Costa, J., Oehmen, J., Rebentisch, E., & Nightingale, D. (2014). Toward a better comprehension of Lean metrics for research and product development management. *R&D Management*, 44(4), 370-383.
- Mäkiahho, P., Vartiainen, K., & Poranen, T. (2022). MMT: a tool for observing metrics in software projects. In *Research Anthology on Agile Software, Software Development, and Testing* (pp. 1077-1089). IGI Global.
- Meidan, A., García-García, J. A., Ramos, I., & Escalona, M. J. (2018). Measuring software process: a systematic mapping study. *ACM Computing Surveys (CSUR)*, 51(3), 1-32.
- Narbaev, T., & De Marco, A. (2014). Combination of growth model and earned schedule to forecast project cost at completion. *Journal of Construction engineering and management*, 140(1), 04013038.
- Nidagundi, P., & Novickis, L. (2016). Introduction to Lean Canvas Transformation Models and Metrics in Software Testing. *Appl. Comput. Syst.*, 19(1), 30.
- Orgut, R. E., Zhu, J., Batouli, M., Mostafavi, A., & Jaselskis, E. J. (2018). Metrics that matter: Core predictive and diagnostic metrics for improved project controls and analytics. *Journal of Construction Engineering and Management*, 144(11), 04018100.
- Pajares, J., & Lopez-Paredes, A. (2011). An extension of the EVM analysis for project monitoring: The Cost Control Index and the Schedule Control Index. *International Journal of Project Management*, 29(5), 615-621.
- Parrend, P., Masai, P., Zanni-Merk, C., & Collet, P. (2014, May). Swarm Projects: Beyond the Metaphor. In *International Conference on Swarm Intelligence Based Optimization* (pp. 131-138). Springer, Cham.
- Pavlova, N., Onyshchenko, S., Obronova, A., Chebanova, T., & Andriievskaya, V. (2021). Creating the Agile Model to Manage the Activities of Project Oriented Transport Companies. *Eastern-European Journal of Enterprise Technologies*, 1(3), 109.
- Perkusich, M., Gorgônio, K. C., Almeida, H., & Perkusich, A. (2017). Assisting the continuous improvement of Scrum projects using metrics and Bayesian networks. *Journal of Software: Evolution and Process*, 29(6), e1835.
- Petersen, K., & Wohlin, C. (2011). Measuring the flow in lean software development. *Software: Practice and experience*, 41(9), 975-996.
- Pradhan, S., & Nanniyur, V. (2021). Large scale quality transformation in hybrid development organizations—A case study. *Journal of Systems and Software*, 171, 110836.
- Prakash B, Viswanathan V. (2018). Ontology based risks management model for agile software development. *Journal of Advanced Research in Dynamical and Control Systems*, 10(11 Special Issue):319–330.
- Pressman, R. S., & Maxim, B. R. (2021). *Engenharia de software-9*. McGraw Hill Brasil.

- Project Management Institute (PMI) (2021): A Guide to the Project Management Body of Knowledge (PMBOK Guide), PMI, Newtown Square, PA, 2021.
- Pozzana, I., Ellinas, C., Kalogridis, G., & Sakellariou, K. (2021). Spreading of performance fluctuations on real-world project networks. *Applied Network Science*, 6(1), 1-15.
- Sadler, H. (2020). ER2C SDMLC: enterprise release risk-centric systems development and maintenance life cycle. *Software Quality Journal*, 28(4), 1755-1787.
- Sajedi-Badashian, A., & Stroulia, E. (2020). Investigating the information value of different sources of evidence of developers' expertise for bug assignment in open-source projects. *IET Software*, 14(7), 748-758.
- Savola, R., Frühwirth, C., & Pietikäinen, A. (2012). Risk-driven security metrics in agile software development: An industrial pilot study. *Journal of Universal Computer Science*, 18(12), 1679-1702.
- Shawky, D. M., & Abd-El-Hafiz, S. K. (2016). Characterizing software development method using metrics. *Journal of Software: Evolution and Process*, 28(2), 82-96.
- Singh, J., Singh, K., & Singh, J. (2019). Reengineering framework to enhance the performance of existing software. *System*, 1139, 120-3.
- Simpson, J. R., Lin, C. T., Sieja, A., Sillau, S. H., & Pell, J. (2021). Optimizing the electronic health record: An inpatient sprint addresses provider burnout and improves electronic health record satisfaction. *Journal of the American Medical Informatics Association*, 28(3), 628-631.
- Rajablu, M., Hamdi, S., Marthandan, G., & Yusoff, W. F. W. (2017). Managing for stakeholders: Introducing stakeholder metrics-integrated model to lead project ethics and success. *International Journal of Project Organisation and Management*, 9(1), 31-56.
- Rajagopalan, J., & Srivastava, P. K. (2018). Introduction of a new metric "Project Health Index"(PHI) to successfully manage IT projects. *Journal of Organizational Change Management*.
- Ries, E. (2011). *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.
- Royce, W. (1970). The software lifecycle model (waterfall model). In *Proc. Westcon* (Vol. 314).
- Sampaio, R. F., & Mancini, M. C. (2007). Systematic review studies: a guide for careful synthesis of the scientific evidence. *Brazilian Journal of Physical Therapy*, 11, 83-89.
- Sato, T., & Hirao, M. (2013). Optimum budget allocation method for projects with critical risks. *International Journal of Project Management*, 31(1), 126-135.
- Sharma, V., Caldas, C. H., & Mulva, S. P. (2021). Development of metrics and an external benchmarking program for healthcare facilities. *International Journal of Construction Management*, 21(6), 615-630.
- Staron, M., & Meding, W. (2016). Mesram—a method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation. *Journal of Systems and Software*, 113, 76-100.
- Spijkman, T., Molenaar, S., Dalpiaz, F., & Brinkkemper, S. (2021). Alignment and granularity of requirements and architecture in agile development: A functional perspective. *Information and Software Technology*, 133, 106535.
- Sommerville, I. (2007). *Engenharia de Software*, 9. Edição. Pearson, Addison Wesley, 8(9), 10, p. 44-45.
- Song, J., Martens, A., & Vanhoucke, M. (2021). Using schedule risk analysis with resource constraints for project control. *European Journal of Operational Research*, 288(3), 736-752.
- Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante, 2019.

- Tahir, T., Rasool, G., & Noman, M. (2018). A systematic mapping study on software measurement programs in SMEs. *E-Informatica Software Engineering Journal*, 12(1).
- Tanveer, B., Vollmer, A. M., Braun, S., & Ali, N. B. (2019). An evaluation of effort estimation supported by change impact analysis in agile software development. *Journal of Software: Evolution and Process*, 31(5), e2165.
- Tranfield, D., Denyer, D., & Smart, P. (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, 14(3), 207-222.
- Wallace, L. G., & Sheetz, S. D. (2014). The adoption of software measures: A technology acceptance model (TAM) perspective. *Information & Management*, 51(2), 249-259.
- Wauters, M., & Vanhoucke, M. (2015). Study of the stability of earned value management forecasting. *Journal of Construction Engineering and Management*, 141(4), 04014086.
- Van Casteren, W. (2017). The Waterfall Model and the Agile Methodologies: A comparison by project characteristics. *Research Gate*, 1-6.
- Vanhoucke, M. (2011). On the dynamic use of project performance and schedule risk information during project tracking. *Omega*, 39(4), 416-426.
- Vanhoucke, M. (2019). Tolerance limits for project control: An overview of different approaches. *Computers & Industrial Engineering*, 127, 467-479.
- Üsfekes, Ç., Tüzün, E., Yılmaz, M., Macit, Y., & Clarke, P. (2019). Auction-based serious game for bug tracking. *IET Software*, 13(5), 386-392.
- Zheng, M., Zada, I., Shahzad, S., Iqbal, J., Shafiq, M., Zeeshan, M., & Ali, A. (2021). Key performance indicators for the integration of the service-oriented architecture and scrum process model for IOT. *Scientific Programming*, 2021.

4 ESTUDO 2: MAPEAMENTO DE MÉTRICAS PARA DESEMPENHO DE PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA²

RESUMO

Em um contexto competitivo onde as organizações precisam entregar produtos ou serviços por meio da gestão ágil de projetos, um processo em que as métricas se tornam insumos importantes para o controle e monitoramento de projetos. O que se busca nesse artigo é compreender se existem organizações que, apesar de conduzirem projetos ágeis, utilizam controle os projetos por meio do uso de métricas tradicionais. Além disso, se busca entender se a utilização de métricas de projetos tradicionais dificultam a visão e o trabalho da liderança corporativa para tomar decisões em projetos ágeis. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis de *software*. Como estratégia metodológica foi adotado uma Revisão Sistemática de Literatura para auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento. Para a coleta de dados foram utilizadas as bases *Web of Science* e *Scopus*. Foram encontrados 83 artigos e os resultados apontaram que os projetos tradicionais seguem controlando os projetos com métricas tradicionais, tal como *Earned Value Management* (EVM). Em projetos ágeis, as métricas mais comuns são métricas associadas *backlog* do produto, *delivery* e qualidade do produto. Este estudo contribui com outras pesquisas que desejarem identificar as métricas utilizadas entre as diferentes abordagens de gestão de projetos e orientar novas pesquisas para trabalhos futuros.

Palavras-chave: Desenvolvimento de Software; Métricas Ágeis e Tradicionais; Scrum; Kanban.

²Estudo publicado na *Journal on Innovation and Sustainability*, v. 14, n. 4 (2023), disponível no Apêndice e em <https://revistas.pucsp.br/index.php/risus/article/view/63363/44256>.

1 INTRODUÇÃO

O Modelo em Cascata (MC) foi criado com o objetivo de representar a execução de fases de um projeto de desenvolvimento de *software* (Royce, 1970). Nesse modelo, cada fase possui dependência com a fase anterior, onde a próxima fase somente poderá ser iniciada quando a fase antecedente for concluída (Sommerville, 2007). O MC é composto pelas fases de Levantamento de Requisitos, Análise de Requisitos, Projeto, Desenvolvimento, Testes e Implantação (Pressman, 2021). Dentre elas, as fases de Levantamento de Requisitos, Análise de Requisitos e Projeto podem exigir grandes esforços durante o planejamento e concepção sobre o escopo que será desenvolvido no projeto e esperado pelo cliente (Van Casteren, 2017).

No contexto das práticas gerenciais, os projetos orientados ao MC sob as práticas tradicionais de gestão de projetos possuem incertezas no início do projeto que podem mudar o escopo e o direcionamento do *software* durante o projeto (Sutherland, 2019).

Em se tratando das práticas gerenciais e o desempenho de projetos, o uso de métricas tornou-se um fator estratégico para que o controle e o monitoramento do projeto se tornem eficazes (Glenwright, 2007). No tocante às práticas tradicionais, as métricas mais utilizadas pelas organizações é o gerenciamento do valor agregado (Junior e Junior, 2019). O gerenciamento do valor agregado ou *Earned Value Management* (EVM), foi proposto pelo *Project Management Institute* (PMI, 2021). O EVM é uma técnica de gerenciamento de projetos para medir o desempenho e o progresso de projetos combinando as medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido (Glenwright, 2007; PMI, 2021). Porém, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as práticas ágeis em gerenciamento de projetos de *software*.

Em relação ao prazo do projeto, as práticas ágeis preconizam que os prazos das atividades devem ser baseados em dados históricos de entrega da equipe de projeto, em vez de estimativas baseadas no planejamento (Sutherland, 2019). No caso do MC, os prazos são baseados nas estimativas definidas na fase de planejamento e dando origem ao cronograma de atividades (Sutherland, 2019), distorcendo assim à utilização das métricas em ambas as práticas. Conforme Choudhury (2019) e Dias e Larieira (2021), essa distorção na utilização das métricas de projetos tradicionais nos projetos ágeis se justifica devido a utilização das abordagens do triângulo de ferro (escopo, prazo e custo) em projetos tradicionais e o triângulo invertido em projetos ágeis. Choudhury (2019) e Dias e Larieira (2021) adicionaram que o triângulo de ferro aborda o escopo fixo com o prazo e o custo variáveis, enquanto no ágil o triângulo invertido trata do prazo e custo fixo com o escopo variável.

No contexto das práticas ágeis, as métricas são orientadas ao valor entregue ao cliente (Gren, Goldman & Jacobsson, 2020). De acordo com Jyothi e Rao (2017) e Budacu e Pocatilu (2018), as métricas ágeis mais utilizadas são (i) *velocity*; (ii) *Work Item Age* (WIA); (iii) *Throughput*, (iv) *Cycle Time* (CT) e (v) *Lead Time* (LT). O *velocity* é uma métrica utilizada no desempenho dos desenvolvedores por meio dos pontos de história do usuário. O WIA é uma métrica para medir o tempo em dias que um requisito está em uma determinada fase do desenvolvimento. Já o *Throughput* é uma métrica para avaliar a capacidade de entrega da equipe de projeto ao final de cada iteração. O CT conta o tempo em que um requisito está em um determinado ciclo ou etapa de desenvolvimento, enquanto o LT é considerado uma medida que avalia o tempo em que um requisito é iniciado em desenvolvimento até ser entregue ao cliente (Budacu & Pocatilu, 2018).

Já as práticas tradicionais utilizam o EVM para medir o desempenho e o progresso de projetos, utilizando métricas baseadas em medidas comparativas entre o esforço planejado em relação ao esforço realizado (PMI, 2021). Glenwright (2007) apresenta as métricas mais utilizadas pelas organizações sob as práticas tradicionais de gerenciamento de projetos são (i) *Schedule Performance Index* (SPI) e (ii) *Cost Performance Index* (CPI). O SPI analisa o progresso real ou o valor agregado do trabalho realizado com o progresso do trabalho planejado até o momento da medição. Já o CPI é uma medida da eficiência de custos realizados com o progresso do custo planejado até o momento da medição. Assim, as métricas possuem relações distintas de acompanhamento durante o ciclo de vida dos projetos.

Em relação às entregas do projeto, nas práticas tradicionais, o produto gerado é entregue somente no final do projeto ou entregue em fases dentro de uma estratégia de cronograma em fases (PMI, 2021). Já nas práticas ágeis o produto é entregue em até quatro semanas (*Sprint*) na forma de incrementos funcionais. A soma dos incrementos funcionais pode viabilizar os lançamentos do *Minimum Viable Product* (MVP) (Ries, 2011). O MVP é caracterizado por ser uma versão funcional e sem erros críticos de um *software* para possível liberação aos clientes para receber o *feedback* mais rápido possível (Sutherland, 2019).

Ao se observar a utilização das métricas para análise do desempenho de projetos de desenvolvimento de *software*, é possível notar divergências entre as práticas ágeis e tradicionais de gestão de projetos. Essa divergência consiste em um cenário em que organizações que conduzem projetos sob as práticas tradicionais decidem implementar as práticas ágeis de gestão de projetos. Porém, a liderança permanece utilizando métricas tradicionais baseadas nas abordagens de custo, prazo e escopo diferentes entre as práticas, podendo causar impactos no desempenho dos projetos ágeis.

O problema que tange e justifica nessa pesquisa é que muitas organizações que conduzem projetos, implementaram o modelo ágil de desenvolvimento de *software* em equipes isoladas e com as métricas ágeis comumente na operação (Jyothi & Rao 2017; dos Santos *et al.* 2018). Além disso, a liderança realiza a medição do desempenho do projeto por meio de métricas tradicionais, o que podem causar impactos no gerenciamento de projetos ágeis, dado que a gestão tradicional e a ágil de projetos operam com abordagens de custo, prazo e escopo de maneiras diferentes (Choudhury, 2019; Dias & Larieira, 2021)

Com base no apresentado sobre a divergência da utilização de métricas para o controle de projetos tradicionais e ágeis, abre-se uma lacuna para a realização desta pesquisa, procurando responder a seguinte questão: **“Quais são as métricas utilizadas para medir o desempenho de projetos nas práticas tradicionais e ágeis?”**. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis de *software*. Para responder tal indagação, foi adotado como estratégia metodológica uma Revisão Sistemática de Literatura (RSL) verificando as métricas tradicionais e ágeis utilizadas no desempenho de projetos. A justificativa para tal escolha se dá pelo papel da RSL que auxilia no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento (Tranfield, Denyer & Smart, 2003).

Na seção seguinte serão apresentados os procedimentos metodológicos. Posteriormente são apresentados os resultados e, por fim, serão apresentadas as conclusões, discutidas as limitações desta pesquisa e indicações para estudos futuros.

2. MATERIAIS E MÉTODOS

Esta pesquisa adotou uma RSL como método a fim de compreender a convergência de duas temáticas relevantes no contexto de gestão de projetos, as métricas, tanto nas práticas tradicionais como nas práticas ágeis. A RSL se difere das tradicionais revisões narrativas por adotar um processo científico sistemático que é replicável e transparente (Tranfield, Denyer & Smart, 2003). Nesse sentido, justifica-se também a sua utilização por uma RSL minimizar o viés na construção de um *corpus* teórico, bem como a possibilidade de construir uma trilha de auditoria das decisões e procedimentos aplicados (Cook, Mulrow & Haynes, 1997).

Os procedimentos para realização desta RSL seguiram seis fases prescritas por Pollock e Berge (2018): (i) esclarecer metas e objetivos de pesquisa; (ii) buscar pesquisas relevantes; (iii) coletar dados; (iv) avaliar a qualidade dos estudos; (v) sintetizar as evidências; (vi) interpretar os achados.

A primeira etapa foi motivada pela questão que norteia esta pesquisa, que foi “quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis?” Assim, foram utilizadas como fontes de pesquisa as bases de dados acadêmicas *Web of Science* e *Scopus*, por serem umas das principais bases de dados para acesso as pesquisas publicadas na área de ciências sociais.

A *string* utilizada para realização das buscas foi ("*metric**" and ("*project managem**") or "*metric**" and ("*agil**" or "*SCRUM*" or "*kanban*" or "*lean**" or "*Devops*" or "*NPS*" or "*Net promoter score*" or "*EBM*" or "*Evidence Based Management*" or "*OKR*" or "*Objective Key Result**"). A pesquisa foi realizada em 01 de maio de 2022. A utilização dos operadores booleanos “and” e “or”, além do uso do símbolo “*” permitem uma maior abrangência e controle na construção da base de pesquisa. Os operadores são aplicados levando em consideração a intersecção das duas áreas de pesquisa estudadas. O uso do asterisco incorpora todas as variações da palavra na posição posterior a que ela se encontra. Destaca-se que não foi aplicado filtro temporal para que fosse possível mapear toda a produção sobre as temáticas estudadas. Portanto, após a primeira fase que compreende o estabelecimento dos objetivos, os pesquisadores passaram para a aplicação da *string* de busca. Os resultados encontrados na primeira rodada passaram pela análise e triagem, apresentado na Figura 1, respeitando o proposto por Pollock e Berge (2018), que apresentam quatro etapas para a elaboração do *corpus* de análise. A Figura 1 apresenta a primeira etapa destaca a *string* utilizada para a pesquisa nas bases de dados. A segunda etapa é a triagem dos resultados com a finalidade de balizar os artigos correspondentes à proposta de pesquisa. Já na terceira etapa foram aplicados os critérios de elegibilidade, onde os artigos foram avaliados segundo os critérios de inclusão e exclusão. A quarta etapa constituiu o *corpus* de análise, resultando na base de dados amostral de artigos para a pesquisa.

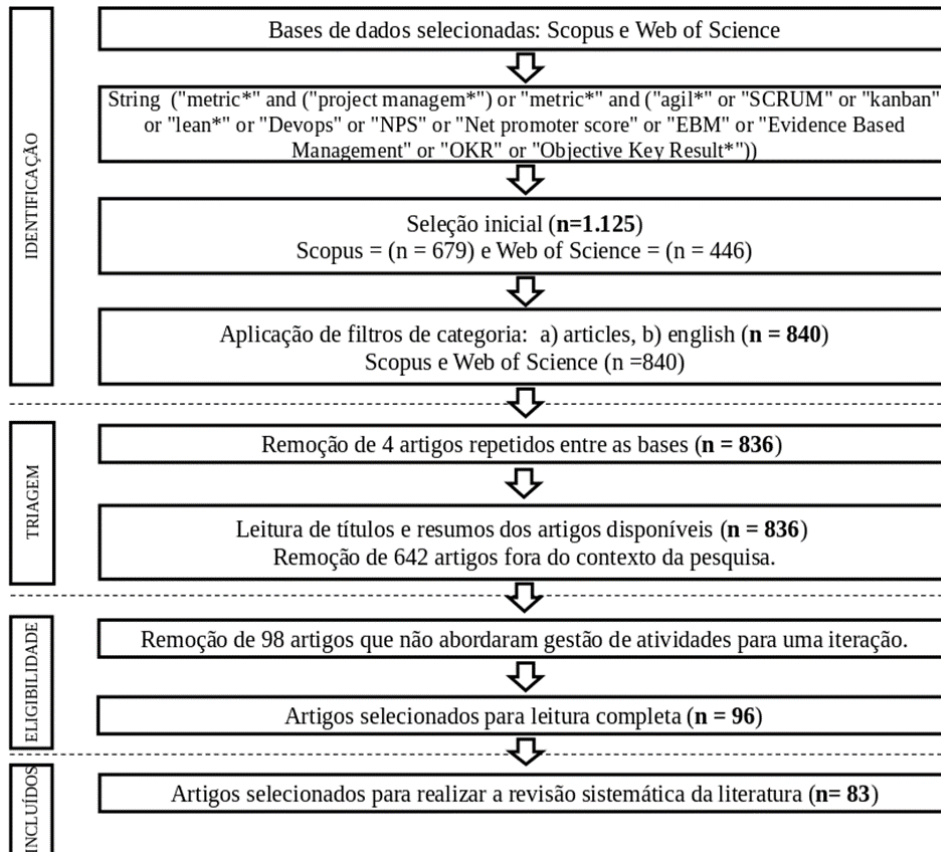


Figura 1 - Etapas da RSL

Fonte: adaptado de Pollock e Berge (2018).

Para a composição da base de dados, filtros foram aplicados durante a pesquisa inicial, como o critério de somente artigos em periódicos, excluindo da base os artigos em congresso, livros, entre outros. Essa exclusão foi definida, devido esses estudos não passarem por uma revisão por pares em alguns casos, além de mitigar a redundância de obras apresentadas em congressos e publicadas em periódicos. As áreas delimitadas para esta pesquisa foram: “*Management*” e “*Business*” para *Web of Science*, “*Business*” para a *Scopus*. Com a consolidação das bases de dados, foram removidos os artigos repetidos com o objetivo de eliminar a redundância. Na sequência, a base de dados foi tratada com auxílio de planilhas eletrônicas do *LibreOffice*. O *LibreOffice* permitiu realizar as análises dos dados e apresentar os resultados a partir da combinação de informações quantitativas pela análise de frequência, bem como informações qualitativas pela categorização dos conteúdos dos artigos. Esta fase da pesquisa permitiu também apresentar uma análise descritiva relevante. A partir da leitura dos títulos e resumos na fase de triagem para verificar a elegibilidade dos artigos foram estabelecidos alguns critérios de inclusão e exclusão, destacados na Tabela 1. Após verificação e leitura dos resumos e introduções dos 840 artigos selecionados previamente, a base final foi composta por 83 artigos aceitos que formaram o *corpus* de pesquisa.

Tabela 1 - Critérios de Inclusão e Exclusão

Critérios de Inclusão	Razão para Inclusão
Artigos que conceituam os construtos estudados	Permitir abordar finalidades da pesquisa: compreender os construtos estudados conforme as obras.
Artigos que abordam as relações entre os construtos estudados	Permitir contemplar alinhamento das obras: compreender as interdependências e relações entre os construtos.
Artigos publicados	Oferecer maior rigor nos argumentos e contribuições teóricas estudadas.
Critérios de Exclusão	Razão para Exclusão
Artigos com foco em Qualidade, Marketing, Saúde, Finanças ou outros fins fora dos construtos determinados.	Excluir artigos que não estejam focados nas questões que oferecerão <i>insights</i> para atender os objetivos da pesquisa.
Artigos sem fundamentação teórica relevante ou de baixa relação com os construtos.	Uma das finalidades do estudo é obter perspectivas futuras de pesquisa, por meio de conhecimento teórico existente em uma estrutura, para a qual os pressupostos teóricos são pré-requisitos.

Fonte: elaborado pelos autores, 2021.

O próximo passo foi de realizar a leitura dos 83 artigos contidos no *corpus* de análise, categorizando os conteúdos em planilhas do *LibreOffice* com o objetivo de apresentar um agrupamento dos achados e comparação das categorias. As atividades aplicadas nessa fase estão alinhadas com as prescrições de Pollock e Berge (2018) nas fases (v) sintetizar as evidências e (vi) interpretar os achados. Embora tenham sido aplicados alguns tratamentos quantitativos. Nessa pesquisa, priorizou-se a análise qualitativa dos artigos a fim de constituir uma matriz que pudesse representar os achados deste estudo.

3. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Nessa seção são apresentados os resultados após a realização dos processos de análise dos artigos e as categorias evidenciadas após a análise aprofundada dos conteúdos dos artigos.

3.1 Evolução da publicação dos artigos

Os artigos selecionados nas bases de dados *Web of Science e Scopus* foram triados e o *corpus* de análise foi constituído por 83 artigos. Os artigos passaram por uma análise mais aprofundada, o que permitiu compreender os estudos sobre a relação entre as métricas ágeis e as métricas tradicionais. Os 83 artigos analisados estão situados temporalmente entre os anos de 2011 até maio de 2021. De acordo com esta base de dados, a publicação de artigos que tratam da relação entre as métricas ágeis e métricas tradicionais ficou entre 4 e 5 artigos de 2011 até 2013. Entre os anos de 2014 até 2017 que as publicações subiram para 8 e 9 artigos publicados. Somente em 2018 que 17 artigos foram publicados e retornando para 8 artigos em 2019, 13 em 2020 e 10 em 2021. Um aspecto relevante é que no ano de 2018 teve o número maior de publicações e correspondeu com 18% de publicações nos últimos dez

anos ou de 2011 a 2021. Após a análise da distribuição temporal dos artigos, também se buscou compreender a frequência de artigos por periódico. A Tabela 2 demonstra os primeiros dez periódicos e a respectiva quantidade de artigos publicados, ordenados pelo maior número de publicações e pela ordem alfabética por periódicos.

Tabela 2 - Periódicos e número de publicações no tema

#	Periódicos	Publicações
1	IEEE Access	15
2	Journal Of Systems And Software	14
3	International Journal Of Project Management	10
4	International Journal Of Lean Six Sigma	10
5	Computer Networks	7
6	Computers & Industrial Engineering	7
7	Journal Of Modelling In Management	6
8	Journal Of Software-Evolution And Process	6
9	Journal Of Manufacturing Technology Management	6
10	International Journal Of Software Engineering And Knowledge Engineering	6

Fonte: dados da pesquisa, 2022.

Com relação ao número de artigos publicados por periódico, os periódicos considerados mais profícuos, de acordo com dados da pesquisa, são: ‘IEEE Access’ com 15 artigos; ‘*Journal Of Systems And Software*’ com 14 artigos e ‘*International Journal Of Project Management*’ com 10 artigos. Nota-se que a grande maioria dos periódicos configura com apenas 1 e 2 artigos publicados, sumarizando um total de 100 periódicos. Apesar do tema de gestão de projetos estar associado a Administração, as três revistas com mais publicações são associadas a engenharia e tecnologia (*IEEE Access* e *Journal Of Systems And Software*) e uma de gestão de projetos (*International Journal Of Project Management*). Na próxima seção serão discutidos artigos mapeados no *corpus* dessa pesquisa, a categorização que os artigos foram submetidos e a discussão dos achados dos artigos que compõem as categorias.

3.2 Análise e discussão das categorias encontradas

Após o mapeamento dos artigos que constituíram o *corpus* de análise, foi realizada uma análise aprofundada dos conteúdos publicados. A leitura e categorização dos conteúdos levou a classificação dos artigos em quatro categorias, apresentados na Tabela 3 através da Categoria, Quantidade e Autores.

Tabela 3 - Classificação dos artigos em 4 categorias

Categoria	Qtde	Autores
Métricas de Projetos Tradicionais	40	Vanhoucke (2011); Chen (2021); Aljedaibi e Khamis (2019); Colin <i>et al.</i> (2015); Rajagopalan e Srivastava (2018); Chen, Chen e Lin (2016); Fleche <i>et al.</i> (2017); Sharma, Caldas e Mulva (2021); Batselier e Vanhoucke (2015); Chen (2018); Pajares e Lopez-Paredes (2011); Vanhoucke (2019); Colin e Vanhoucke (2014); Song, Marten e Vanhoucke (2021); Kerkhove e Vanhoucke (2017); Geng <i>et al.</i> (2018); Sato e Hirao (2013); Staron e Meding (2016); Spijkman <i>et al.</i> (2021); Ibrahim, Hanna e Kievet (2020); Chen <i>et al.</i> (2020); Ko e Cho (2020); Yousefi <i>et al.</i> (2019); Orgut <i>et al.</i> (2018); Kristiansen e Ritala (2018); Rajablu <i>et al.</i> (2017); Han, Choi e O'Connor (2017); Hanna (2016); Kim, Kim e Cho (2016); Hazır, (2015); Wauters e Vanhoucke (2015); Grau e Back (2015); Chen, H. L. (2014); Narbaev e De Marco (2014); Jethani (2013); El Asmar, Hanna e Loh (2013); Hanna, Tadt e Whited (2012); Iqbal, Naeem e Khan (2012); Pozzana <i>et al.</i> (2021); Colin e Vanhoucke (2015).
Métricas Ágeis de Fluxo	9	Grimaldi <i>et al.</i> (2016); Dennehy e Conboy (2018); Mas, Mesquida e Pacheco (2020); Ahmed <i>et al.</i> (2017); Sajedi-Badashian e Stroulia (2020); Petersen e Wohlin (2011); Choraś <i>et al.</i> (2020); Meidan <i>et al.</i> (2018); Sadler (2020).
Métricas Ágeis de Backlog do Produto	9	Azzeq e Nassif (2016); Wallace e Sheetz (2014); Parrend <i>et al.</i> (2014); Kayes, Sarker e Chakareski (2016); Zheng <i>et al.</i> (2021); Pavlova <i>et al.</i> (2021); Antinyan (2014); Prakash e Viswanathan (2018); Asha e Mani (2018).
Métricas Ágeis de Produto	25	Nidagundi e Novickis (2016); Mäkiäho, Vartiainen, e Poranen (2022); Chang (2015); Mascarenhas Hornos <i>et al.</i> (2014); Singh, Singh e Singh (2019); Simpson <i>et al.</i> (2021); Day <i>et al.</i> (2019); Üsfekes, <i>et al.</i> (2019); Abdelrahman Aljemabi, Wang e Saleh (2020); Costa, Vasconcelos, e Fragoso (2020); Ibba <i>et al.</i> (2018); Concas <i>et al.</i> (2012); Shawky e Abd-El-Hafiz (2016); Tahir, Rasool e Noman (2018); Isong e Obeten (2013); Tanveer <i>et al.</i> (2019); Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018); Yamashita e Counsell (2013); Bakhtiary, Gandomani e Salajegheh (2020); Hernández <i>et al.</i> (2020); Athanasiou <i>et al.</i> (2014); Gao <i>et al.</i> (2011); Perkusich <i>et al.</i> (2017); Savola, Frühwirth e Pietikäinen (2012).
Total	83	

Fonte: dados da pesquisa, 2022.

As quatro categorias encontradas compreendem uma abstração observada nas leituras dos artigos. Cabe explicitar que o processo de análise inicial dos artigos foi individual e, depois, para seleção das categorias passou-se pela análise do pesquisador desta pesquisa. O processo de categorização foi realizado pela discussão e estabelecimento de um consenso sobre qual era a maior aderência de todos os artigos em cada uma das categorias constituídas.

Na próxima seção, serão discutidas as categorias para explicitar os achados desta pesquisa e as categorias aqui apresentadas representam uma forma de explicação e organização dos conteúdos estudados.

3.3 Métricas de Projetos Tradicionais

Como visto, as práticas de gerenciamento de projetos tradicionais se baseiam, para medir o desempenho dos projetos, a partir de um conjunto de métricas orientados frente a um planejamento prévio. Desse modo, as métricas são caracterizadas, em sua essência, utilizando

as medidas de escopo, tempo e custos, mediante a um planejamento previamente estabelecido. Tais medidas são comparadas com a posição atual do projeto (trabalho realizado) em relação ao planejado até o momento da medição (trabalho realizado).

Frente ao uso das práticas tradicionais de gestão de projetos, Vanhoucke (2011) apresentou como métrica, no Montecarlo para simular o progresso do projeto sob a estrutura analítica do projeto (EAP), utilizando dados do gerenciamento de valor agregado (EVM) como possíveis sinais de alertas e acionadores para a necessidade de ações corretivas. Ainda sob o olhar da EAP, Colin e Vanhoucke (2015) apresentaram um modelo multivariado adotando como métrica o EVM/ES, resultando no *Earned Value Management/Earned Schedule*. A métrica é utilizada pelo gerente caso o projeto apresente algum risco de maneira preventiva. Kerkhove e Vanhoucke (2017) adotaram a técnica *Earned Incentive Metric*, empregando o EVM/ES com incentivos de custo e/ou tempo. Como resultado, os autores destacaram que *Earned Incentive Metric* apresentou em papel mais significativo que o EVM/ES em uma amostra de 4.200 projetos com tamanhos e estruturas de contrato diferentes ou parametrizando o tamanho de um projeto grande (Chen *et al.*, 2020).

Sobre a adoção do EVM como métrica, Chen, Chen e Lin (2016) concluíram que a melhoria no poder preditivo do *Planned Value* (PV), calculado pelo erro percentual médio absoluto (*Mean Absolute Percentage Error*), antes da execução do projeto fornece aos gestores informações preditivas confiáveis no desempenho do *Earned Value* (EV) e *Actual Cost Value* (ACV), permitindo uma ação proativa efetiva para garantir resultados de desempenho favoráveis. Chen (2021) e Hazir (2015) adotaram como métrica variação de desempenho na fase de planejamento que explica uma parte substancial da dificuldade do projeto na conclusão, mediante a um modelo de classificação de problemas de projetos, com o objetivo de identificar problemas nos projetos que provavelmente falharão antes de eles começarem. No mesmo sentido, Sharma, Caldas e Mulva (2021) utilizaram o EVM nas áreas de custo, cronograma, segurança, retrabalho e mudanças no projeto com o propósito de apoiar à tomada de decisão, facilitar o orçamento de capital, o impacto das condições de mercado e na melhoria contínua e melhores práticas de gestão.

Nessa mesma linha de métrica, Yousefi *et al.* (2019) apresentam o controle estatístico em índices do *Earned Duration Management* (EDM) para melhor investigar as variações do desempenho do cronograma do projeto ou para controlar projetos de inovação radical. Ainda em relação ao cronograma, Han *et al.* (2017) destacam o uso de métricas de qualidade de cronogramas baseados no limite de duração máxima (30 dias); Proporção de tarefas detalhadas para marcos; Porcentagem de tarefas no caminho crítico; Número de defasagens; Tipo de relacionamento (terminar para começar); e Marcos sem predecessor ou sucessor.

Wauters e Vanhoucke (2015) criaram um experimento computacional para avaliar a estabilidade usando a nova métrica de defasagem média proposta usando o *Coefficient of Variation* (CV) do EVM. Grau e Back (2015), no mesmo contexto, destacam que as métricas de controle do cronograma devem apoiar o gestor a ser proativo, em vez de reativo, no reconhecimento de tendências e eventos melhora significativamente a capacidade da equipe de minimizar a diferença entre os resultados previstos e reais. Chen (2014) combinou séries temporais e análise de regressão para desenvolver PV em um modelo de resposta EV para prever *Work Perform* (WP) de EV. Usando a relação entre *Budgeted Cost* (BC) e o WP estimado representado no modelo de resposta EV, Chen (2014) desenvolveu ainda EV em um modelo ACV para prever *Actual Cost* (AC) de ACV.

Em relação a duração do projeto, Batselier e Vanhoucke (2015) adotaram duas técnicas baseadas no EVM que integram, respectivamente, retrabalho e sensibilidade à atividade no tempo de EVM prevendo as extensões do projeto. Lopez-Paredes (2011) introduziram duas novas métricas denominadas por Índice de Controle de Custos (ICC1) e Índice de Controle de Cronograma (ICC2). Ambos os índices compararam as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o risco, custo e cronograma, o que pode ser aplicado nas fases do ciclo de vida do projeto (Chen, 2018).

Sato e Hirao (2013) contribuíram para estabelecer o conceito de orçamento ótimo e esclarecer essa condição e a teoria permite a análise do equilíbrio custo/risco não apenas no nível do projeto, mas também no nível da rede de atividades. Kim, Kim e Cho (2016) estudaram um processo com relação linear entre atividades em que as saídas de um negócio precedente se tornam diretamente os insumos de um sucessor. Narbaev e De Marco (2014) calcularam o *Cost Estimate At Completion* (CEAC) que é um número considerado como novidade de pesquisa e que contribui para a extensão do corpo de conhecimento EVM. Narbaev e De Marco (2014) apresentaram uma nova abordagem metodológica que foi desenvolvida para fornecer previsões CEAC mais precisas, confiáveis e eficazes para os estágios iniciais e intermediários da execução do projeto de construção.

Em relação ao resultado dos projetos, Aljedaibi e Khamis (2019) desenvolveram o *Critical Success Factors* (CSF) é um método para medir, monitorar e controlar fatores críticos de sucesso de sistemas de *software* grande, por meio de escala de metas, custos, infraestrutura de TI, gestão de projetos, entre outros. Rajagopalan e Srivastava (2018) adotaram um método único usando uma variável chamada de *Project Health Index*, capaz de medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base no *Corrective And Preventive*

Actions. Ibrahim, Hanna e Kievet (2020) examinaram as diferenças de desempenho tempo de processamento de pedido de mudança, aumento no prazo do cronograma, aumento de custo, mudança percentual do projeto, qualidade geral dos sistemas do projeto, mudança percentual relacionada à qualidade/valor, mudança percentual relacionada ao *design*.

Durante as fases de iniciação e planejamento do projeto, métricas da abordagem tradicional são incorporadas para prever o uso de recursos humanos e outras necessidades de execução de atividades para apoio ao controle e o monitoramento de projetos (PMI, 2017). Assim, Song, Martens e Vanhoucke (2021) apresentaram a métrica de *Using Schedule Risk Analysis* (SRA) com a restrição de recursos (*Resource Constrained - RC*) para melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos. Pozzana *et al.* (2021) apresentaram um conjunto de dados composto por projetos de engenharia descritos por suas redes de atividades, registros do desempenho e se a atividade foi entregue no prazo.

Em se tratando do envolvimento das métricas com os envolvidos nos projetos, Rajablu *et al.* (2017) desenvolveram o *Stakeholder Metrics-integrated Management Model* que integra as partes interessadas que permite gerenciá-las identificando, comunicando, engajando, capacitando e atendendo as partes interessadas com métricas e ferramentas. Além disso, Hanna, Tadt e Whited (2012) utilizaram análise de *bootstrap* para avaliar o desempenho de um projeto de infraestrutura. Hanna (2016) avaliaram com métrica o desempenho geral mais alto no que se refere à implementação de técnicas de construção enxuta, equipe de liderança de projeto engajada e capacitada e um alto envolvimento de *stakeholders* em todo o projeto.

Algumas métricas possuem foco no desenvolvimento do produto. Fleche *et al.* (2017) implantaram métricas quantitativas e não intrusivas para os *designers*, onde cada métrica utiliza variáveis do *design* do produto planejado. Spijkman *et al.* (2021) propuseram o modelo *Requirements Engineering for Software Architecture* (RE4SA) que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. Além disso, Spijkman *et al.* (2021) forneceram métricas para quantificar o alinhamento entre requisitos e arquitetura para detectar sinais de granularidade, concentrando-se em valores discrepantes em um conjunto de dados. Jethani (2013) apresentou o esforço no nível da tarefa e monitoramento do progresso; Monitoramento de marcos e cronogramas; e Monitoramento de defeitos, com base nas atividades regulares de planejamento e monitoramento de projetos de desenvolvimento de *software*. A visão é estendida por Iqbal, Naeem e Khan (2012), que completaram que a rastreabilidade avalia os vínculos entre os requisitos dentro de um documento e métricas de integridade de requisitos, que verificam se os requisitos especificados estão completos ou não para o desenvolvimento do produto.

Em ambiente de escritório de projetos, Geng *et al.* (2018) integraram o aprendizado organizacional e o desenvolvimento do conhecimento com um processo de seleção de projetos, propondo uma estrutura para avaliar essa integração baseada no *Analytic Hierarchy Process*, ou por meio de uma avaliação de eficiência a ser realizada por meio da atualização da lista de análise de projetos (Ko & Cho, 2020). Por fim, a Tabela 4 apresenta uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Tradicionais para auxiliar no entendimento de cada métrica e a sua contribuição à gestão tradicional.

Tabela 4 – Métricas de Projetos Tradicionais, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição de uso	Métrica Relacionada
<i>Earned Value Management</i> (EVM)	Medir o progresso e desempenho de projetos combinando as medidas de escopo, tempo e custos.	O escopo é fixo e mantém o tempo e custos estimados.	-
<i>Earned Value Management/ Earned Schedule</i> (EVM/ES)	Medir possíveis riscos de maneira preventiva sobre o cronograma do projeto.	Restringe-se aos riscos relativos ao prazo.	EVM
<i>Earned Duration Management</i> (EDM)	Medir as variações do desempenho do cronograma do projeto ou para controlar projetos de inovação radical.	Restringe-se a medir as variações (não as causas) no desempenho do projeto.	EVM, EVM/ES
<i>Planned Value</i> (PV)	Medir o custo orçado para o trabalho programado para ser feito.	Mantém o foco no custo orçado para apenas ser utilizado como <i>baseline</i> para avaliar o desempenho do custo realizado.	EVM
<i>Actual Cost</i> (AC)	Medir o custo realizado pelo trabalho realizado.	Mantém o foco no custo realizado e não avalia os motivos pelo desempenho.	EVM
<i>Earned Value</i> (EV)	Medir o valor agregado com base no escopo, prazo e custo.	Se restringe a avaliar o desempenho do projeto com base no escopo, prazo e custo.	EVM
<i>Coefficient of Variation</i> (CV)	Experimento computacional para avaliar a estabilidade usando a nova métrica de defasagem.	Restringe-se a medir as variações (não as causas) no desempenho do projeto.	EVM
<i>Work Perform</i> (WP)	Medir a combinação de séries temporais e análise de regressão para desenvolver PV em um modelo de resposta EV para prever WP.	Restringe-se em avaliar o desempenho do cronograma.	PV/EV
<i>Budgeted Cost</i> (BC)	Medir o custo orçado estimado.	Restringe-se em avaliar o desempenho de custos.	EV/WP
<i>Actual Cost Value</i> (ACV)	Medir a relação entre BC e o WP estimado representado no modelo de resposta EV, para prever o AC de ACV	Restringe-se em avaliar o desempenho de custos.	AC/EV
Índice de Controle de Custos (ICC1)	Compara as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o cronograma do projeto.	Com base em medições do EVM, restringe-se e avaliar o desempenho de custos.	EVM

Índice de Controle de Cronograma (ICC2)	Compara as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o custo do projeto.	Com base em medições do EVM, restringe-se e avaliar o desempenho do cronograma.	EVM
<i>Cost Estimate at Completion</i> (CEAC)	Medir a estimativa de custo na conclusão que é um número que contribui para a extensão do corpo de conhecimento EVM.	Com base em medições do EVM, restringe-se e avaliar o desempenho de custos.	EVM
<i>Critical Success Factors</i> (CSF)	Medir, monitorar e controlar fatores críticos de sucesso por meio de escala de metas, custos, infraestrutura de TI e gestão de projetos.	Restringe-se em avaliar uma série de variáveis para determinar o sucesso do projeto.	-
<i>Project Health Index</i> (PHI)	Medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base em ações de prevenções e correções.	Está restrito em medir a saúde como progresso do projeto com base em ações de prevenções e correções.	-
<i>Using Schedule Risk Analysis</i> (SRA)	Utiliza a restrição de recursos para medir e melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos.	Restrito em medir o desempenho do projeto sob a ótica de recursos restritos.	-
<i>Stakeholder Metrics-integrated Management Model</i> (SMMM)	Integra as partes interessadas com métricas de identificação, comunicação, engajamento, capacitação e atendimento.	Restringe-se em medir o nível de integração dos <i>stakeholders</i> com o projeto.	-
<i>Requirements Engineering for Software Architecture</i> (RE4SA)	Medir a conexão entre os artefatos do projeto e que facilita a comunicação dentro da equipe de desenvolvimento de <i>software</i> .	Restringe-se em medir o relacionamento entre artefatos do projeto.	-

Fonte: elaborado pelo autor.

Os artigos dessa categoria apresentaram discussões importantes sobre as principais métricas utilizadas pelas organizações para o controle e monitoramento de projetos tradicionais. A partir dessa perspectiva, as métricas como EVM, EV, AC, PV e ES são as mais citadas na literatura e nota-se a introdução de novas métricas como EDM e métricas com *benchmarks* associados para o processo de requisição de informações em projetos.

3.4 Métricas Ágeis de Fluxo

As métricas ágeis de fluxo visam medir o desempenho de projetos ágeis durante o fluxo de desenvolvimento de software ou de produto (Jyothi e Rao, 2017; dos Santos *et al.*, 2018). Assim, as métricas ágeis de fluxo procuram medir desde o ponto de comprometimento (*Sprint Backlog*) até a entrega da funcionalidade para os clientes/*stakeholder*. As métricas

comumente utilizadas nesse contexto são o *Lead Time*, *WIP*, *Cycle Time*, *Work Item Age* entre outras que procuram medir as etapas de desenvolvimento de *software*.

Inserido no *Scrum*, Parrend *et al.* (2014) avaliaram o impacto das práticas de gerenciamento de projetos nos dois primeiros *Sprints*, medindo as práticas de elegibilidade do incremento com impacto mensurável no desempenho das equipes e nível de exigência que levam ao fracasso se não forem implementadas. Parrend *et al.* (2014) destacaram a importância dos princípios do *Scrum* e as propriedades *swarming* são práticas relevantes de sucesso e um sintoma de gerenciamento de desempenho. O *swarming* é uma técnica e uma dinâmica de gestão que consiste em empregar a inteligência coletiva da equipe na execução de uma atividade crítica para resultados mais eficazes do projeto (Parrend *et al.*, 2014).

Dennehy e Conboy (2018) realizaram a avaliação por meio de um estudo empírico do uso de fluxo de trabalho com *Kanban*, obtendo dois resultados. Primeiro, um conjunto de desafios empíricos, tais como a mudança organizacional e a configuração de quadros *Kanban*. Segundo, identificaram que as métricas de *Lead Time* e *Velocity* são as mais utilizadas para o planejamento e nos processos de melhoria contínua no projeto (Dennehy & Conboy, 2018).

Os autores Mas, Mesquida e Pacheco (2020) realizaram um estudo para saber se empresa é ágil e madura, e se um conjunto de métricas de desempenho estiver em conformidade com as melhores práticas de gerenciamento de projetos. Um conjunto de métricas foram identificadas nas empresas ágeis estudadas, tais como *efficiency flow*, *lead time*, *cycle time*, *reopens e tasks finished flow* (Mesquida & Pacheco, 2020). A métrica *efficiency flow* mede quantos itens do *backlog* são entregues por ciclo (Mesquida & Pacheco, 2020). O *lead time* mede o tempo em que um item do *backlog* é iniciado até o momento que é entregue ao cliente (Mesquida & Pacheco, 2020). A métrica *reopens* mede o número de tarefas ou incidentes reabertos e a métrica *tasks finished flow* mede as tarefas finalizadas no fluxo de desenvolvimento, diferentemente do *Throughput*, que mede funcionalidades entregues ou histórias de usuário. (Mesquida & Pacheco, 2020).

Grimaldi *et al.* (2016) apresentaram como uma estrutura ágil escalável, baseada em SAFe 3.0 e mostrou como implementar e medir o desempenho considerando que as funções e responsabilidades são divididas entre a organização e o fornecedor. Grimaldi *et al.* (2016) utilizaram métricas como eficácia e eficiência, custo-hora real e planejada, impedimentos e custo do atraso durante as rodadas de *Sprint*. No mesmo sentido que Ahmed *et al.* (2017) discutiram sobre as métricas *Sprint Burn Down*, *Epic Burn Down Velocity Chart*, *Control Chart* e a maneira como muitos dos cálculos de métricas são fortemente acoplados aos pontos de história para itens como o Épico ou *Sprint*.

Sajedi-Badashian e Stroulia (2020) mostraram que é possível melhorar e medir a eficácia do processo de atribuição de *bugs* por meio na análise de dados do projeto. Desse modo, Sajedi-Badashian e Stroulia (2020) encontraram uma maneira de atribuir a pessoa mais indicada para a resolução do problema e reduzir o *time to market* do produto.

Do ponto de vista de Petersen e Wohlin (2011), o *Lean* fornece ferramentas de análise e melhoria com foco no ciclo de vida do desenvolvimento, enquanto o ágil se concentra em soluções e prescreve conjuntos de práticas para obter agilidade. Petersen e Wohlin (2011) sugeriram análises sobre as métricas como reduzir o *lead time* para obter alta capacidade de resposta às necessidades dos clientes e fornecer um sistema de rastreamento que mostre o status do desenvolvimento de *software*. No sentido do monitoramento, as métricas de processo propostas por Choraś *et al.* (2020) foram utilizadas continuamente para ajudar a garantir a qualidade e estabilidade do *software* como a estimativa de tarefas e rastreamento tarefas em tempo real. Meidan *et al.* (2018) identificaram que o desenvolvimento ágil e empresa de pequeno/médio porte foram os contextos de pesquisa mais frequentemente identificados. Meidan *et al.* (2018) demonstraram que a capacidade, desempenho e maturidade foram os atributos de processo mais medidos, enquanto esforço, desempenho e risco foram os atributos de projeto mais medidos.

O modelo de ciclo de vida de desenvolvimento e manutenção de sistemas proposto por Sadler (2020) aplicou efetivamente diferentes níveis de rigor e métricas de qualidade mais rígidas, dependendo do risco associado aos lançamentos. O modelo de Sadler (2020) foi baseado nas normas relacionadas ao gerenciamento de riscos denominado ER²C SDMLC (*Enterprise Release Risk-Centric Systems Development and Maintenance Life Cycle*). Dentre as métricas avaliadas por Sadler (2020), tivemos a Clareza do Artefato e Envolvimento, Propriedade do Cliente e Aplicabilidade e Rastreabilidade.

Asha e Mani (2018) apresentaram uma metodologia de teste denominada por Teste Simples de Teste de Aceitação aplicando as técnicas de Desenvolvimento Orientado a Testes de Aceitação que podem ser executados como casos de teste de *software* no contexto do DevOps. Além disso, Asha e Mani (2018) utilizaram os algoritmos Tarjan e Breadth-First Search que renderizam 100% de cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste aplicando as métricas de componentes funcionais, aumentando a frequência de entrega contínua e reduzindo o *time to market*.

A Tabela 5 a seguir realiza uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de Fluxo para auxiliar no entendimento de cada métrica e a contribuição de cada uma delas à gestão ágil de projetos.

Tabela 5 – Métricas Ágeis de Fluxo, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
Elegibilidade do Incremento	Avalia o impacto das práticas de gerenciamento de projetos nos dois primeiros <i>Sprints</i> .	Com foco na avaliação entre escopo e incremento, não avalia a agregação de valor do incremento.	-
<i>Lead Time</i> (LT)	Mede a entrega a funcionalidade desde o comprometimento até a entrega ao cliente	Com foco em prazo, não avalia a agregação de valor da entrega da funcionalidade.	-
<i>Cycle Time</i> (CT)	Mede etapas de desenvolvimento do software dentro da iteração.	Com foco em desempenho de prazo, não avalia o valor da entrega.	-
<i>Velocity</i>	Mede quantos pontos de história foram feitos por <i>Sprint</i> .	Com foco em desempenho de esforço, não avalia o valor da entrega da HU.	-
<i>Efficiency Flow</i> (EF)	Mede quantos itens do <i>backlog</i> são entregues por ciclo.	Não avalia a agregação de valor da entrega da funcionalidade.	LT
<i>Reopens</i>	Mede o número de atividades reabertas.	Somente avalia o retrabalho e não mede as causas para aprendizado.	TFF
<i>Tasks Finished Flow</i> (TFF)	Mede as tarefas finalizadas dentro do fluxo de desenvolvimento.	Com foco em tarefas realizadas, não avalia a agregação de valor da entrega do item do <i>backlog</i> .	<i>Reopens</i>
Eficácia e eficiência	Eficácia e eficiência de entregas do projeto durante a rodada da <i>Sprint</i>	Necessário medir as entregas na perspectiva do cliente e/ou do mercado.	LT, TFF, <i>Velocity</i> , EF
Custo hora real e planejada	Custo hora real e planejada de entregas do projeto durante a <i>Sprint</i> .	Medida que contradiz a expectativa de custos fixos em projetos ágeis.	LT, EF
Impedimentos	Impedimentos mapeados durante a rodada da <i>Sprint</i> .	Somente avalia o retrabalho e não mede as causas para aprendizado.	-
Custo do atraso	Custo de atraso de entregas do projeto durante a rodada da <i>Sprint</i> .	Está concentrado em medir o custo do atraso e não mede as causas para reduzir riscos.	LT, TFF, <i>Velocity</i> , EF
<i>Sprint Burn Down</i> (SBD)	Métrica sobre pontos de história no <i>Sprint</i>	Com foco em desempenho de esforço, não avalia o valor da entrega da HU.	<i>Velocity</i> , LT
<i>Epic Burn Down</i> (EBD)	Métrica sobre pontos de história no Épico	Com foco em desempenho de esforço, não avalia a agregação de valor da entrega do épico.	<i>Velocity</i> , LT
<i>Control Chart</i> (CC)	Mede o <i>Cycle Time</i> (ou <i>Lead Time</i>) na <i>Sprint</i>	Com foco em prazo, não avalia a agregação de valor da entrega da funcionalidade.	LT, CT
Atribuição de <i>bugs</i>	Atribui a pessoa mais indicada para a resolução do problema e reduz o <i>time to market</i> do produto.	Apesar de auxiliar na redução do <i>time to market</i> , não mede o valor entregue.	TFF, <i>Reopens</i>
Reduzir <i>Lead Time</i>	Reduz o <i>lead time</i> para obter alta capacidade de resposta às necessidades dos clientes	Com foco em reduzir o prazo da entrega, não avalia a agregação de valor da entrega do item do <i>backlog</i> .	LT
Status Desenvolvimento	Fornece um rastreamento do status do desenvolvimento de <i>software</i> .	O status do desenvolvimento não garante que o escopo e/ou o valor será entregue.	CT, <i>Velocity</i> , SBD, CC
Estimativa de Tarefas (ET)	Estimativa de tarefas para auxiliar na qualidade e estabilidade do software	A estimativa de tarefas tem foco apenas em prazo.	-

Rastreamento de Tarefas (RT)	Rastreamento de tarefas para ajudar na qualidade e estabilidade do software	O rastreamento de tarefas tem foco apenas em status de tarefas.	AR
Capacidade do processo	Medir a capacidade do processo ágil.	Com restrição em medir a capacidade do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Desempenho do processo	Medir o desempenho do processo ágil.	Com restrição em medir o desempenho do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Maturidade do Processo	Medir a maturidade do processo ágil.	Com restrição em medir a maturidade do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Esforço no projeto	Medir o esforço do projeto ágil.	O esforço no projeto tem foco apenas em prazo.	LT, CT, EF, CC, SBD, EBD, ET
Desempenho no projeto	Medir o desempenho do projeto ágil.	O desempenho no projeto tem foco apenas em custo e prazo.	LT, CT, EF, CC, SBD, EBD, ET
Risco no projeto	Medir os riscos do projeto ágil.	Medir riscos não garante sucesso ou entrega de valor pelo projeto.	-
Clareza do Artefato (CA)	Medir a clareza do artefato produzido pelo projeto.	Medir a clareza do artefato pode auxiliar na gestão do <i>backlog</i> , mas não garante entrega de valor.	EPC
Envolvimento e Propriedade do Cliente (EPC)	Medir o envolvimento e propriedade do cliente sobre o artefato produzido pelo projeto.	Medir o envolvimento e propriedade do cliente sobre o artefato auxilia na gestão do <i>backlog</i> . Não garante entrega de valor.	CA
Aplicabilidade e Rastreabilidade (AR)	Medir a aplicabilidade e rastreabilidade do artefato produzido pelo projeto.	Medir a aplicabilidade e rastreabilidade do artefato não garante entrega de valor.	RT
Componentes funcionais	Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste.	Medir a cobertura de cenários e histórias usuário auxilia na qualidade do <i>software</i> , mas não garante entrega de valor.	-

Fonte: elaborado pelo autor.

Os artigos dos autores apresentaram as métricas ágeis relacionadas ao fluxo operacional de projetos utilizando o *Framework Scrum* e o *Kanban*. As métricas mais citadas nos estudos foram as de *Efficiency Flow*, *Lead Time*, *Cycle Time*, *Velocity*, *Sprint Burn Down*, *Epic Burn Down* e *Control Chart*. Verificou-se também que muitos dos cálculos de métricas são fortemente acoplados aos pontos de história como estimativa e monitoramento. Além disso, métricas ligadas os itens de *backlog*, como *reopens*, *tasks finished flow* e *time spent extremes*, e outras métricas baseadas no SAFe 3.0 como eficácia, eficiência, custo-hora real e planejada, resíduos, impedimentos e custo do atraso foram encontradas.

3.5 Métricas Ágeis de Backlog do Produto

As métricas ágeis de *Backlog do Produto* visam medir o desempenho de projetos ágeis durante a etapa de definição do escopo do projeto que, em projetos ágeis, são organizados no *backlog* do produto (Mosser, Pulgar & Reinharz, 2022). O *backlog* do produto é composto pelo escopo do projeto dividido em diversos itens ou HUs que são descobertos e detalhados ao longo do desenvolvimento do produto do projeto (Mosser, Pulgar & Reinharz, 2022). Assim, as métricas ágeis de *Backlog do Produto* procuram medir o esforço de desenvolvimento da funcionalidade, nível de priorização e complexidade por *Use Case Points*, e estimativas por pontos de história.

Kayes, Sarker e Chakareski (2016) propuseram uma métrica no *Product Backlog Rating* (PBR) para avaliar o processo de teste no *Scrum* que forneceu uma estrutura adicional para o gerenciamento da qualidade e uma visão abrangente do processo de teste ao longo do ciclo de desenvolvimento do produto. O PBR de Kayes, Sarker e Chakareski (2016) considera a complexidade dos recursos a serem desenvolvidos em uma iteração do *Scrum*, avalia as classificações dos testes e oferece uma pontuação numérica do processo de teste. No mesmo contexto do processo *Scrum*, Zheng *et al.* (2021) mapearam as métricas de processo e do *Service-Oriented Architecture* (SOA) para auxiliar nas decisões de planejamento e *design* durante a *Sprint*. Zheng *et al.* (2021) sugeriram os *Key Performance Indicators* (KPIs) para ajudar os *stakeholders* a entender os imperativos da técnica proposta no planejamento de recursos, avaliação do desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos baseados em *Scrum* e SOA.

Pavlova *et al.* (2021) determinaram os parâmetros do projeto como as características da entrega futura como o custo, prazo e possíveis desvios de prazo e custo. Tais parâmetros do estudo de Pavlova *et al.* (2021) resultaram em métricas para avaliar o grau de singularidade do projeto e o grau de complexidade do projeto. Para a métrica de complexidade, Pavlova *et al.* (2021) consideraram o número de variantes de soluções tecnológicas, o número de operações para cada variante e o número de alternativas para execução da operação para cada variante da tecnologia do produto do projeto. Wallace e Sheetz (2014) destacaram que a utilidade percebida de uma medida de software pode ser medida por uma avaliação da aplicabilidade da medida ao longo do ciclo de vida, dependência da linguagem de programação, capacidade de prescrever soluções e a validade da medida para definir a complexidade dos requisitos.

Azzeh e Nassif (2016) propuseram um modelo híbrido voltado para medir o esforço e a produtividade para apoiar o processo de estimativa de esforço inicial baseada no tamanho do

Use Case Point (UCP). As análises de Azzeh e Nassif (2016) contaram com um número significativo de projetos coletados que utilizaram como base o UCP em suas estimativas.

A ontologia desenvolvida por Prakash e Viswanathan (2018) acumulou conhecimento sobre os critérios de avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos. Prakash e Viswanathan (2018) incluíram axiomas que determinam a qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com os requisitos funcionais e não-funcionais e versões de produtos.

A Tabela 6 realiza uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de *Backlog* para auxiliar no entendimento de cada métrica e a contribuição de cada uma delas à gestão ágil de projetos.

Tabela 6 – Métricas Ágeis de *Backlog*, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
<i>Product Backlog Rating</i> (PBR)	Mede o nível de prioridade do item do <i>backlog</i>	Medir o nível de priorização dos itens do <i>backlog</i> não garante a priorização no sentido de valor entregue.	-
KPI de Imperativos da Proposta Técnica (IPT)	Auxilia no planejamento de recursos, testar o desempenho da agilidade, valor dos negócios, eficiência da equipe e complexidade de projeto.	Falta de foco na avaliação do KPI com abrangência em diversas fases do projeto num único KPI.	-
Grau de singularidade do projeto	Características da entrega futura como o custo, prazo, possíveis desvios de prazo e custo.	Não faz sentido medir a singularidade de projetos ágeis por custo e prazo (fixos).	IPT, GCP
Grau de Complexidade do Projeto (GCP)	Número de variantes de soluções tecnológicas, o número de operações para cada variante e o número de alternativas para execução da operação para cada variante.	Avalia somente questões tecnológicas para verificar a complexidade sem considerar O mercado e escopo para medir a complexidade.	IPT, GCP
Esforço de desenvolvimento de <i>software</i> (EDS)	Avaliação da aplicabilidade ao longo do ciclo de vida, dependência da linguagem de programação específica, capacidade de prescrever soluções.	Falta de foco na avaliação da métrica com abrangência em diversas fases do projeto numa única métrica.	IPT, GCP
Esforço e produtividade	Apoiar a estimativa de esforço inicial baseada no tamanho do <i>Use Case Point</i> (UCP)	Usar UCP como métrica de planejado versus realizado não garante a entrega o valor esperado.	IPT, GCP, EDS
Avaliação da qualidade das HUs	Avalia a qualidade das HUs, prioridade e o risco das HUs, relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.	Não realiza a medição das HUs que entregaram o valor esperado.	PBR, GCP

Fonte: elaborado pelo autor.

As métricas de *Backlog* de Produto que foram apresentadas nos artigos, trazem métricas importantes para a gestão de requisitos em projetos ágeis. Assim, destacam-se as métricas como o *Product Backlog Rating*, qualidade e valor dos negócios, complexidade de projetos

baseados em *Scrum* e SOA, complexidade do requisito, complexidade da arquitetura de solução e complexidade da mudança. Outros achados tivemos com a estimativa de esforço inicial baseada no tamanho do UCP, avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos e a medição do tamanho do requisito a partir do comportamento criado pelo *Behavior Driven Development* (BDD) usando a notação de Gherkin (Moult & Krijnen, 2020).

3.6 Métricas Ágeis de Produto

As métricas ágeis de produto visam medir a qualidade de construção e de ações para produzir um produto de melhor qualidade e para reduzir o seu *time to market* (Kristensen & Shafiee, 2019; Ries, 2011). Dessa forma, as métricas ágeis de produto nessa sessão medem tanto atividades para melhorar a qualidade do código-fonte, melhoria do processo de testes de *software*, a gestão e rastreabilidade de *bugs* e o processo de desenvolvimento orientado a testes. Além disso, a categoria buscou apresentar as métricas para medir o nível de satisfação do cliente/*stakeholder* com o produto recebido.

Mäkiahho, Vartiainen e Poranen (2022) apresentaram uma ferramenta de monitoramento de métricas que foi desenvolvida por alunos de graduação e pós-graduação para apoiar os membros do projeto, gerentes de projeto e gestores. Segundo Mäkiahho, Vartiainen e Poranen (2022) a ferramenta permitiu acompanhar a quantidade de *commits*, horas trabalhadas e se os requisitos permaneciam muito tempo em estágios no processo de desenvolvimento.

Os experimentos de Chang (2015) foram usados para detectar itens de risco de software a partir do *cluster* de dados oriundos de projetos. Tais dados foram apresentados por Chang (2015) no formato de métricas com os riscos foram mitigados e fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de *software*. Dentre as métricas apresentadas por Mascarenhas *et al.* (2014), temos o crescimento nas vendas de novos produtos, o orçamento dedicado à análise ou verificação do cliente, projetos orientados para o cliente, projetos e programas respeitando custos e orçamento, marcos respeitados e componentes reutilizáveis.

A pesquisa de Singh *et al.* (2019) propuseram análises de complexidade de programas orientados a objetos em Java com base na métrica *Chidamber-Kemerer* (CK) e o desempenho são medidos com uso da métrica *Mean Time to Execution*. Como resultado indicado por Singh *et al.* (2019), realizar reengenharia usando práticas ágeis é benéfico em termos de implementação de requisitos, estimativa de custo e melhoria de desempenho. Nesse sentido, Isong e Obeten (2013) realizaram uma validação empírica das métricas CK e *Source Lines of*

Code (SLOC). A validação empírica de Isong e Obeten (2013) das métricas CK e SLOC e os modelos que as utilizaram em três aspectos: (1) nível de significância e insignificância com propensão a falhas (PF), (2) estado de validação e (3) utilidade na qualidade de software. Segundo Isong e Obeten (2013), as métricas SLOC, *Coupling between Objects* (CBO), *Response for a Class* (RFC), *Weighted Methods per Class* (WMC) possuem forte relação com PF e são consideradas os melhores preditores de PF com impacto na qualidade de *software*.

Nesse sentido que Concas *et al.* (2012) também estudou diversas iniciativas em projetos ágeis que utilizaram as métricas CK, SLOC, CBO, RFC e WMC para medir a qualidade de *software* em diversos projetos. Ainda sobre a qualidade de código-fonte, Yamashita e Counsell (2013) investigaram o potencial de *code smells* (potenciais problemas no código) para refletir indicadores de manutenibilidade em nível de *software*. Ao comparar os *code smells* com outras abordagens, Yamashita e Counsell (2013) descobriram que as medidas estruturais forneceram mais informações sobre qual sistema tinha o *design* equilibrado, mas essa medida ignorou quando as tarefas de manutenção eram de tamanho pequeno ou médio. O estudo de Athanasiou *et al.* (2014) apresentou um modelo que avalia a qualidade do código de teste combinando métricas de código-fonte que refletem três aspectos principais da qualidade do código de teste: integridade, eficácia e manutenibilidade. Athanasiou *et al.* (2014) encontraram uma correlação positiva e estatisticamente significativa entre a qualidade do código de teste, rendimento e produtividade. As contribuições do estudo de Athanasiou *et al.* (2014) iniciaram a partir do modelo que combina métricas para fornecer uma medida da qualidade do código de teste e a investigação empírica que demonstrou uma correlação positiva significativa entre a qualidade do código de teste e na resolução de *bugs*.

Sobre a experiência do usuário do produto, de Day *et al.* (2019) utilizaram o *Net Promoter Score* (NPS) para avaliar a viabilidade de um sistema de apoio à decisão no Registro Eletrônico de Saúde (RES). O NPS foi avaliado como resultado de esforços em testes de usabilidade e a implementação em larga escala com efeito positivo no paciente e na tomada de decisão compartilhada (Day *et al.*, 2019). Simpson *et al.* (2021) procurou reduzir a carga do RES com um *Sprint* de otimização de duas semanas que trabalhou com 19 provedores de prática avançada em uma unidade de especialidade. Simpson *et al.* (2021) mediram o NPS o tempo gasto no RES com base nos dados de log do usuário, onde o NPS aumentou de 6 para 60 e para o resultado da *Sprint* de duas semanas o NPS foi de 93.

Üsfekes *et al.* (2019) investigaram por meio de uma revisão da literatura a utilidade de mecanismos de incentivo para rastreamento eficiente de *bugs* no *Application Lifecycle Management*. Para Üsfekes *et al.* (2019), isso colabora positivamente para os resultados de métricas que olham para a quantidade de *bugs* resolvidos por unidade de tempo. Abdelrahman

Aljemabi *et al.* (2020) mostraram que a *Bug Tracking System-based*, exibem três diferentes níveis de coordenação (Plano, Consciente e Reflexivo) com atividades de colaboração e na métrica de tempo médio para reparo para avaliar o tempo de resolução de *bugs*.

Costa, Vasconcelos e Fragoso (2020) destacaram as etapas de identificar as métricas aplicáveis à arquitetura dos projetos e para quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos. As métricas foram propostas por Costa, Vasconcelos e Fragoso, B. (2020) foram organizadas em métricas de Funcionalidade, Confiabilidade, Eficiência, Manutenibilidade, Portabilidade e Alinhamento.

Ibban *et al.* (2018) entenderam as atividades de engenharia de software relacionadas a Ofertas Iniciais de Moedas (*Initial Coin Offering – ICO*) que é uma forma inovadora de financiar projetos baseados em *blockchain*. Além disso, Ibban *et al.* (2018) buscaram fazer uma comparação entre ICOs e ICOs ágeis. Cerca de 5% das ICOs examinadas por Ibban *et al.* (2018) aplicam práticas ágeis e analisaram as ICOs ágeis em termos de métricas de código-fonte, versões de idioma e uso de ferramentas de teste. Nidagundi e Novickis (2016) descobriram possibilidades de adoção dos modelos e métricas de transformação *Lean* no plano de teste de software para simplificar o processo de teste para uso posterior dessas métricas de teste em Canvas. Com o Canvas, Nidagundi e Novickis (2016) afirmaram que as métricas de teste podem nos dar *feedback* para o processo de teste de *software*. Ainda no tema de qualidade, Shawky e Abd-El-Hafiz (2016) apresentaram uma abordagem que estuda o efeito do processo de desenvolvimento na qualidade *software* e verificaram se existe relação entre o tipo de método de desenvolvimento utilizado e os valores de algumas métricas indicadoras de qualidade. Shawky e Abd-El-Hafiz (2016) afirmaram que as métricas usadas incluem métricas de complexidade, acoplamento e coesão, sendo as duas últimas sendo utilizadas para medir a complexidade. Ainda sobre processo de software, Tahir, Rasool e Noman (2018) identificaram 29 modelos de medição de software e 4 ferramentas projetadas para gerentes de projeto em pequenas e médias empresas. Assim, o estudo de Tahir, Rasool e Noman (2018) apresentou que 51% dos modelos de medição são referentes a melhoria de processo de software, 60% sobre uso de especialistas em medição e experiência, 40% sobre o uso de padrões de medição e 22% sobre o uso de ferramentas automatizadas.

Tanveer *et al.* (2019) desenvolveram um método híbrido que utiliza informações de análise de impacto de mudança na estimativa de esforço e do modelo de estimação baseado em árvores impulsionadas por gradiente (AIG). A avaliação subjetiva de Tanveer *et al.* (2019) sobre o *Hybrid Effort Estimation in Agile Software Development* (HyEEASE) e do *Planning Poker* (PK) foi sobre medir a facilidade de uso percebida, utilidade e aprendizagem. Tanveer *et al.* (2019) avaliaram que o PK obteve classificações superiores ao HyEEASE em facilidade

de uso percebida e capacidade de aprendizado, entretanto, os avaliados consideraram que HyEEASe requer menos esforço mental. Em relação à métrica de precisão da estimativa, Tanveer *et al.* (2019) concluíram uma estimativa puramente baseada no modelo AIG não teve um desempenho melhor do que uma estimativa baseada no HyEEASe.

Pradhan e Nanniyur (2021) apresentaram um estudo de caso sobre a transformação em larga escala de um sistema de gestão de qualidade legado para um sistema moderno com aprendizado de máquina, dados conectados, operações integradas e *Big Data*. Pradhan e Nanniyur (2021) identificaram que as métricas de qualidade precisam abranger a *Sprint* ou Projeto, Lançamentos, Portfólio e Visão Empresarial. Pradhan e Nanniyur (2021) validaram a eficácia da transformação nos perfis de defeitos recebidos, taxas de falhas e cronograma de maturidade de lançamento. Kamulegeya, Mugwanya e Hebig (2018) buscaram entender o uso e os benefícios percebidos da medição em *startups* de *software* na África Oriental e identificaram que as *startups* estão usando uma série de métricas orientadas a negócios e produtos e as *startups* mais antigas utilizam as métricas orientadas a projetos e as métricas organizacionais são as métricas menos utilizadas.

Bakhtiary, Gandomani e Salajegheh (2020) mediram a aplicação das abordagens de *Test-Driven Development* (TDD) e Não-TDD (NTDD) para compará-los, dada a qualidade do resultado do trabalho e a produtividade dos desenvolvedores em uma empresa de *software*. Bakhtiary, Gandomani e Salajegheh (2020) mostraram que ao aumentar a escala do projeto, o número de defeitos descobertos antes e após o lançamento aumentou mais na abordagem NTDD em relação ao TDD. Bakhtiary, Gandomani e Salajegheh (2020) observaram que houve uma redução na satisfação do cliente no NTDD em comparação com a abordagem TDD em cada ciclo, e teve desempenho melhor. Outra análise de Bakhtiary, Gandomani e Salajegheh (2020) e Antinyan (2014) trouxeram que medir a complexidade ciclomática que corresponde o nível de complexidade de um programa de computador e resultou que a abordagem TDD reduziu a complexidade no código em comparação com a abordagem NTDD. No mesmo sentido da produtividade, Hernández, *et al.* (2020) estabeleceu o nível de relevância das métricas de produtividade de equipes de desenvolvimento ágil de *software*. Para atingir o objetivo, o nível de relevância das métricas de produtividade da equipe foi identificado por Hernández *et al.* (2020) em três aspectos fundamentais: (i) entrega antecipada e frequente de *software*, (ii) agregação de valor e (iii) aspectos relacionados ao esforço. Hernández *et al.* (2020) sugeriram que é relevante para os profissionais medirem a produtividade entregando *software* com antecedência e frequência com alta relevância o esforço de implantação de *software* que resolva os problemas de clientes. Hernández *et al.*

(2020) afirmaram que o cálculo sistemático do retorno do investimento é visto como uma forma relevante de medir a agregação de valor.

Com o enfoque no problema de seleção de atributos no contexto da estimativa de qualidade de *software*, o estudo de Gao *et al.* (2011) investigou comparativa é apresentada para avaliar nossa proposta de abordagem híbrida de seleção de atributos. Dado esse contexto, Gao *et al.* (2011) destacaram que uma seleção inteligente de métricas de *software* antes de treinar um modelo de previsão de defeitos provavelmente melhorará o resultado, removendo recursos redundantes e menos importantes.

Perkusich *et al.* (2017) sugeriram que o modelo com Redes Bayesianas e que abordasse a avaliação da qualidade de processo e de produto a partir de métricas das métricas de qualidade do incremento, dos eventos do *Scrum* e do código-fonte. Assim, Perkusich *et al.* (2017) identificaram nos nós uma oportunidade para analisar a qualidade do trabalho da equipe do projeto e avaliar as chances de sucesso do projeto.

Savola, Frühwirth e Pietikäinen (2012) demonstraram o potencial prático das métricas de segurança orientadas a riscos, particularmente ao oferecer alguma visibilidade inicial da eficácia e eficiência da segurança. A principal contribuição do estudo de Savola, Frühwirth e Pietikäinen (2012) está na avaliação do potencial e significado de uma metodologia de desenvolvimento de métricas de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de *software*. Assim, a Tabela 7 apresenta uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de Produto para auxiliar no entendimento de cada métrica e a sua contribuição na gestão ágil de projetos.

Tabela 7 – Métricas Ágeis de Produto, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
Quantidade de <i>Commits</i> (QC)	Quantidade de novos códigos-fonte publicados.	Avaliar código-fonte publicados não é uma avaliação de produtividade.	-
Horas Trabalhadas (HT)	Quantidade de horas trabalhadas durante um período.	Não avalia horas produzidas que é diferente de horas trabalhadas.	TED
Tempo nos Estágios em Dev (TED)	Tempo do requisito nos estágios de desenvolvimento.	Como o <i>Work Item Age</i> do <i>Kanban</i> , restringe-se na medição dos estágios para apontar problemas pontuais.	HT
Características dos riscos de software	Fornecer informações sobre modelos de risco que podem ser usados para entender as características dos riscos de <i>software</i> .	Modelos de risco certamente não avaliam hipóteses de negócio para avaliar riscos de software.	-
Crescimento nas vendas de novos produtos (CVNP)	Mede o aumento nas vendas de novos produtos.	Apesar da boa prática da métrica, restringe-se a venda e não avalia as partes do produto que os clientes mais gostam.	POC, ODVC, NPS
Orçamento	Mede os orçamentos com dedicação	Restringe-se em validar o orçamento	POC,

Dedicado à Verificação do Cliente (ODVC)	de análise e verificação do cliente.	junto ao cliente. Caso o escopo mude (que é bem provável que ocorra), o orçamento teria que ser revalidado.	CVNP, NPS
Projetos Orientados para o Cliente (POC)	Mede o número de projetos criados e orientados para o cliente.	Dado que todos os projetos deveriam ser orientados ao cliente, não existe muita aplicabilidade dessa métrica.	ODVC, NPS
Projetos respeitando custos e orçamento (PRCO)	Mede o número de projetos e programas respeitando custos e orçamento.	Em projetos ágeis o custo é fixo. Essa medida restringe a avaliação da relação de valor entregue versus custos realizado.	TED, POC, ODVC, HT, MR
Marcos Respeitados (MR)	Mede o número de marcos respeitados e entregues.	Marcos ou épicos, restringe-se em avaliar a entrega sem avaliar também os resultados do <i>feedback</i> dos clientes.	POC, ODVC
Componentes Reutilizáveis (CR)	Mede o número de componentes de software reutilizáveis dentro ou entre projetos.	Restringe-se na reutilização de componentes de software, mas poderia verificar o quanto desse reuso agregou na redução de falhas e desempenho.	NI, NM, NP, BRUT
<i>Chidamber-Kemerer</i> (CK)	Análises de complexidade de código orientado a objetos em Java.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NI, NM, NP, BRUT
<i>Mean Time to Execution</i> (MTE)	Mede o desempenho do código com o tempo médio de execução	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	SLC
<i>Source Lines of Code</i> (SLC)	Mede o número de linhas do código-fonte do programa.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	MTE
<i>Coupling between Objects</i>	Representa o número de classes acopladas a uma determinada classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	MS, QCF, CC
<i>Response for a Class</i>	Número total de métodos que potencialmente podem ser executados em resposta a uma mensagem recebida por um objeto de uma classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
<i>Weighted Methods per Class</i> (WMC)	Mede a soma da complexidade dos métodos em uma classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NI, NM, NP, BRUT
Bugs Resolvidos por Unidade de Tempo (BRUT)	Métrica que olha para a quantidade de <i>bugs</i> resolvidos por unidade de tempo.	Com restrição de falhas resolvidas, não faz avaliação se a redução contribuiu para o <i>time-to-market</i> .	NM
Nível de Alinhamento da Funcionalidade (NAF)	Mede o nível de alinhamento das funcionalidades com as estratégias organizacionais.	Com restrição de alinhamento funcional com estratégia organizacional, não faz avaliação se tal alinhamento resulta em valor de negócio.	NC, NPS
Nível de Confiabilidade (NC)	Mede o nível de confiabilidade em termos de qualidade de software.	O nível de confiabilidade pode ser uma parte da avaliação de valor para segurança da informação.	NAF, NM
Nível de Eficiência e Eficácia (NEE)	Mede o nível de eficiência em termos de qualidade de software.	O nível de eficiência pode ser uma parte da avaliação de valor.	NE, NC, NM, NI

Nível de Eficácia (NE)	Mede o nível de eficácia em termos de qualidade de <i>software</i> .	O nível de eficácia pode ser uma parte da avaliação de valor.	NRR, NM, NI
Nível de Integridade (NI)	Mede o nível de integridade em termos de qualidade de <i>software</i> .	O nível de integridade pode ser uma parte da avaliação de valor para segurança da informação.	BRUT, NC, NP
Nível de Manutenção (NM)	Mede o nível de manutenibilidade em termos de qualidade de <i>software</i> .	O nível de manutenibilidade pode ser uma parte da avaliação de <i>time-to-market</i> .	BRUT, NC, NP
Nível de Portabilidade (NP)	Mede o nível de portabilidade do <i>software</i> com outras arquiteturas de <i>software</i> .	Restringe-se na reutilização de componentes de <i>software</i> , mas poderia verificar o quanto desse reuso agregou na redução de falhas e desempenho.	NEE, NM
Teste em Canvas	Métrica de transformação <i>Lean</i> no plano de teste de <i>software</i> para simplificar o processo de teste.	Enxugar o processo de teste pode contribuir apenas ao <i>time-to-market</i> .	NEE, NM, BRUT
<i>Net Promoter Score</i>	Mede o nível de satisfação dos clientes em relação do produto recebido e sua experiência com ele.	Métrica comumente utilizada para avaliar e experiência geral dos clientes e não por determinadas partes do produto.	CVNP, ODVC, POC, PRCO, MR
Complexidade por acoplamento	Mede a complexidade por acoplamento que define o número de recursos de <i>software</i> por solução.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
Complexidade por coesão	Mede a complexidade por coesão que define o número de coesão intra componentes de <i>software</i> .	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
<i>Previsão por Planning Poker</i>	Mede o nível de compreensão e assertividade da estimativa.	Restringe-se em contribuir com o planejamento das <i>Sprints</i> .	-
Qualidade por <i>Sprint</i>	Mede o nível de qualidade do <i>software</i> por <i>Sprint</i> .	O nível de qualidade pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Incremento (QI)	Mede o nível de qualidade do <i>software</i> por <i>release</i> ou incremento.	O nível de qualidade por <i>release</i> ou incremento pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Solução (QS)	Mede o nível de qualidade do <i>software</i> por Solução.	O nível de qualidade por solução pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Portfólio (QP)	Mede o nível de qualidade do <i>software</i> por Portfólio de projetos.	O nível de qualidade por portfólio pode ser uma parte da avaliação de valor organizacional.	BRUT, NC
Qualidade por Visão Empresarial	Mede o nível de qualidade do <i>software</i> alinhado com a Visão Empresarial.	O nível de qualidade por visão empresarial pode ser uma parte da avaliação de valor organizacional.	BRUT, NC, CVNP, ODVC, POC, PRCO, MR
<i>Code smells</i> (CS)	Mede potenciais problemas em código-fonte.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	QCF, QC
Usou TDD (UT)	Mede se o projeto utilizou o <i>Test-Driven Development</i>	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NUT
Não usou TDD (NUT)	Mede se o projeto não utilizou o <i>Test-Driven Development</i>	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	UT

Complexidade ciclomática (CC)	Mede a complexidade ciclomática (complexidade do código-fonte).	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF
Qualidade do código-fonte (QCF)	Mede a qualidade do código-fonte.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QC
Qualidade dos Eventos do Scrum (QES)	Mede a qualidade dos eventos do Scrum, como <i>Sprint Planning</i> , <i>Daily</i> , <i>Sprint Review</i> e <i>Sprint Retrospective</i> .	Restringe-se em avaliar os eventos do processo do Scrum.	TED
Entrega Antecipada e Frequente de Software (EAFS)	Mede a relevância da produtividade entregando <i>software</i> com antecedência e frequência.	Avaliar a produtividade e a eficiência versus eficácia, sendo que a entrega antecipada e frequente reduz o <i>time-to-market</i> e aumenta o <i>feedback</i> .	QI, QS, QP
Agregação de valor	Mede o retorno do investimento como uma forma relevante de medir a agregação de valor	Medir o retorno do investimento pode ser uma parte da avaliação de valor.	NC, VCNP, ODVC, POC, PRCO,MR
Aspectos relacionados ao esforço	Mede a relevância do esforço de implantação de <i>software</i> que resolva os problemas de clientes.	Usar relevância do esforço de implantação de <i>software</i> não garante que a produção algo que não entrega o valor esperado.	ODVC, POC,PRCO
Modelo de previsão de defeitos	Modelo que mede a previsão de defeitos de <i>software</i> com base em outras métricas.	Restrito apenas para reduzir o risco de falhas durante as fases de teste de <i>software</i> .	EAFS, QI, QS, BRUT.
Segurança orientada a riscos	Métrica de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de <i>software</i> .	Medir a segurança orientada a risco pode ser uma parte da avaliação de valor para segurança da informação.	-
Melhoria de processo de <i>software</i>	Métricas para avaliar melhorias no processo de desenvolvimento de <i>software</i> .	Melhorar o processo de desenvolvimento pode contribuir apenas a entrega contínua e frequente de <i>software</i> .	EAFS, QES

Fonte: elaborado pelo autor.

Os artigos apresentaram métricas que avaliaram diferentes aspectos na avaliação no desenvolvimento de produto, seu *feedback* e o valor entregue. Assim, estudos apresentaram métricas baseadas em modelos de risco que podem ser usados para entender as características dos riscos de *software*, crescimento nas vendas de novos produtos, medição de projetos orientados para o cliente, projetos e programas respeitando custos e orçamento, marcos respeitados e componentes de *software* reutilizáveis. No tocante a complexidade de programas orientados a objetos, autores dessa categoria propuseram as métricas de CK e CBO para avaliar o nível de significância e insignificância com propensão a falhas, estado de validação e utilidade na qualidade de *software* como *bugs* resolvidos por unidade de tempo. Outras métricas aplicáveis à arquitetura dos projetos e para quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos, como métricas de funcionalidade, confiabilidade, eficiência, manutenibilidade, portabilidade e alinhamento de negócio. Para rastrear as abordagens ágeis, outros estudos apresentaram as métricas de complexidade, acoplamento e

coesão. Outros autores mediram o nível de satisfação de clientes em projetos que usam o TDD e avaliaram o nível de satisfação de clientes em relação ao produto, onde identificou-se uma grande adoção da medida do NPS. Por fim, a seção seguinte apresenta, para cada categoria, a classificação das métricas utilizadas para cada prática, tanto ágil como tradicional.

4. CLASSIFICAÇÃO DAS MÉTRICAS POR CATEGORIA

A Tabela 8 destaca a classificação das métricas encontradas e suas respectivas categorias mapeadas durante o processo de RSL, além das subcategorias e processos de medição mapeados após a análise dos resultados. A classificação por Fase do Ágil apresentada na Tabela 8 foi inspirada nas etapas *Initiate*, *Discovery*, *Delivery* e *Release* dos projetos ágeis sugerido pelo *The Agile Landscape – Deloitte* (TAL). Dessa forma, a categoria de Métricas de Projetos Tradicionais não foram classificadas segundo o TAL.

A partir das etapas *Initiate*, *Discovery*, *Delivery* e *Release* do modelo TAL e fazendo um relacionamento com as métricas identificadas na Tabela 8³, são identificadas algumas lacunas no âmbito da medição em projetos ágeis. Dessa maneira, observa-se que a etapa de *Initiate* não foi incluída na classificação devido à ausência de métricas que avaliassem as ideias iniciais do produto para o projeto ágil como, por exemplo, medir o número de hipóteses de produto que se confirmaram como importantes para os clientes versus as que refutaram. Outra lacuna identificada, apesar de etapa de *Release* ser identificada na Tabela 8, muitas métricas de lançamento de produto com ou sem problemas de funcionamento, com *feedback* em nível de valor e funcional da parte do mercado e de clientes.

Assim, a etapa de *Release* somente contemplou métricas relacionadas a versão de lançamento, sem se preocupar com os resultados após o uso de tal versão no mercado e pelos clientes. Em suma, as lacunas se apresentam nas etapas de *Initiate* e *Release*, onde são cruciais para se identificar quais ideias (*Initiate*) foram realmente convertidas em valor para o cliente (*Release*). Desde a gestão tradicional de projetos que a etapa de *Initiate* dá suporte à gestão de portfólio e programas. A etapa de *Discovery*, *Delivery* e *Release* suportam as iniciativas de projetos ou times. O nível de portfólio comumente gerencia um escopo e custos macro dos projetos. O nível de programa gerencia os projetos que possuem dependência técnica e/ou funcional (Lappi *et al.*, 2019). O nível de projeto ou times gerenciam as iniciativas em nível de time de desenvolvimento (Kock & Gemünden, 2019).

³ A Tabela 19 presente no item 11 Anexo, contém o detalhamento do relacionamento das métricas encontradas no Estudo 2, separadas pelas etapas de *Discovery* e *Delivery*, objetivo de cada métrica e seus respectivos autores.

Tabela 8 - Classificação das métricas por categoria e subcategoria

Categoria	Subcategoria	Fase	Métrica	Resumo das Métricas
Métricas Ágeis de Backlog do Produto	Complexidade	<i>Delivery</i>	Grau de complexidade do projeto e KPI de Imperativos da técnica proposta.	Com o número de variantes de soluções tecnológicas, auxilia no planejamento de recursos, testar o desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos.
	Prioridade	<i>Delivery</i>	<i>Product Backlog Rating</i> e Avaliação da qualidade das histórias de usuários (HU).	Mede o nível de prioridade do item do <i>backlog</i> e ajuda na qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.
	Monitoramento do Projeto	<i>Delivery</i>	Esforço de desenvolvimento de software, Esforço e produtividade e Grau de singularidade do projeto.	Avaliação da aplicabilidade ao longo do ciclo de vida, dependência da linguagem de programação específica e capacidade de prescrever soluções. Determina as características da entrega como o custo, prazo, possíveis desvios de prazo e custo.
Métricas Ágeis de Fluxo	Fluxo ou Processo	<i>Delivery</i>	Reduzir Lead Time, <i>Lead Time</i> , <i>Cycle Time</i> , <i>Efficiency flow</i> , <i>Tasks finished flow</i> , <i>Control Chart</i> , Capacidade do processo, Desempenho do processo e Maturidade do Processo.	Mede a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente e permite medir quantos itens do <i>backlog</i> são entregues por ciclo. Permite reduzir o <i>lead time</i> para alta capacidade de resposta às necessidades de clientes.
	Iteração ou <i>Sprint</i>	<i>Delivery</i>	Eficácia e eficiência, <i>Sprint Burn Down</i> , impedimentos, Custo do atraso, Custo hora real e planejada e Previsão por <i>Planning Poker</i> .	Realiza a medição por <i>Sprint</i> ou iteração considerando os pontos de história, nível de assertividade das estimativas, impedimentos, custos para medir a eficiência e eficácia do projeto.
	Incremento ou Artefato	<i>Delivery</i>	Clareza do Artefato, Envolvimento e Propriedade do Cliente, Aplicabilidade e Rastreabilidade	Medir a clareza, rastreabilidade e aplicabilidade do artefato do projeto, os níveis de envolvimento e propriedade do cliente aos entregáveis do projeto.
	Estimativa e Esforço	<i>Delivery</i>	<i>Epic Burn Down</i> , <i>Velocity</i> , Estimativa de tarefas e Esforço no projeto e Status Desenvolvimento.	Métrica sobre pontos de história no Épico (abrange uma ou muitas <i>Sprints</i>), mede o desempenho (<i>velocity</i>) através dos pontos de história feitos por <i>Sprint</i> . Realiza a estimativa de tarefas para ajudar na qualidade do software.
	Monitoramento	<i>Delivery</i>	Componentes funcionais, Desempenho no	Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do

Categoria	Subcategoria	Fase	Métrica	Resumo das Métricas
	do Projeto		projeto e Risco no projeto.	processo de teste e medir o desempenho do projeto. Medir os riscos do projeto ágil para dar visibilidade ao <i>backlog</i> do produto e sua priorização.
	Qualidade do Produto	<i>Delivery</i>	Rastreamento de tarefas, Atribuição de <i>bugs</i> e <i>Reopens</i> .	Rastreamento de tarefas para ajudar a garantir a qualidade e estabilidade do software. Através da atribuição da pessoa mais indicada para a resolução do problema aberto ou reaberto e reduzindo o time to <i>market</i> do produto.
Métricas Ágeis do Produto	Monitoramento do Projeto	<i>Delivery</i>	Horas trabalhadas em atividades, Tempo nos estágios de desenvolvimento, Aspectos relacionados ao esforço, Características dos riscos de software, Segurança orientada a riscos, Projetos/programas respeitando custos e orçamento e Marcos respeitados.	Analisa a quantidade de horas trabalhadas durante um período (<i>Sprint</i> ou ciclo), o tempo do requisito nos estágios de desenvolvimento, marcos e esforço para entregar ao cliente. Fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de software.
	Qualidade do Código-Fonte	<i>Delivery/Release</i>	<i>Chidamber-Kemerer</i> , <i>Source Lines of Code</i> , <i>Coupling between Objects</i> , <i>Response for a Class</i> , <i>Weighted Methods per Class</i> , <i>Mean Time to Execution</i> , Componentes reutilizáveis, Complexidade por acoplamento, Complexidade por coesão, <i>Code smells</i> , Qualidade do código-fonte, Complexidade ciclomática, Usou <i>Test-Driven Development</i> , Não usou <i>Test-Driven Development</i> e Quantidade de <i>commits</i> .	Análises de complexidade de programas orientados a objetos, códigos-fonte publicados, de número de linhas de código, reuso de componentes e a avaliação da qualidade de implementação por melhores práticas orientadas a objetos com baixo acoplamento e alta coesão.
	Qualidade de Software	<i>Delivery/Release</i>	<i>Bugs</i> resolvidos por unidade de tempo, Nível de Confiabilidade, Nível de Eficiência/Eficácia, Nível de Eficácia, Nível de Integridade, Nível de <i>Manutenabilidade</i> , Nível de Portabilidade, Teste em <i>Canvas</i> , Qualidade por <i>Sprint</i> , Qualidade por <i>Release/Incremento</i> , Qualidade por Solução,	Realiza a medição da quantidade de <i>bugs</i> resolvidos por unidade de tempo e mede o nível de confiabilidade, eficiência, eficácia, integridade, manutenibilidade e portabilidade em termos de qualidade de software. Utiliza práticas <i>Lean</i> no plano de teste de software para simplificar o processo de teste.

Categoria	Subcategoria	Fase	Métrica	Resumo das Métricas
			Qualidade por Portfólio, Qualidade por Visão, Empresarial, Qualidade dos Eventos do Scrum e Modelo de previsão de defeitos.	
	Relação Clientes e Stakeholders	<i>Delivery/Release</i>	Crescimento nas vendas de novos produtos, Projetos orientados para o cliente, Orçamento dedicado à análise ou verificação do cliente, Nível de Alinhamento da Funcionalidade, Agregação de valor e <i>Net Promoter Score</i> .	Mede os orçamentos com dedicação de análise e verificação do cliente e o número de projetos criados e orientados para o cliente. Mede o nível de satisfação dos clientes em relação do produto recebido, sua experiência e agregação de valor.
	Entrega Contínua	<i>Delivery/Release</i>	Melhoria de processo de software e Entrega antecipada e frequente de software.	Utilizou métricas de produto para avaliar melhorias no processo de desenvolvimento de <i>software</i> e a relevância da produtividade entregando software com antecedência e frequência.
Métricas de Projetos Tradicionais	Análise do Valor Agregado	Não se aplica	EVM (<i>Earned Value Management</i>) e <i>Work Perform</i> .	Mede o desempenho e o progresso de projetos, com a capacidade de combinar medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido. Combina séries temporais e análise de regressão para desenvolver PV (<i>planned value</i>) e resposta EV (<i>earned value</i>) para prever <i>work perform</i> .
	Qualidade do Produto	Não se aplica	Monitoramento de defeitos, Qualidade geral dos sistemas do projeto e Mudança percentual relacionada à qualidade/valor.	Monitora os defeitos identificados durante o desenvolvimento do projeto e monitora a mudança do percentual relacionado à qualidade/valor do produto.
	Monitoramento do Projeto	Não se aplica	Variação de Desempenho, Número de defasagens, EDM (<i>Earned Duration Management</i>), Índice de Controle de Cronograma, Desempenho de cada atividade, Monitoramento de marcos e cronogramas, Análise <i>bootstrap</i> , <i>Project Health Index</i> , Esforço da tarefa e monitoramento, Índice de Controle de Custos, Análise do equilíbrio custo/risco, Análise de Risco do Cronograma com	Análise de inicialização para avaliar o desempenho de um projeto e a classificação de problemas de projetos para identificar problemas em projetos que provavelmente falharão antes de começarem. Calcula o número de defasagens (atrasos) no cronograma e mede os custos do projeto. Utiliza o conceito de orçamento ótimo no nível do projeto e no nível da rede de atividades.

Categoria	Subcategoria	Fase	Métrica	Resumo das Métricas
			restrição de recursos e <i>Critical Success Factors</i> .	
	Alinhamento Organizacional / Conhecimento	Não se aplica	Seleção de Projetos.	Integraram o aprendizado organizacional e o desenvolvimento do conhecimento adquirido por meio do processo de seleção de projetos.
	Requisitos do Produto	Não se aplica	Modelo RE4SA e Integridade de requisitos.	Modelo <i>Requirements Engineering for Software Architecture</i> (RE4SA) que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. Avalia se os requisitos especificados são completos ou não para o desenvolvimento.
	Análise da Qualidade do Cronograma	Não se aplica	Proporção de tarefas detalhadas para marcos, Porcentagem de tarefas no caminho crítico, Tipo de relacionamento, Marcos sem predecessor ou sucessor e Impacto Predecessor no Sucessor.	Calcula o percentual de tarefas do projeto no caminho crítico do cronograma e investiga a proporção de tarefas do projeto que foram detalhadas para marcos específicos. Avalia os marcos do cronograma sem definição de predecessor e sucessor e calcula o tipo de relacionamento de tarefas predecessoras.
	Gestão de Mudanças	Não se aplica	Tempo de resolução de pedido de mudança e Mudança percentual relacionada ao design.	Calcula o tempo de processamento de pedido de mudança no projeto e avalia a mudança percentual relacionada ao design.
	Relação Cliente/ Stakeholders	Não se aplica	<i>Stakeholder Metrics-integrated Management Model</i>	Integra as partes interessadas com métricas de identificação, comunicação, engajamento, capacitação e atendimento.

Fonte: dados da pesquisa, 2022.

Dessa forma, a Figura 2 apresenta o relacionamento entre as etapas do TAL e as categorias identificadas durante essa pesquisa e destaca uma lacuna na ausência de métricas na etapa de Initiate.

Assim, identificou-se métricas de Gestão Orçamentária (GO) que não se relacionam com todas as fases do TAL.

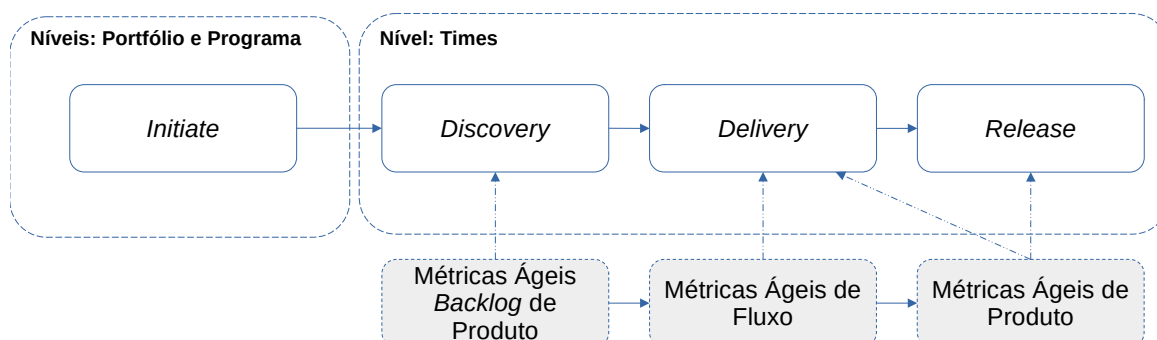


Figura 2 – Relacionamento entre as etapas do TAL e a categorização realizada na Tabela 5
Fonte: dados da pesquisa, 2022.

Conforme os achados das métricas da Tabela 8, a GO é compreendida por métricas como Custo do Atraso, Custo-Hora Real e Planejada, Desempenho no Projeto (Métricas Ágeis de Fluxo) e Projetos/Programas respeitando Custos e Orçamento (Métricas Ágeis do Produto). Conforme o PMI (2021), a gestão de projetos conta com três medidas básicas de gestão de custos, sendo o Custo Real (CR) comparado ao Custo Planejado (CP), Variação de Custo (*Cost Variance - CV*) e o Índice de Desempenho de Custo (*Cost Performance Index - CPI*). Assim, PMI (2021) determina que o CR comparado ao CP, trata-se de uma medida de custo que compara o custo realizado de mão de obra ou recursos com o custo estimado; o CV é uma variação de custo simples sendo determinada comparando o custo realizado de uma entrega com o custo estimado e, quando usado com gerenciamento de valor agregado, é a diferença entre o valor agregado e o custo real; e o CPI é uma medida de gerenciamento de valor agregado que indica a eficiência com que o trabalho está sendo executado em relação ao custo orçado do trabalho. Assim, entende-se a oportunidade de medir o custo planejado *versus* o realizado dos entregáveis do projeto por ciclos menores iniciando desde a execução de Iterações ou *Sprints*, dos entregáveis realizados por trimestre ou *release* pelo programa, e as entregas esperadas em nível de portfólio de projetos.

Grimaldi *et al.* (2016) apresentaram como uma estrutura ágil escalável, baseada em SAFe 3.0 e mostrou como implementar e medir o desempenho considerando que as funções e responsabilidades são divididas entre a organização e o fornecedor. Grimaldi *et al.* (2016) utilizaram métricas como eficácia e eficiência, custo-hora real e planejada, impedimentos e custo do atraso durante as rodadas de *Sprint*. No mesmo sentido que Ahmed *et al.* (2017) discutiram sobre as métricas *Sprint Burn Down*, *Epic Burn Down Velocity Chart*, *Control Chart* e a maneira como muitos dos cálculos de métricas são fortemente acoplados aos pontos de história para itens como o Épico ou *Sprint*.

Do ponto de vista de Petersen e Wohlin (2011), o *Lean* fornece ferramentas de análise e melhoria com foco no ciclo de vida do desenvolvimento, enquanto o ágil se concentra em soluções e prescreve conjuntos de práticas para obter agilidade. Petersen e Wohlin (2011) sugeriram análises sobre as métricas como reduzir o *lead time* para obter alta capacidade de resposta às necessidades dos clientes e fornecer um sistema de rastreamento que mostre o status do desenvolvimento de *software*. No sentido do monitoramento, as métricas de processo propostas por Choraś *et al.* (2020) foram utilizadas continuamente para ajudar a garantir a qualidade e estabilidade do *software* como a estimativa de tarefas e rastreamento tarefas em tempo real. Assim, pode-se identificar um *gap* sobre a ausência de métricas que aborde todas as etapas do projeto com objetivo de obter novas discussões sobre um processo de medição desde o portfólio, programa e time de projetos para trabalhos futuros é necessário. Esse contraponto sobre as métricas apresentadas pela Tabela 8 pode ser dividido em dois cenários dentro do contexto das práticas de gestão ágil de projetos: 1) escassez de propostas de métricas que venham medir as fases de *initiate*, *discovery*, *delivery* e a entrega contínua (*release*), mesmo que isoladamente, no desenvolvimento de um produto e 2) escassez de propostas de um processo de medição das etapas de *discovery*, *delivery* e entrega contínua para que assim seja possível medir e monitorar o projeto desde a iniciação até a entrega em partes (*Sprints*) e/ou no encerramento do projeto.

Com base na Tabela 5 e aos cenários 1 e 2 acima apresentados, verifica-se ainda a lacuna de processos de gestão ágil de projetos que contribuam como um *framework* de métricas ágeis capaz de ser um instrumento gerencial de maneira a orientar a liderança na gestão de projetos ágeis e para a tomada de decisão a partir de métricas estratégicas para a organização e para o desenvolvimento ágil de produtos. Os projetos ágeis que desenvolvem produtos, precisam testar hipóteses para resolver problemas organizacionais ou de clientes. Nesse sentido que o resultado oriundo das hipóteses deve ser medido para a tomada de decisão de refutar ou seguir adiante com a estratégia organizacional.

5 CONSIDERAÇÕES FINAIS

A partir da pesquisa realizada nas bases *Web of Science* e *Scopus*, a RSL encontrou-se 83 artigos que foram selecionados, dos quais apenas 42 abordavam aspectos relacionados a métricas de projetos tradicionais, 10 sobre as métricas ágeis de fluxo, 7 sobre as métricas ágeis de *backlog* do produto e 24 sobre as métricas ágeis de produto.

Apesar do histórico da adoção dos projetos tradicionais existir há décadas, o total dos artigos sobre as métricas em projetos tradicionais (40) encontradas nesse estudo não apresentaram um número tão maior que os resultados das métricas ágeis (43 artigos). Assim, percebeu-se que as métricas tradicionais focaram em apresentar a métrica do EVM com a adoção de melhorias na métrica, tal como melhorar a previsibilidade de riscos e alcançar o melhor valor agregado e o sucesso do projeto. Outras métricas muito citadas foram as adaptações inspiradas no EVM como o EDM e a utilização do SPI buscando medir o desempenho do cronograma.

Nas métricas ágeis, percebeu-se que as métricas se concentraram no *discovery* e no *delivery*, ou seja, em relação a medir o *backlog* do produto (9 artigos) no *discovery* e no fluxo e produto, tivemos 34 artigos no *delivery*. As métricas de *backlog* do produto se concentraram em medir a qualidade do requisito, enquanto as de produto, verificou-se uma grande concentração de métricas com preocupações com a qualidade do produto, ou seja, no tocante a quantidade de *bugs* resolvidos, tempo de resolução de problemas, qualidade do código-fonte e sobre a satisfação do cliente. Sobre as métricas de fluxo, identificou-se métricas como o *Lead Time*, *Velocity*, *Efficiency Flow*, *Cycle Time*, *Sprint Burn Down*, *Epic Burn Down*, *Control Chart*, entre outras. Assim, a Figura 2 apresenta uma lacuna de métricas que suporte um processo de gestão de projetos a partir do portfólio, programa e times de projeto e não apenas a gestão e métricas isoladas nas etapas de *Discovery*, *Delivery* e *Release*. Além disso, na fase de *Initiate*, pode-se identificar um *gap* sobre a ausência de métricas para a gestão de iniciativas para o portfólio de projeto. Por fim, entende-se com esse estudo que ainda existe uma lacuna de processos de gestão ágil de projetos que contribuam como um *framework* capaz de ser um instrumento gerencial com métricas estratégicas para a organização e para projetos de desenvolvimento ágil de produtos. Assim, projetos ágeis precisam testar hipóteses para resolver problemas organizacionais ou de clientes e precisam ser medidas para a tomada de decisão do que fazer a seguir.

Espera-se que este estudo possa contribuir com outras pesquisas que desejam identificar as métricas utilizadas entre as abordagens ágil e tradicional de gestão de projetos. Além disso, orientar novas pesquisas para trabalhos futuros no sentido de analisar uma possível lacuna entre na utilização de métricas tradicionais em projetos ágeis, que se compreende também em ser uma limitação dessa pesquisa. Além disso, é preciso estender a pesquisa para uma verificação em bases de patentes e na literatura cinzenta.

REFERÊNCIAS

- Abdelrahman Aljemabi, M. *et al.* (2020). Mining social collaboration patterns in developer social networks. *IET Software*, 14(7), 839-849.
- Ahmed, A. R. *et al.* (2017). Impact of story point estimation on product using metrics in scrum development process. *Intern. Journal of Advanced Computer Science and Applications*, 8(4).
- Aljedaibi, W. & Khamis, S.(2019).Towards Measuring the Project Management Process During Large Scale Software System Implementation Phase.The ISC Intern. Journal of Info. Security,11(3),161-172.
- Antinyan, V. (2014). Monitoring evolution of code complexity and magnitude of changes. *Acta Cybernetica*, 21(3), 367-382.
- Asha, N. & Mani, P. (2018). Knowledge-based Acceptance Test driven agile Approach for Quality Software Development. *International Journal of Resent Tech. and Engineering*, 7.
- Athanasiou, D. *et al.* (2014). Test code quality and its relation to issue handling performance. *IEEE Transac. on Software Eng*, 40(11),1100-1125.
- Azzeh, M.& Nassif, A. B. (2016). A hybrid model for estimating software project effort from Use Case Points. *Applied Soft Computing*, 49, 981-989.
- Bakhtary, V. *et al.* (2020).The effectiveness of test-driven development approach on software projects: A multi-case study.*Bulletin of Electrical Eng and Info.*,9(5),2030-2037.
- Batselier, J.& Vanhoucke, M. (2015). Evaluation of deterministic state-of-the-art forecasting approaches for project duration based on earned value management. *International Journal of Project Management*, 33(7), 1588-1596.
- Budacu, E. N.& Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, 22(4), 70-79.
- Chang, C. P. (2015). Software risk modeling by clustering project metrics. *International Journal of Software Engineering and Knowledge Engineering*, 25(06), 1053-1076.
- Chen, H. L. (2018). Early prediction of project duration: A longitudinal study. *Engineering Management Journal*, 30(3), 191-202.
- Chen, H. L. (2021). Early identification of distressed capital projects: A longitudinal approach. *International Journal of Managing Projects in Business*.
- Chen, H. L. (2014). Improving forecasting accuracy of project earned value metrics: Linear modeling approach. *Journal of Management in Engineering*, 30(2), 135-145.
- Chen, H. L., Chen, W. T.& Lin, Y. L. (2016). Earned value project management: Improving the predictive power of planned value. *International Journal of Project Management*, 34(1), 22-29.
- Chen, Z. *et al.* (2020).A Bayesian approach to set the tolerance limits for a statistical project control method. *International Journal of Production Research*,58(10), 3150-3163.
- Choraś, M. *et al.* (2020). Measuring and improving agile processes in a small-size software development company. *IEEE access*, 8, 78452-78466.
- Choudhury, I. (2019). *Agile Methods for Engineering*. In *Management for Scientists*. Emerald Publishing Limited.

- Costa, J., Vasconcelos, A.& Fragoso, B. (2020). An Enterprise Architecture Approach for Assessing the Alignment Between Projects and Goals. *International Journal of Information Technology Project Management (IJITPM)*, 11(3), 55-76.
- Colin, J. *et al.* (2015). A multivariate approach for top-down project control using earned value management. *Decision Support Systems*, 79, 65-76.
- Colin, J.& Vanhoucke, M. (2014). Setting tolerance limits for statistical project control using earned value management. *Omega*, 49, 107-122.
- Colin, J. *et al.* (2015). A comparison of the performance of various project control methods using earned value management systems. *Expert Systems with Applications*, 42(6), 3159-3175.
- Concas, G. *et al.* (2012). An empirical study of software metrics for assessing the phases of an agile project. *International Journal of Software Engineering and Knowledge Engineering*, 22(04), 525-548.
- Day, F. C. *et al.* (2019). Feasibility study of an EHR-integrated mobile shared decision making application. *International journal of medical informatics*, 124, 24-30.
- Dennehy, D.& Conboy, K. (2018). Identifying challenges and a research agenda for flow in software project management. *Project Management Journal*, 49(6), 103-118.
- Dias, K. R. S., & Larieira, L. C. C. (2021). Hybrid Project Management Method for managing ICT project's scope: a case study in a Brazilian company Método híbrido de gestão do escopo de projetos de TIC: estudo de caso em uma empresa brasileira. *Brazilian Journal of Development*, 7(8), 75984-76014.
- El Asmar, M. *et al.* (2013). Quantifying performance for the integrated project delivery system as compared to established delivery systems. *Journal of construction engineering and management*, 139(11), 04013012.
- Fleche, D. *et al.* (2017). Collaborative project: Evolution of computer-aided design data completeness as management information. *Concurrent Engineering*, 25(3), 212-228.
- Gao, K. *et al.* (2011). Choosing software metrics for defect prediction: an investigation on feature selection techniques. *Software: Practice and Exp.*, 41(5), 579-606.
- Geng, S. *et al.* (2018). Knowledge contribution as a factor in project selection. *Project Management Journal*, 49(1), 25-41.
- Glenwright, E. (2007). A survey of the 30 most serious flaws in scheduling. In *PMICoS 2007 Annual Conference*.
- Grau, D.& Back, W. E. (2015). Predictability index: Novel metric to assess cost and schedule performance. *Journal of Construction Eng. and Management*, 141(12), 04015043.
- Gren, L., Goldman, A.& Jacobsson, C. (2020). Agile ways of working: a team maturity perspective. *Journal of Software: Evolution and Process*, 32(6), e2244.
- Grimaldi, P. *et al.* (2016). An agile, measurable and scalable approach to deliver software applications in a large enterprise. *Intern. Journal of Agile Systems and Manag.* 9(4), 326-339.
- Han, S., Choi, J. O.& O'Connor, J. T. (2017). Quality of baseline schedules: Lessons from higher education capital facility projects. *Journal of Professional Issues in Engineering Education and Practice*, 143(1), 04016017.

- Hanna, A. S. (2016). Benchmark performance metrics for integrated project delivery. *Journal of Construction Engineering and Management*, 142(9), 04016040.
- Hanna, A. S. *et al.* (2012). Request for information: benchmarks and metrics for major highway projects. *Journal of construction engineering and management*, 138(12), 1347-1352.
- Hazır, Ö. (2015). A review of analytical models, approaches and decision support tools in project monitoring and control. *International Journal of Project Manag.* 33(4), 808-815.
- Hernández, G. *et al.* (2019). Productivity Metrics for an Agile Software Develop. Team: A Systematic Review. *TecnoLógicas*, 22(SPE), 63-81.
- Ibba, S. *et al.* (2018). Initial coin offerings and agile practices. *Future Internet*, 10(11), 103.
- Ibrahim, M. W. *et al.* (2020). Quantitative comparison of project performance between project delivery systems. *Journal of management in engineering*, 36(6), 04020082.
- Iqbal, S. *et al.* (2012). Yet another set of requirement metrics for software projects.
- Isong, B.& Obeten, E. (2013). A systematic review of the empirical validation of object-oriented metrics towards fault-proneness prediction. *International Journal of Software Engineering and Knowledge Engineering*, 23(10), 1513-1540.
- Jethani, K. (2013). Software metrics for effective project management. *International Journal of System Assurance Engineering and Management*, 4(4), 335-340.
- Junior, V. B. *et al.* (2019).Elaboração de um Plano de Gerenciamento de Projeto para Implantação de Laboratório de Metrologia Utilizando o Guia PMBOK 5ªEdição.Cippus,7(2),127-145.
- Jyothi, V. E.& Rao, K. N. (2017). Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Eng. Research and Science*, 4(3), 237085.
- Kamulegeya, G. *et al.* (2018). Measurements in the early stage software start-ups: a multiple case study in a nascent ecosystem. *Found. of Computing and Decision Sciences*,43(4),251-280.
- Kayes, I. *et al.* (2016). Product backlog rating: a case study on measuring test quality in scrum. *Innovations in Systems and Software Eng.* 12(4), 303-317.
- Kerkhove,L.P. & Vanhoucke, M.(2017).Extensions of earned value management: Using the earned incentive metric to improve signal quality. *Intern. Journal of Project Management*,35(2),148-168.
- Kristiansen, J. N.& Ritala, P. (2018). Measuring radical innovation project success: typical metrics don't work. *Journal of Business Strategy*.
- Kim, T., Kim, Y. W.& Cho, H. (2016). Customer earned value: performance indicator from flow and value generation view. *Journal of management in Eng.* 32(1),04015017.
- Ko, B. S.& Cho, M. S.(2020).Evaluating the relative efficiency of defense R&D projects in a multi-project environment using EVM and CAIV measures. *ICIC Express Letters, Part B: Applications*,11(7), 675-682.
- Kock, A.& Gemünden, H. G. (2019). Project lineage management and project portfolio success. *Project Management Journal*, 50(5), 587-601.

- Yamashita, A.& Counsell, S. (2013). Code smells as system-level indicators of maintainability: An empirical study. *Journal of Systems and Software*, 86(10), 2639-2653.
- Yousefi, N. *et al.* (2019). Using statistical control charts to monitor duration-based performance of project. arXiv preprint arXiv:1902.02270.
- Lappi, T. M. *et al.* (2019). Project governance and portfolio management in government digitalization. *Transforming Government: People, Process and Policy*, 13(2), 159-196.
- Mas, A. *et al.* (2020). Supporting the deployment of ISO-based project manag. processes with agile metrics. *Computer Standards & Interfaces*, 70, 103405.
- Mascarenhas Hornos da Costa, J., Oehmen, J., Rebentisch, E.& Nightingale, D. (2014). Toward a better comprehension of Lean metrics for research and product development management. *R&D Management*, 44(4), 370-383.
- Mäkiäho, P. *et al.* (2022). MMT: a tool for observing metrics in software projects. In *Research Anthology on Agile Software, Software Develop and Testing* (pp. 1077-1089). IGI Global.
- Meidan, A. *et al.* (2018). Measuring software process: a systematic mapping study. *ACM Computing Surveys (CSUR)*, 51(3), 1-32.
- Narbaev, T. *et al.* (2014). Combination of growth model and earned schedule to forecast project cost at completion. *Journal of Construction engineering and management*, 140(1), 04013038.
- Nidagundi, P.& Novickis, L. (2016). Introduction to Lean Canvas Transformation Models and Metrics in Software Testing. *Appl. Comput. Syst.*, 19(1), 30.
- Orgut, R. E. *et al.* (2018). Metrics that matter: Core predictive and diagnostic metrics for improved project controls and analytics. *Journal of Construction Engineering and Management*, 144(11), 04018100.
- Pajares, J.& Lopez-Paredes, A. (2011). An extension of the EVM analysis for project monitoring: The Cost Control Index and the Schedule Control Index. *International Journal of Project Management*, 29(5), 615-621.
- Parrend, P. *et al.* (2014, May). Swarm Projects: Beyond the Metaphor. In *International Conference on Swarm Intelligence Based Optimization* (pp. 131-138). Springer, Cham.
- Pavlova, N. *et al.* (2021). Creating the Agile Model to Manage the Activities of Project Oriented Transport Companies. *Eastern-European Journal of Enterprise Tech*, 1(3), 109.
- Perkusich, M. *et al.* (2017). Assisting the continuous improvement of Scrum projects using metrics and Bayesian networks. *Journal of Software: Evolution and Process*, 29(6), e1835.
- Petersen, K.& Wohlin, C. (2011). Measuring the flow in lean software development. *Software: Practice and experience*, 41(9), 975-996.
- Pradhan, S.& Nanniyur, V. (2021). Large scale quality transformation in hybrid development organizations—A case study. *Journal of Systems and Software*, 171, 110836.
- Prakash B. *et al.* (2018). Ontology based risks management model for agile software development. *Journal of Advanced Research in Dynamical and Control Systems*, 10(11 Special Issue):319–330.

- Pressman, R. S. & Maxim, B. R. (2021). *Engenharia de software-9*. McGraw Hill Brasil.
- Project Management Institute (PMI) (2021): A Guide to the Project Management Body of Knowledge (PMBOK Guide), PMI, Newtown Square, PA, 2021.
- Pozzana, I. *et al.* (2021). Spreading of performance fluctuations on real-world project networks. *Applied Network Science*, 6(1), 1-15.
- Rajablu, M. *et al.* (2017). Managing for stakeholders: Introducing stakeholder metrics-integrated model to lead project ethics and success. *International Journal of Project Organisation and Management*, 9(1), 31-56.
- Rajagopalan, J. *et al.* (2018). Introduction of a new metric “Project Health Index”(PHI) to successfully manage IT projects. *Journal of Organizational Change Management*.
- Ries, E. (2011). *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.
- Royce, W. (1970). The software lifecycle model(waterfall model). In *Proc. Westcon*, V.314.
- Sampaio, R. F. & Mancini, M. C. (2007). Systematic review studies: a guide for careful synthesis of the scientific evidence. *Brazilian Journal of Physical Therapy*, 11, 83-89.
- Sato, T. & Hirao, M. (2013). Optimum budget allocation method for projects with critical risks. *International Journal of Project Management*, 31(1), 126-135.
- Savola, R. *et al.* (2012). Risk-driven security metrics in agile software development: An industrial pilot study. *Journal of Universal Computer Science*, 18(12), 1679-1702.
- Sadler, H. (2020). ER2C SDMLC: enterprise release risk-centric systems development and maintenance life cycle. *Software Quality Journal*, 28(4), 1755-1787.
- Sajedi-Badashian, A. & Stroulia, E. (2020). Investigating the info. value of different sources of evidence of developers' expertise for bug assignment in open-source projects. *IET Software*, 14(7), 748-758.
- Sharma, V. *et al.* (2021). Development of metrics and an external benchmarking program for healthcare facilities. *International Journal of Construction Management*, 21(6), 615-630.
- Shawky, D. M. & Abd-El-Hafiz, S. K. (2016). Characterizing software development method using metrics. *Journal of Software: Evolution and Process*, 28(2), 82-96.
- Staron, M. & Meding, W. (2016). Mesram—a method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation. *Journal of Systems and Software*, 113, 76-100.
- Singh, J., Singh, K. & Singh, J. (2019). Reengineering framework to enhance the performance of existing software. *System*, 1139, 120-3.
- Simpson, J. R. *et al.* (2021). Optimizing the electronic health record: An inpatient sprint addresses provider burnout and improves electronic health record satisfaction. *Journal of the American Medical Informatics Association*, 28(3), 628-631.
- Spijkman, T. *et al.* (2021). Alignment and granularity of requirements and architecture in agile development: A functional perspective. *Information and Software Technology*, 133, 106535.
- Sommerville, I. (2007). *Eng. de Software*, 9. Ed. Pearson, Addison Wesley, 8(9), 10, p.44-45.

- Song, J., Martens, A.& Vanhoucke, M. (2021). Using schedule risk analysis with resource constraints for project control. *European Journal of Operational Research*, 288(3), 736-752.
- Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante, 2019.
- Tahir, T. *et al.* (2018). A systematic mapping study on software measurement programs in SMEs. *E-Informatica Software Engineering Journal*, 12(1).
- Tanveer, B. *et al.* (2019). An evaluation of effort estimation supported by change impact analysis in agile software develop. *Journal of Software: Evolution and Process*, 31(5), e2165.
- Tranfield,D. *et al.* (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review.*British journal of management*,14(3),207-222.
- Wallace, L. G.& Sheetz, S. D. (2014). The adoption of software measures: A technology acceptance model (TAM) perspective. *Information & Management*, 51(2), 249-259.
- Wauters, M.& Vanhoucke, M. (2015). Study of the stability of earned value management forecasting. *Journal of Construction Engineering and Management*, 141(4), 04014086.
- Van Casteren, W. (2017). *The Waterfall Model and the Agile Methodologies: A comparison by project characteristics*. Research Gate, 1-6.
- Vanhoucke, M. (2011). On the dynamic use of project performance and schedule risk information during projecttracking. *Omega*, 39(4), 416-426.
- Vanhoucke, M. (2019). Tolerance limits for project control: An overview of different approaches. *Computers & Industrial Engineering*, 127, 467-479.
- Üsfekes, Ç. *et al.* (2019). Auction-based serious game for bug tracking. *IET Software*, 13(5), 386-392.
- Zheng, M. *et al.* (2021). Key performance indicators for the integration of the service-oriented architecture and scrum process model for IOT. *Scientific Programming*, 2021.

5 ESTUDO 3: ESTUDO DE PATENTES SOBRE MÉTRICAS PARA A GESTÃO ÁGIL DE PROJETOS⁴

RESUMO

Em um contexto de novos *frameworks* e práticas para a gestão de projetos ágeis em constante mudança, a importância de realizar o monitoramento e controle de projetos por meio de métricas se faz necessário. São respostas às rápidas mudanças e a agilidade organizacional que precisam ser medidas em todo o ciclo de vida dos projetos. O objetivo desse artigo foi realizar uma pesquisa em base de dados de patentes de domínio público para identificar possíveis patentes sobre o desenvolvimento de um processo orientado a métricas ágeis. Como estratégia metodológica foi adotado uma análise exploratória com base em informações disponibilizadas nos bancos de dados de patentes em domínio público. Para a coleta de dados, foram utilizadas as ferramentas PATENTSCOPE, ESPACENET e LENS para o procedimento de pesquisa qualitativa exploratória em patentes. Os resultados apontaram a existência de patentes que propuseram métricas ágeis somente nas etapas de *delivery* e *release*, o que abre uma lacuna para outras etapas em projetos ágeis que precisam de medição para atender as necessidades da gestão de projetos ágeis.

Palavras-chave: Gestão Ágil de Projetos; Métricas Ágeis; *Agile Landscape*; Desenvolvimento de Software; *Discovery* e *Delivery*.

⁴Estudo publicado na *South American Development Society Journal*, v. 9 n. 27 (2023), disponível no Apêndice e em <https://sadsj.org/index.php/revista/article/view/663/521>.

1. INTRODUÇÃO

A abordagem ágil em gestão de projetos para o desenvolvimento de *software* tem crescido ao longo dos anos, desde o advento do Manifesto Ágil em 2001 (Agile Manifesto, 2001). Como Penha, da Silva e Russo (2020) destacaram, a gestão ágil de projetos inclui processos, técnicas e ferramentas de agilidade para introduzir o desenvolvimento contínuo e incremental de *software* de valor aos clientes e *stakeholders*. (Sutherland, 2019). Essa visão vai de encontro do modelo de desenvolvimento de *software* e de gestão tradicional com grandes etapas de planejamento somente no início do projeto. (Sommerville, 2007; Pressman, 2021). Conforme Sutherland (2019), Rubin (2017) e Adkins (2020), novas abordagens para gestão de projetos ágeis surgiram para suportar o processo incremental para entrega de *software*, tais como o *Extreme Programming*, *Scrum*, *Kanban*, *SAFe*, entre outras.

Em relação ao desempenho dos projetos desenvolvimento de *software*, Jyothi e Rao (2017) e dos Santos *et al.* (2018) destacam que na gestão ágil de tais projetos, o contexto muda para o valor entregue ao cliente ao invés de, segundo o *Project Management Institute* (PMI, 2017), as métricas tradicionais orientadas ao planejamento total do projeto, com destaque ao *Earned Value Management* (EVM). Nesse sentido, as métricas tradicionais utilizadas para apoio aos líderes podem ser consideradas inadequadas, ou se aplicadas fora do contexto de gerenciamento ágil de projetos de desenvolvimento de *software* (Budacu & Pocatilu, 2018), podendo causar impactos no controle e gerenciamento de tais projetos, como a visão tradicional de escopo fixo quando no ágil o escopo é estimado e a utilização do EVM em projetos ágeis quando o ágil considera o custo e o tempo são fixos. (Misra & Omorodion, 2011; Oza & Korkala, 2012).

Segundo Jyothi e Rao (2017) e dos Santos *et al.* (2018), em um ambiente ágil de projetos de desenvolvimento de *software*, as métricas se concentram, predominantemente, na etapa de *delivery* e *release* do projeto. Para Jyothi e Rao (2017), o *delivery* é uma etapa de desenvolvimento e entrega de um incremento de *software* e a *release* como uma etapa de entrega de uma versão estável para a produção, clientes e mercado. Budacu e Pocatilu (2018) destacaram que as métricas ágeis mais utilizadas são (a) *velocity*; (b) *Work Item Age* (WIA); (c) *Throughput*, (d) *Cycle Time* (CT) e (e) *Lead Time* (LT).

O *velocity* é uma métrica para avaliação utilizada no desempenho de entrega dos desenvolvedores de *software* por meio dos pontos de histórias do usuário. O WIA é uma métrica para medir o tempo em dias que um requisito está em uma etapa ou ciclo de

desenvolvimento de *software*. Já o *Throughput* representa uma métrica para avaliação da capacidade de entrega da equipe ao final de cada iteração. O CT conta o tempo em que um requisito está em um determinado ciclo ou etapas de desenvolvimento de *software*, enquanto o LT é considerado uma medida que avalia o tempo em que um requisito funcional é comprometido para início do desenvolvimento *software* até o mesmo ser entregue ao cliente, mercado ou *stakeholder*. (Budacu & Pocatilu, 2018). Em contrapartida, as métricas tradicionais se concentram na etapa de controle e monitoramento do projeto com objetivo de controlar o planejamento *versus* o realizado, como o EVM (PMI, 2017).

Apesar disso, as métricas calculadas apenas nas etapas de *delivery* e *release* não oferecem a visão de *Key Performance Indicators* (KPIs), o qual a liderança empresarial necessita para a tomada de decisão na gestão de projetos e estratégia organizacional (Jyothi & Rao, 2017). Em outras palavras, Budacu e Pocatilu (2018) apresentaram que o uso de tais métricas ainda é muito fragmentado em questões operacionais do projeto e que não garante que clientes, executivos da organização e *stakeholders* estejam alinhados entre valor percebido pelo cliente as metas da organização. Assim, a utilização de métricas apenas nas etapas de *discovery* e *delivery* não fornecem a visibilidade nas etapas que precedem a gestão de projetos e quando se busca colocar o cliente no centro no desenvolvimento de produtos de *software*.

Assim, o objetivo desse artigo foi realizar uma pesquisa patentária em base de dados de domínio público para identificar se existem patentes sobre o desenvolvimento de um processo orientado a métricas ágeis. Vale destacar que a pesquisa procurou identificar métricas em todas as etapas do modelo *The Agile Landscape – Deloitte* (TAL) (Paragano, 2021), que compreendem o *initiate*, *discovery*, *delivery* e *release*.

A pesquisa utilizou uma análise exploratória com base em informações disponibilizadas nos bancos de dados de patentes em domínio público (Creswell, 2014). Para a coleta de dados, foram utilizadas as ferramentas PATENTSCOPE, ESPACENET e LENS - *Patent Search and Analyse* para o procedimento metodológico aplicado na pesquisa qualitativa exploratória em patentes como sugerido por Paranhos e Ribeiro (2018). Os resultados obtidos permitiram a identificação de métricas ágeis presentes em patentes para atender demandas de organizações no contexto de medição em gestão ágil de projetos. Os resultados do mapeamento das métricas ágeis nas patentes podem contribuir para a proposição de modelos centrados em auxiliar as organizações na operacionalização de métricas ágeis, bem como ajudar as organizações a criarem processos orientados às métricas ágeis mais

adequadas para um contexto que envolve a entrega de valor para clientes e alinhamento organizacional e estratégico.

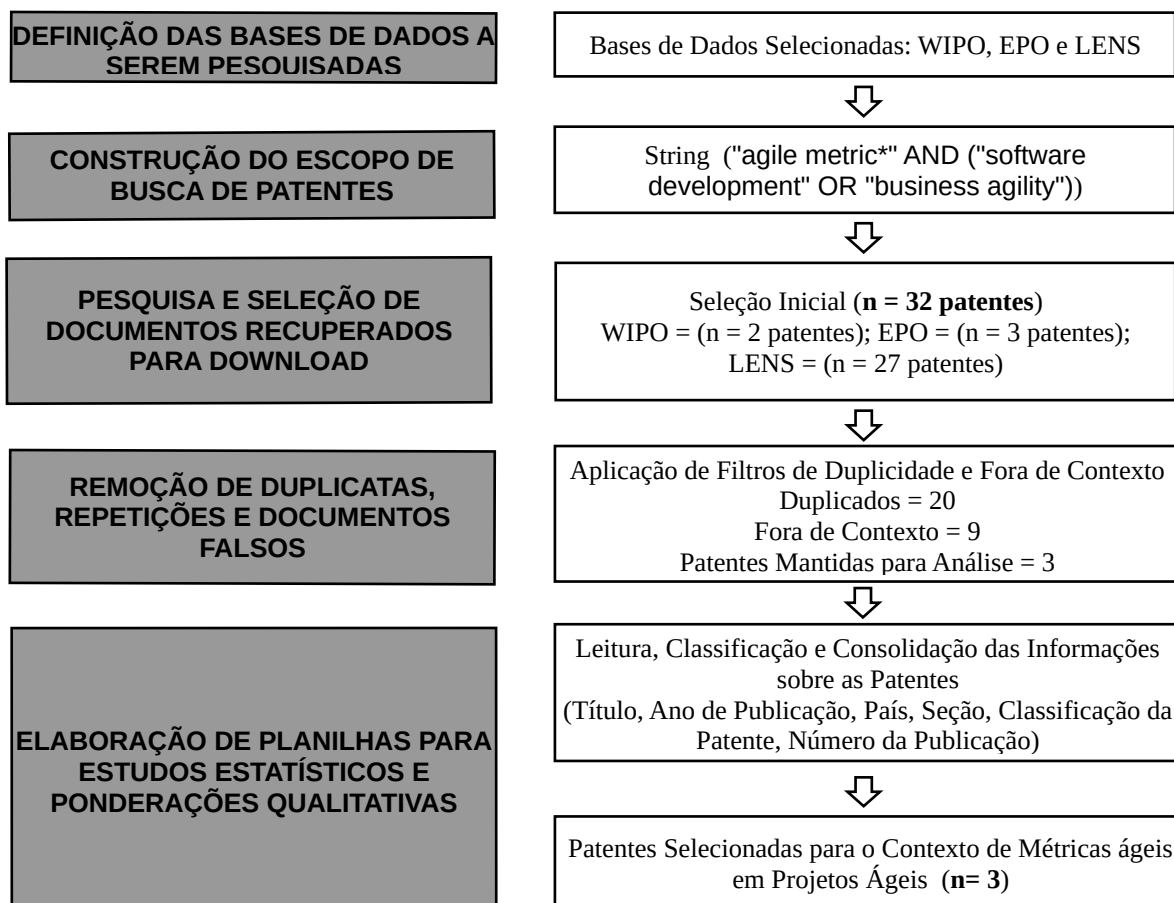
A próxima seção abordará a fundamentação teórica que norteia este estudo: etapas de concepção de requisitos, etapas de desenvolvimento de *software*, métricas ágeis e patentes. Em seguida, o método metodológico adotado será apresentado através de seus procedimentos, seguidos da apresentação dos resultados e, por fim, a conclusão do artigo com a apresentação das limitações e recomendações da pesquisa para estudos futuros.

2. MATERIAIS E MÉTODOS

Esta pesquisa utilizou a estratégia de abordagem qualitativa e exploratória visando compreender se existem estudos com patentes registradas em torno do uso de métricas em gestão de projetos ágeis. A abordagem metodológica desse estudo seguiu o proposto por Creswell (2014) por meio da pesquisa qualitativa explanatória. No campo da Administração, a pesquisa exploratória é comumente conduzida com propósitos profissionais e para subsidiar trabalhos acadêmicos. (Gil e Reis Neto, 2020).

No tocante ao mapeamento de patentes presente no protocolo metodológico a seguir, foram seguidos cinco passos conforme sugerido por Paranhos e Ribeiro (2018): (1) Definição das bases de dados a serem pesquisadas; (2) Construção do escopo de busca de patentes; (3) Pesquisa e seleção de documentos recuperados para download; (4) Remoção de duplicatas, repetições e documentos falsos; (5) Elaboração de planilhas para estudos estatísticos e ponderações qualitativas. O protocolo metodológico é representado na Figura 1 e apresenta todas as etapas do processo de Paranhos e Ribeiro (2018).

Figura 1. Procedimento metodológico aplicado na pesquisa qualitativa exploratória



Fonte: elaborado pelos autores conforme Paranhos e Ribeiro (2018).

Assim, a Figura 1 apresenta a primeira etapa que é descrita como a “Definição das Bases de Dados a Serem Pesquisadas” e, nessa etapa, foram selecionadas as bases *European Patent Office* (EPO), *World Intellectual Property Organization* (WIPO) e *Leans* (*Patent Search Service*) que são bancos de dados de patentes de conhecimento e domínio público. Como ferramental tecnológico foram utilizadas as ferramentas *PATENTSCOPE*, *ESACENET* e *LENS - Patent Search and Analysis* para a coleta de dados sobre as patentes.

Na segunda etapa (Figura 1) definida como “Construção do Escopo de Busca de Patentes”, foi definido o escopo de busca através da *string* de pesquisa (“*agile metric**” and (“*software development*” or “*business agility*”)) que foi executada nas bases de dados EPO, WIPO e LENS. Na próxima etapa (Figura 1) definida por “Pesquisa e Seleção de Documentos Seleccionados para *Download*” executada nas bases de dados WIPO, EPO e LENS, foram encontradas 32 patentes somadas com as três bases de dados, sendo duas patentes na base da WIPO, três patentes na base do EPO e 27 patentes na base do LENS. Em seguida, foi realizada a quarta fase (Figura 1) de “Remoção de Duplicatas, Repetições e

Documentos Falsos”, na qual foi verificada a ocorrência de patentes repetidas entre as bases e a remoção de duplicatas num total de 20 casos. Ainda nessa etapa (Figura 1), identificou-se nove casos fora do contexto da pesquisa, restando assim um total final de três patentes para continuidade da análise.

Na última fase (Figura 1), “Elaboração de Planilhas para Estudos Estatísticos e Ponderação Qualitativa”, os dados de três patentes foram importadas para uma planilha no LibreOffice e para a elaboração das tabelas para sequência da análise. Dessa forma, as informações importadas na planilha foram tratadas para realizar uma análise mais detalhada com objetivo de compreender a representação das patentes selecionadas. Foi realizada a consolidação das informações quantitativas e o agrupamento das informações por meio da *International Patent Classification* (IPC) que representa uma forma de classificação que utiliza letras e números em sua constituição para indexar patentes (WIPO, 2023).

Além dessa classificação, realizamos uma análise para determinar se as métricas apresentadas nas patentes atendiam o modelo TAL que considera o ciclo de iterações em quatro etapas definidas por *Initiate*, *Discovery*, *Delivery* e *Release*. A etapa de *Initiate* representa a concepção das ideias iniciais do projeto e que constitui a um *Backlog* do Produto inicial. (Sutherland, 2019). A etapa de *Discovery* representa o detalhamento dos requisitos, identificação de hipóteses de mercado a serem validadas, definição de jornada de usuário, protótipos e refinamento para ser objeto de entrada de uma iteração. A etapa de *Delivery* representa uma iteração onde são desenvolvidos os requisitos funcionais em forma de *software*. Por fim, a etapa de *Release* representa a liberação de uma versão estável de *software* ou um *Minimum Viable Product* (MVP) para clientes ou mercado.

A partir das abordagens de classificação das informações consolidadas e tratadas na planilha do LibreOffice, foram geradas tabelas e figuras para sintetizar os resultados e trazer à tona o entendimento das patentes selecionadas que serão apresentadas na próxima seção.

3. RESULTADOS

Esta seção apresenta os resultados do levantamento de patentes de conhecimento e domínio público. Inicialmente, é apresentado o mapeamento e a classificação das patentes que constituíram o corpus de análise da pesquisa. A seguir, são apresentadas as categorias destacadas após uma análise aprofundada do conteúdo das patentes.

3. 1 Mapeamento e Classificação de Patentes

As três patentes identificadas na pesquisa que abordam aspectos de métricas ágeis em gestão ágil de projetos em um período que vai de 2013 a 2023. Na análise dos resultados, foram localizados os países envolvidos no depósito das patentes em bases de dados do LENS, WIPO e EPO. Apesar do baixo número de patentes selecionadas neste estudo, os resultados mostram que os Estados Unidos possuem três patentes relacionadas com métricas ágeis e a China com apenas uma patente, conforme indicado na Tabela 1.

Tabela 1 - Classificação por IPC das patentes selecionadas nesse estudo

ID	Título	Ano de Publicação	País	Total de Patentes	Seção	Classificação da Patente	Número da Publicação	IPC
1	<i>System and method for computer development data aggregation</i>	2022	Estados Unidos	2	G	G06F	US202117501578A	G06F8/77
2	<i>Method and Apparatus for Calculating Performance Indicators</i>	2014	Estados Unidos		G	G06Q	US201213715708A	G06Q10/06
3	<i>GDASD task allocation method based on reliability</i>	2019	China	1	G	G06Q	CN110555613A	G06Q10/06

Fonte: elaborado pelos autores com base nos resultados da pesquisa.

Ao se observar a Tabela 1, é possível notar que as patentes estão classificadas nos grupos G06F e G06Q, sendo uma patente classificada no grupo G06F publicada nos Estados Unidos e duas patentes classificadas no G06Q, sendo uma patente publicada nos Estados Unidos e outra na China. Após identificar os países depositários de patentes que abordam o tema com métricas ágeis em projetos ágeis, incluindo como verificar a evolução do assunto ao longo dos anos e a demonstração das patentes de acordo com sua classificação. Dessa maneira, a próxima etapa visou avaliar, dentro da proposta do estudo, patentes que tratam de aspectos relacionados as métricas ágeis.

A Tabela 1 mostra que as patentes selecionadas estão concentradas em apenas uma seção de classificação G (física). Para contextualizar as patentes que compõem a seleção final, as características de cada patente foram analisadas quanto às métricas ágeis em gestão ágil de projetos. Dessa forma, a Tabela 2 destaca as patentes, apresenta o conceito e classificação

entorno das etapas segundo o modelo TAL e as métricas utilizadas em cada uma dessas etapas. Além disso, a Tabela 2 apresenta que essa pesquisa somente localizou patentes nas etapas de *Delivery* e *Release* (TAL).

Tabela 2 - Patentes selecionadas conforme *The Agile Landscape* (Deloitte)

ID	Título	Objetivo	Contexto		Métricas Delivery	Análise: Métricas Delivery	Métricas Release	Análise: Métricas Release
			Delivery	Release				
1	<i>System and method for computer development data aggregation</i>	Um sistema e método para integrar informações de desenvolvimento do projeto com base no ciclo de vida de desenvolvimento de software.	Aplica métricas de tempo de entrega, capacidade de entrega e velocidade de desenvolvimento de software.	Aplica métricas de entrega automatizada de software, de qualidade de código-fonte e quantidade de publicações de código-fonte no repositório.	<i>Lead Time, Velocity e Throughput.</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Fluxo onde foram classificadas as métricas da etapa operacional ou de desenvolvimento de software.	<i>Deploy automatizado, Code Quality e Pull requests.</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Produto onde foram classificadas as métricas relacionadas com a qualidade de software (produto) e com a cultura DevOps.
2	<i>Method and Apparatus for Calculating Performance Indicators</i>	Um método e aparelho para gerar indicadores de desempenho e melhorar o desempenho do time de desenvolvimento de software com base em métricas.	Aplica métricas de tempo médio de entrega do time de desenvolvimento e do tempo médio de ciclo nas etapas de desenvolvimento.	Não se Aplica	<i>Throughput Médio, Cycle Time e Cycle Time Médio.</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Fluxo onde foram classificadas as métricas da etapa operacional ou de desenvolvimento de software.	Não se Aplica	Não se Aplica
3	<i>GDASD task allocation method based on reliability</i>	Fornecer um método de alocação de tarefa com base na confiabilidade no problema de alocação de tarefas no desenvolvimento ágil distribuído remoto.	Não há métricas ágeis. Trata-se de aplicação de um método de alocação de tarefas com base na confiabilidade.	Não se Aplica	Não se Aplica	Não se Aplica	Não se Aplica	Não se Aplica

Fonte: elaborado pelos autores com base nos resultados da pesquisa.

Conforme a Tabela 2, a patente 1 apresentou as métricas ágeis nas etapas de *Delivery* e *Release*. A patente 2 apresentou somente as métricas ágeis na etapa de *Delivery* e, por fim, a patente 3 apresentou um método de alocação de tarefas em desenvolvimento ágil distribuído e remoto e, dessa forma, não apresentou relação alguma com o modelo TAL. Assim, a patente 1 apresentou as métricas relacionadas com a categoria de Métricas Ágeis de Fluxo (conforme o estudo 2) onde foram classificadas as métricas da etapa de desenvolvimento de software. Ainda na patente 1 temos as métricas relacionadas com a categoria de Métricas Ágeis de Produto (conforme o estudo 2), onde foram classificadas as métricas relacionadas com a qualidade de software e com as práticas de *DevOps* (Toh, Sahibuddin & Mahrin, 2019).

Em relação a adoção de métricas em projetos ágeis e apresentadas na Tabela 2, a etapa de *Delivery* contém as métricas de tempo de entrega, capacidade de entrega e da velocidade do time de desenvolvimento de *software*. Na etapa de *Release*, identificou-se as métricas de entrega automatizada de *software* (*DevOps*), de qualidade de código-fonte e quantidade de publicações de código-fonte no repositório de código.

Em relação as etapas do TAL de cada métrica ágil, a patente 1 apresentou as métricas ágeis de *Lead Time*, *Velocity* e *Througput* para a etapa de *Delivery* e as métricas ágeis de *Deploy Automatizado*, *Code Quality* e *Pull Requests* para a etapa de *Release*. A patente 2 apresentou as métricas de *Throughtput Médio*, *Cycle Time* e *Cycle Time Médio* para a etapa de *Delivery*. Por fim, a patente id 3 apresentou um método para calcular métricas de alocação de tarefas com base na confiabilidade e, assim, não se encaixa no contexto final dessa análise onde consideramos fundamental a classifica por meio do modelo TAL.

4. ANÁLISE E DISCUSSÃO

Os resultados apresentados pela Tabela 2 trouxeram apenas três patentes selecionadas para análise de métricas em projetos de desenvolvimento de *software*. Assim, somente duas patentes que apresentaram as métricas ágeis em projetos de desenvolvimento de *software* nas etapas de *Delivery* e *Release*, e com relacionamento com o modelo TAL. Além disso, as métricas desse estudo estão relacionadas com as categorias de Métricas Ágeis de Produto (conforme o estudo 2) e Métricas Ágeis de Fluxo (conforme o estudo 2) que são representadas pelo modelo TAL pelo *Delivey* e *Release*, respectivamente.

Assim, não foi possível encontrar um processo de medição contínuo nas bases de patentes de domínio público para todas as etapas do modelo TAL. Além disso, também não foi possível identificar alguma proposta que mensure um processo de desenvolvimento de *software*, como o *Scrum*. Por fim, não foram encontradas métricas para coletar o *feedback* do

cliente ou do mercado após o processo de *Release* do *software*. Tal questão leva a uma lacuna que sinaliza a importância de um processo de medição que contemple todas as etapas do modelo TAL e que possa suportar com métricas um processo de melhoria contínua para o *software* em desenvolvimento e/ou para as mudanças organizacionais, estruturais e na liderança através da organização que conduz o projeto.

5. CONCLUSÃO

Esta pesquisa teve como objetivo pesquisar nas bases de patentes de domínio público WIPO, EPO e LENS as possíveis patentes com propostas de métricas ágeis em gestão ágil de projetos. Na seleção inicial, foram encontradas 32 patentes para leitura e análise. Diante dos filtros aplicados, somente três patentes apresentaram métricas que abordassem, de maneira geral, o apoio ao desenvolvimento de *software* e aderência ao modelo TAL. Dentre as três patentes, somente duas apresentaram métricas ágeis que envolvessem as etapas de Fluxo de Valor e as etapas de *Delivery* e *Release*, sendo assim aderentes em partes ao modelo TAL. A terceira patente apresentou somente métricas de alocação de pessoas em atividades em um projeto ágil.

Não foram identificadas patentes que tratassem de métricas nas etapas de *Initiate* e *Discovery* conforme o TAL e que fossem específicas para obter *feedback* sobre o produto de *software* de valor e para validação de hipóteses do produto, clientes e mercado. Além disso, não foram identificadas as patentes que apresentassem métricas ágeis que refletissem o conhecimento obtido a partir do *feedback* do incremento (*release*) de *software* e de clientes (Sutherland, 2019), na validação de hipóteses (Gothelf, 2022; Ries, 2011) para confirmar ou refutar o valor de determinados requisitos para refletir diretamente no *Backlog* do Produto (Sutherland, 2019; Gothelf, 2022; Ries, 2011).

Também não foram encontradas métricas que sinalizassem mudanças necessárias nos times operacionais do projeto (Sutherland, 2019), na estrutura organizacional como os entregáveis não entregues por impedimentos não resolvidos (Kristensen & Shafiee, 2019), no portfólio de projeto como os itens do *backlog* do portfólio não entregues no trimestre corrente (PMI, 2017), e estratégia organizacional e de produto (Kristensen & Shafiee, 2019; Gothelf, 2022; Ries, 2011). Isso remete à necessidade de um processo de melhoria contínua que seja suportado por métricas de *feedback* de processo de desenvolvimento e de produto. Nesse sentido que o processo de melhoria proverá as melhorias embasadas em métricas em todas as etapas do TAL que podem auxiliar a organização no planejamento de mudanças nas estratégias organizacionais, no portfólio de projetos, nos programas e times de projeto. Tais

mudanças podem ajudar a organização no desenvolvimento de software que entregue o valor esperado por clientes e *stakeholders*. Por fim, não foram encontradas propostas de um processo de medição contínuo e incremental para a melhoria contínua do processo de desenvolvimento de *software* nas organizações nas etapas de *Initiate*, *Discovery*, *Delivery* e *Release* para entrega mais assertiva de valor aos clientes e para melhoria na estrutura organizacional, gestão do portfólio e estratégia organizacional.

Este estudo possui limitações que tange à pesquisa de processos de medição em gerenciamento de projetos ágeis que podem ser uma base sólida para a identificação das métricas mais indicadas para cada processo em um ambiente de agilidade de negócios de uma organização. Assim, a partir desse prisma, o processo de medição em nível organizacional de projetos pode ajudar gestores e líderes no alinhamento das estratégias organizacionais, em como trazer uma nova visão sobre como medir o valor entregue, além de abrir uma lacuna para estudos futuros.

REFERÊNCIAS

- ADKINS, Lyssa. Coaching agile teams: a companion for ScrumMasters, agile coaches, and project managers in transition. Pearson Education India, 2010.
- Agile Manifesto (2001). Manifesto for Agile Software Development. Disponível em <<http://agilemanifesto.org/>> Acesso em: 26 jun. 2023.
- BUDACU, Eduard Nicolae; POCATILU, Paul. Real Time Agile Metrics for Measuring Team Performance. *Informatica economica*, v. 22, n. 4, 2018.
- CRESWELL, John W. Research: Qualitative, quantitative, and mixed methods approaches. California. EUA: Sage, 2003.
- DA SILVA, Luciano Ferreira; RUSSO, Rosária de Fátima Segger Macri; DE OLIVEIRA, Paulo Sergio Gonçalves. Quantitativa ou qualitativa? um alinhamento entre pesquisa, pesquisador e achados em pesquisas sociais. *Revista Pretexto*, p. 30-45, 2018.
- DOS SANTOS, Paulo Sérgio Medeiros *et al.* On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *Journal of Software Engineering Research and Development*, v. 6, n. 1, p. 1-29, 2018.
- GIL, Antonio Carlos; DOS REIS NETO, Aline Crespo. Survey de experiência como pesquisa qualitativa básica em administração. *Ciencias da Administração*, v. 22, n. 56, p. 125-137, 2020.
- GOTHELF, Jeff. Lean UX: Applying lean principles to improve user experience. " O'Reilly Media, Inc.", 2013.
- JYOTHI, V. Esther; RAO, K. Nageswara. Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Engineering Research and Science*, v. 4, n. 3, 2017.

- KRISTENSEN, Saeedeh Shafiee; SHAFIEE, Sara. Rethinking organization design to enforce organizational agility. In: 11th Symposium on Competence-Based Strategic Management. 2019.
- MISRA, Sanjay; OMORODION, Martha. Survey on agile metrics and their inter-relationship with other traditional development metrics. ACM SIGSOFT Software Engineering Notes, v. 36, n. 6, p. 1-3, 2011.
- OZA, Nilay; KORKALA, Mikko. Lessons learned in implementing agile software development metrics. 2012.
- PARAGANO, Carmine. Agile Contracting, come gestire un contratto. PROJECT MANAGER (IL), n. 2021/46, 2021.
- PARANHOS, Rita de Cassia Santos; RIBEIRO, Núbia Moura. Importância da prospecção tecnológica em base em patentes e seus objetivos da busca. Cadernos de Prospecção, v. 11, n. 5, p. 1274, 2018.
- PENHA, Renato; DA SILVA, Luciano Ferreira; RUSSO, Rosária de Fátima Segger Macri. Escalando as práticas ágeis. Revista de Gestão e Projetos, v. 11, n. 2, p. 1-11, 2020.
- PMI. (2017). A Guide to the project management body of knowledge (6th edição). Disponível em <<https://doi.org/10.5860/CHOICE.34-1636>>. Acessado em: 25 de Jun. 2023.
- PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de software-9. McGraw Hill Brasil, 2021.
- RIES, Eric. The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses. Currency, 2011.
- RUBIN, Kenneth S. Essential Scrum: A practical guide to the most popular Agile process. Addison-Wesley, 2012.
- Sommerville, Ian. Engenharia de Software, 9. Edição. Pearson, Addison Wesley, 8(9), 10, p. 44-45, 2007.
- SUTHERLAND, Jeff. SCRUM: A arte de fazer o dobro de trabalho na metade do tempo. Leya, 2014.
- TOH, M. Zulfahmi; SAHIBUDDIN, Shamsul; MAHRIN, Mohd Naz'ri. Adoption issues in DevOps from the perspective of continuous delivery pipeline. In: Proceedings of the 2019 8th International Conference on Software and Computer Applications. 2019. p. 173-177.
- WIPO (2023). International Patent Classification (IPC). Disponível em: <<https://www.wipo.int/classifications/ipc/en/>>. Acessado em: 25 de Jun. 2023>.

6 ESTUDO 4: FMA: *FRAMEWORK* DE MÉTRICAS ÁGEIS COMO FERRAMENTA DE APOIO A GESTÃO ÁGIL DE PROJETOS

RESUMO

A mudança na mentalidade organizacional ao longo dos anos possui um fator importante no controle e monitoramento na gestão de projetos, principalmente no contexto do desenvolvimento de *software*. O objetivo desse estudo foi realizar a validação do *Framework* de Métricas Ágeis por meio de praticantes em gestão ágil de projetos. Foi utilizada a pesquisa qualitativa e metodologia empírica com o uso do *focus group* com praticantes de gestão de projetos em diferentes organizações. Assim, diversas alterações e contribuições trouxeram maior transparência, visibilidade e robustez ao *Framework* de Métricas Ágeis em diferentes organizações, níveis de maturidade e escala em projetos, além de se aderente às práticas ágeis como *Design Sprint*, *Lean Inception*, *Design Thinking*, *Scrum*, *Kanban*, *Extreme Programming* e *DevOps*. Por fim, limitações e trabalhos futuros foram identificados com o objetivo de explorar os custos de implementação e validar sua eficácia em diferentes organizações.

Palavras-chave: Gestão Ágil de Projetos; Métricas Ágeis; *Framework Ágil*; Desenvolvimento de Software; Integração e Entrega Contínua.

1 INTRODUÇÃO

A mudança do *mindset* tradicional de controle e monitoramento de projetos em times de desenvolvimento de *software* que estejam utilizando um *framework* ágil pode ser considerado um dos grandes desafios das organizações (Ries, 2011; Sutherland, 2019). O *mindset* de orientação ao planejamento, presente na gestão tradicional de projetos, conforme Sutherland (2019) e Ries (2011), quando aplicado aos projetos ágeis pode acarretar falhas no processo de tomadas de decisão, gerando impactos financeiros nas organizações. Na visão de Jyothi e Rao (2017) e dos Santos *et al.* (2018), na gestão ágil de projetos, o contexto da orientação ao planejamento muda para o valor entregue ao cliente. Além disso, e em se tratando no processo de controle e monitoramento dos projetos, as métricas se concentram, em sua maioria, na etapa de *delivery* do ciclo de vida do projeto ágil.

Conforme Jyothi e Rao (2017), o *delivery* é uma etapa do projeto ágil para o desenvolvimento e entrega de um incremento de *software*. Para os autores, as métricas calculadas apenas na etapa de *delivery* não oferecem a visão como instrumento gerencial para apoio a liderança, em sua visão corporativa, para a tomada de decisão na gestão ágil de projetos. Nesse sentido, Jyothi e Rao (2017) e dos Santos *et al.* (2018) destacam que os projetos ágeis são medidos por métricas como o *Cycle Time* e o *Lead Time*, ambas com a finalidade de avaliar o desempenho do projeto e do produto gerado para os *stakeholders*. Entretanto, a forma que se mede o desempenho nos projetos de desenvolvimento de *software* ainda é muito fragmentada em questões operacionais do projeto (Jyothi & Rao 2017). Isto é, medir o desempenho apenas durante a concepção e o desenvolvimento do projeto não garante que clientes, executivos da organização e *stakeholders* estejam olhando para a mesma perspectiva quando se trata de métricas, estratégias e as metas da organização (dos Santos *et al.* 2018). Como consequência, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as ágeis em gerenciamento de projetos de desenvolvimento de *software* (Huijgens *et al.*, 2017; Budacu & Pocatilu, 2018).

Assim, a maneira de medir o desempenho de tais projetos pode não ser adequada para que a organização e os *stakeholders* internos ou externos, tenham uma visão unificada do projeto dentro de um ambiente sob as práticas ágeis de gerenciamento de projetos (Dorninger & Ziebermayr, 2021). Além disso, a falta de métricas que contemplem as etapas de *Discovery*, *Delivery* e *Release* que venham a direcionar as estratégias de projeto, de produto e organizacionais para entrega de valor aos clientes e *stakeholders* de maneira mais rápida e confiável (Dorninger & Ziebermayr, 2021).

De tal modo, a utilização de ambientes de desenvolvimento e operações com foco na adoção de práticas ágeis e na busca pela melhoria contínua, entrega e implementação dos projetos de *software* e na transformação digital auxilia times no desenvolvimento de produtos digitais orientados ao valor de seus clientes e do mercado (Gokarna & Singh, 2021). Nesse contexto, a ausência de tais métricas e/ou a utilização de métricas de projetos tradicionais, dificultam a visão e o trabalho da liderança corporativa para tomar decisões fundamentais em projetos ágeis (Süß, Swift & Escott, 2022). Esse tipo de liderança pode ser definido como uma liderança multidimensional que começa com um desejo de servir (Greenleaf, 1998), seguido por uma intenção de liderar e desenvolver outros (Spears, 2010), para finalmente alcançar um nível mais elevado dos objetivos em benefício de indivíduos e das organizações (Van Dierendonck, 2011).

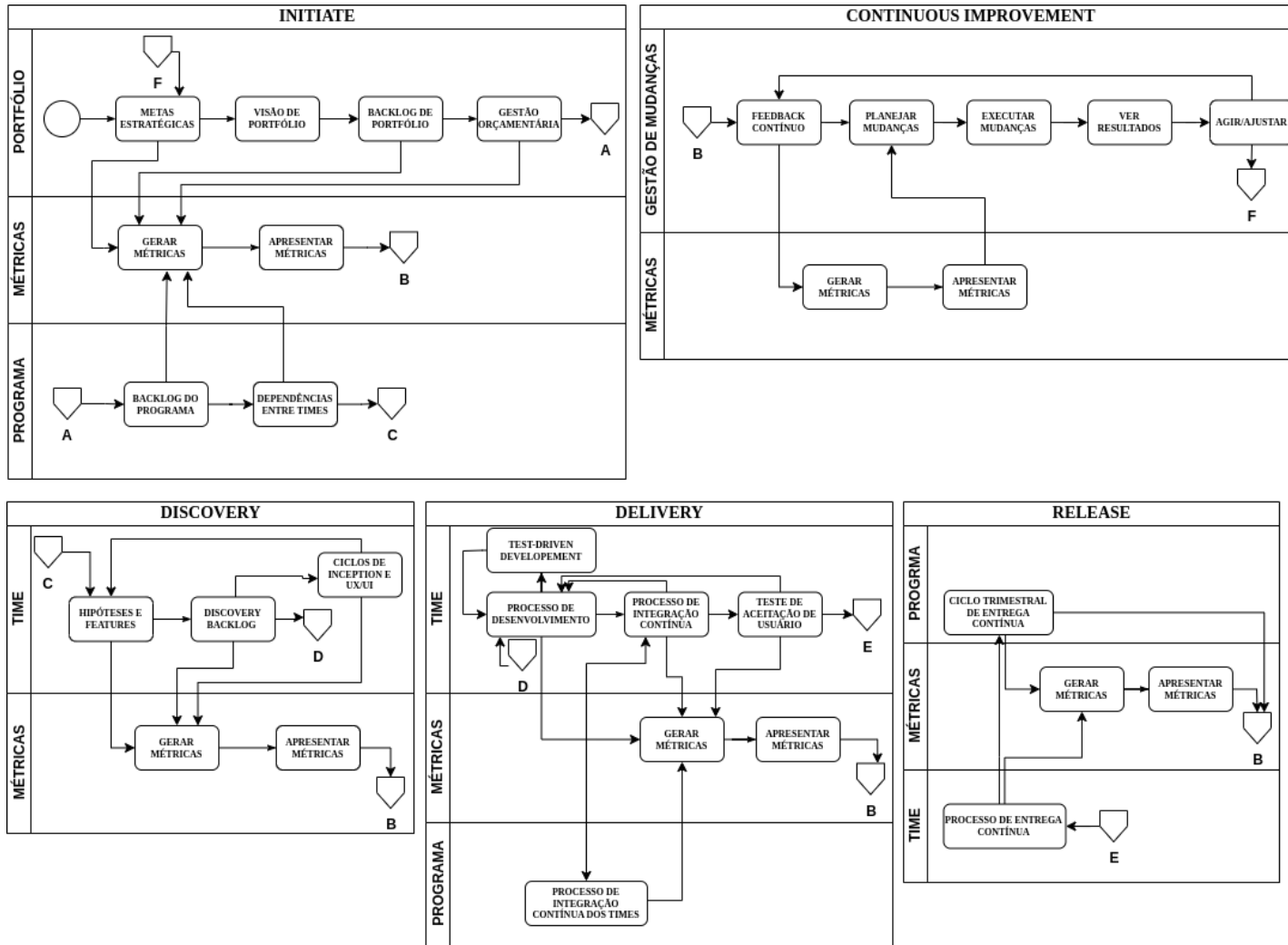
Nesse sentido e conforme Azanha Neto *et al.* (2022), a gestão e liderança em seus diferentes níveis de atuação dentro da organização e nos projetos, possuem a uma percepção difusa entre as práticas tradicionais e ágeis no gerenciamento de projetos de desenvolvimento de *software*. Ainda Azanha Neto *et al.* (2022) afirmaram que as práticas ágeis sugerem que os prazos das atividades informados aos líderes se baseiem em dados históricos de entrega, em vez de depender de um planejamento e controle das atividades após uma fase de iniciação do projeto conforme a gestão de projetos tradicional preconiza.

Assim, um *framework* de métricas ágeis que possa medir desde a concepção de metas organizacionais e portfólios, programas, esteiras de desenvolvimento e a melhoria contínua possa contribuir como uma ferramenta consistente para auxiliar os gestores em decisões importantes nas estratégias organizacionais e de produto. Assim, o objetivo desse estudo foi realizar a validação do *Framework* de Métricas Ágeis por meio de seções com praticantes de gestão de projetos em diferentes organizações.

2 **FRAMEWORK DE MÉTRICAS ÁGEIS**

A Figura 1 apresenta o *Framework* de Métricas Ágeis (FMA) que foi desenhado como uma possível ferramenta para ajudar times e líderes no controle e monitoramento de todas as etapas do gerenciamento de projetos em ambientes ágeis. O *framework* foi apresentado na seção 01 do FG em relação ao modelo *The Agile Landscape* (TAL)(Paragano, 2021). Assim, por se tratar de um fluxo de metas organizacionais e gestão orçamentária, desenvolvimento de produto e medição e *feedback* contínuo, o fluxo do FMA é um processo recursivo, pois seu fluxo é contínuo como se espera de qualquer *framework* ágil como o *Scrum*, além de ser retroalimentado pelas estratégias e metas da organização que o utiliza.

Figura 1. Proposta apresentada na seção 01 para o *Framework* de Métricas Ágeis



Fonte: elaborado pelo autor, 2023

O fluxo do FMA inicia na etapa de *Iniciate* com atividades na *lane* de *Portfólio* que elabora os itens importantes para o próximo trimestre que, uma vez definido o *backlog*, alimenta a *lane* de Programa onde se concentra gestores e times que conduzem os projetos dentro do trimestre. Assim, o Programa alimenta as etapas de *Discovery* e *Delivery* com as necessidades para o próximo trimestre, mas serão atendidas em iterações com duração de uma ou 4 semanas. A partir desse ponto, a fase de *Release* será alimentada com as entregas contínuas que fornecerá as entregas previstas com os *feedbacks* esperados. Desse modo, a etapa de *Release* irá alimentar o *feedback* do incremento para a etapa *Continuous Improvement*. Além disso, todas as etapas irão alimentar a etapa de *Continuous Improvement* com as métricas rastreáveis em cada etapa, viabilizando o *feedback* contínuo e o planejamento de mudanças em temas como produto, estratégias e metas organizacionais, gestão de projetos, agilidade organizacional, agilidade em projetos, novas métricas, no próprio FMA, entre outras.

O FMA é um *framework* que opera de forma integrada com times de projetos para o mesmo produto ou produtos que operam dentro da mesma iniciativa ou perspectiva de negócio. Dessa maneira, o FMA trabalha em períodos fundamentais de entrega como os períodos quinzenais (nível de time) e trimestrais (níveis de programa e portfólio). Esse modelo de entregas quinzenais e trimestrais sincronizados é uma abordagem para agilidade de negócios, pois permitem que as entregas sejam monitoradas e revisadas em ciclos curtos e permitindo que seja decidido em mudar a estratégia ou persistir com ela.

3 MATERIAIS E MÉTODOS

Este estudo é de abordagem qualitativa, caracterizado por um conjunto de procedimentos para a compreensão de um problema social (Thiollent, 2022). Uma pesquisa qualitativa permite estudar os contextos com maior proximidade, em uma interação entre pesquisador e objeto de pesquisa, permitindo também a utilização de paradigmas diferentes e técnicas de análise na investigação (da Silva, Russo & De Oliveira, 2018). De maneira geral, o grupo focal é utilizado na compreensão das diferentes percepções e comportamentos sobre de um evento, processo, prática, produto ou serviço e consiste justamente na interação entre os participantes e o pesquisador, que objetiva colher dados a partir da discussão focada em tópicos específicos e diretivos (Silva & Braga, 2023; Creswell, 2013).

Morgan e Krueger (1993) definem *focus group* (FG) como a técnica de análise e validação de dados de forma interativa controlada, que ajudam a compreender como e o porquê de as pessoas apresentarem determinados comportamentos e atitudes em determinadas situações.

Assim, foi entendida como entrevistas em grupo a conversa dirigida nas reuniões sem a finalidade de chegar à validação de algum ponto na discussão, mas sim para explorar ou compreender melhor alguns aspectos da pesquisa ou processo construído.

A pesquisa empírica foi conduzida em formato *online*, formato que possibilitou a realização de entrevistas com profissionais especializados que seria difícil de reunir fisicamente em um único local, visto o período de pandemia que a realização desta pesquisa está inserida. Pesquisas indicam que os participantes preferem a conveniência de grupos focais online devido à flexibilidade na programação e a capacidade de participar de casa ou do escritório (Zwaanswijk, & van Dulmen, 2014) e a distinção entre o mundo “real” e o mundo “virtual” está se tornando cada vez mais confusa (Stewart & Shamdasani, 2017). As sessões de FG ocorreram via ferramenta de comunicação *Google Meet*, que é utilizada internamente pela empresa, mediante consentimento dos líderes da empresa e dos entrevistados.

A amostra foi composta por 05 profissionais com experiência no gerenciamento de projetos ágeis e no processo de tomada de decisões por meio do uso de métricas ágeis, um entrevistador e um facilitador. A seção 01 foi composta por profissionais de gestão de projetos de sistemas corporativos e bancários, arquitetura e projetos de *software* e consultor. A seção 02 foi composta por profissionais de gestão de projetos de sistemas corporativos e bancários, arquitetura e projetos de *software*, projetos de consultoria para produtos digitais e consultor (Tabela 1). Os profissionais foram selecionados por terem contato direto com o controle e monitoramento de projetos por meio do uso de métricas ágeis em todas e por atuarem em todas as etapas do ciclo de vida de projetos ágeis.

Tabela 1. Participantes entrevistas e *focus group* (FG)

Código	Cargo	Descrição	Empresa
E1	<i>Product Owner</i>	Está na posição atual há 2,5 anos; líder o tema produtos demandando um time de 10 pessoas, sendo 1 <i>Scrum Master</i> e 9 desenvolvedores.	Instituição Financeira Brasileira.
E2	Consultor e Professor de Projetos	Está no cargo atual há mais de 15 anos com experiência em gestão de projetos há mais de 20 anos. É consultor e professor de tecnologia e de gestão de projetos. Possui certificado <i>Project Management Professional</i> (PMP) pelo <i>Project Management Institute</i> (PMI).	Instituição de Ensino Superior e de Pós-Graduação.
E3	<i>Head of Project Delivery Services</i>	Está na posição atual por 3 meses e com experiência como gerente e diretor de projetos por mais de 25 anos. No cargo atual está liderando diversos departamentos e times.	Empresa serviços de tecnologia e de desenvolvimento software de gestão corporativa.
E4	<i>Project Delivery Manager</i>	Está na posição atual por 2 anos liderando mais de 50 consultores atuando em diversos projetos de produtos digitais.	Empresa serviços de tecnologia e de desenvolvimento

			software para diversos setores da economia.
E5	Gestor de Arquitetura e Projetos de <i>Software</i>	Está no cargo há 2 anos com uma carreira consolidada em arquitetura e projetos de software há mais 11 anos em empresa brasileira de grande porte. Além disso, é cofundador de duas empresas de tecnologia para produtos digitais web e mobile, e professor de tecnologia em universidade.	Empresa de projetos e fabricação de balanças de precisão.

Fonte: elaborado pelo autor, 2023

Cada seção de grupo focal foi iniciada com uma breve apresentação do entrevistador e do facilitador e então o *framework* foi apresentado em relação ao modelo TAL. Como procedimento de coleta dos dados, foi utilizado um questionário não estruturado para dar liberdade aos entrevistados a comentarem e discutirem os pontos mais relevantes do *framework* e com isso os entrevistados puderam apresentar seus respectivos pontos de vista. Todas as sessões foram gravadas com consentimento dos entrevistados (Tabela 1). A seção 1 foi realizado no dia 19 de setembro de 2023 e teve uma duração de 90 minutos, enquanto a seção 2 foi realizada dia 03 de outubro de 2023 com duração de 45 minutos.

Como procedimento de análise dos dados, após cada grupo focal a gravação foi assistida com o propósito de revisar as sugestões e questionamentos realizado pelos entrevistados. Todos os apontamentos foram listados e discutidos entre o entrevistador e o facilitador de modo a chegar em um consenso sobre a aceitação da sugestão ou não. Então, as sugestões acatadas foram incorporadas ao *framework* apresentado na seção 2 e o próximo grupo focal foi realizado com o modelo revisado. Os participantes da seção 2 aprovaram as alterações realizadas e afirmaram que as alterações atenderam suas expectativas para a validação do *framework*.

Dessa maneira, esta pesquisa utilizou a estratégia de abordagem de metodologia qualitativa através da reunião de 05 profissionais de gestão ágil de projetos que representam o objeto de estudo através do grupo focal para obtenção dos dados sobre o *framework* de métricas apresentado como resultado desse estudo. O *framework* de métricas para a gestão ágil de projetos utilizou o modelo TAL para o desenho dos processos do *framework* que constitui as etapas de *Continuous Initiate*, *Continuous Discovery*, *Continuous Delivery* e *Continuous Release*. Por fim, o *framework* termina com a etapa de *Continuous Improvement* que retroalimenta a etapa de *Initiate* como um fluxo contínuo de gestão ágil de projetos.

4 RESULTADOS

A seguir são apresentados os resultados obtidos das sessões de FG utilizadas no processo de validação do *framework* (Tabela 2). O entendimento dos entrevistados sobre as métricas

ágeis e o modelo TAL apresentou convergência em diversos aspectos. De maneira geral, na seção 1 os participantes afirmaram que o FMA aborda temas importantes, principalmente no âmbito do modelo de ciclo trimestral, entrega e *feedback* contínua e a geração métricas em diversas etapas do projeto para necessidades e visões diferentes de cada público-alvo.

Tabela 2. Resultado das mudanças solicitadas pelos participantes da FG e suas contribuições ao FMA

Sugestão de Mudança	Assunto	Solicitante	Contribuição
Incluir <i>Continuous</i> nas demais etapas.	<i>Initiate</i> <i>Discovery</i> <i>Delivery</i> <i>Release</i>	E4	Padronizar a nomenclatura e manter a continuidade do fluxo do FMA
Categorizar as Métricas.	Métricas Ágeis	E1	Facilitar a identificação e a compreensão das métricas.
Rastrear Métricas	Métricas Ágeis	E2 e E3	Identificar a origem geração da métrica.
Aplicar o FMA em diferentes modelos e tamanhos de organizações.	FMA	E4	Ter a flexibilidade de implantar o FMA em diferentes organizações no mesmo projeto.
Aplicar o FMA em diferentes níveis de maturidade das organizações.	Nível de Time	E5	Aplicar o FMA apenas em nível de time com as etapas de <i>Continuous Discovery</i> , <i>Continuous Delivery</i> e <i>Continuous Release</i>
Aplicar o FMA em diferentes níveis de maturidade das organizações.	Nível de Programa e Portfólio	E5	Aplicar o FMA em nível <i>Continuous Initiate</i> e <i>Continuous Improvement</i> para as organizações com maior nível maturidade em gestão de projetos.
Aplicar o FMA em uma organização cliente (contratante) e em uma organização fornecedora (contratada).	FMA	E4	A organização cliente aplicaria o FMA em nível de <i>Continuous Initiate</i> e <i>Continuous Improvement</i> , enquanto a organização fornecedora adotaria as etapas de <i>Continuous Discovery</i> , <i>Continuous Delivery</i> e <i>Continuous Release</i> .
Visão de Produto.	<i>Continuous Initiate</i>	E1	Deve haver a visão de produto em nível de Programa para ajudar no engajamento dos times, entendimento das necessidades e mapeamento de dependências.
Etapas de <i>Continuous Improvement</i> retornar para as etapas.	<i>Continuous Initiate</i> <i>Continuous Discovery</i> <i>Continuous Delivery</i> <i>Continuous Release</i>	E2	Etapas de <i>Continuous Improvement</i> retornar para todas as etapas com o <i>feedback</i> e planos de ação para melhorias e ajustes nesta etapa do FMA.
Adoção de linhas pontilhadas no retorno de <i>Continuous Improvement</i> .	<i>Continuous Initiate</i> <i>Continuous Discovery</i> <i>Continuous Delivery</i> <i>Continuous Release</i>	E3	Adição de linhas pontilhadas para os retornos para representar que nem sempre haverá mudanças na determinada etapa, ou seja, opcionais ou por demanda.
Incluir práticas como <i>Design Sprint</i> ,	<i>Continuous</i>	E1	Ao incluir práticas como <i>Design Sprint</i> , <i>Lean</i>

<i>Lean Inception</i> e/ou <i>Design Thinking</i> .	<i>Discovery</i>		<i>Inception</i> e/ou <i>Design Thinking</i> durante a etapa de <i>Continuous Discovery</i> pode contribuir com práticas ágeis passíveis de serem adotadas junto ao FMA.
Planejamento da Iteração.	<i>Continuous Discovery</i>	E3	Deve haver o planejamento da iteração para representar o lote de entrega da iteração corrente.
Incluir práticas ágeis.	<i>Continuous Delivery</i>	E1	Incluir práticas como <i>Scrum Framework</i> , <i>Método Kanban</i> , <i>Extreme Programming</i> e <i>DevOps</i> fortalecerão o FMA.
<i>Contínuous Integration/Contínuous Delivery</i>	<i>Continuous Delivery</i>	E5	Incluir a prática <i>DevOps (Contínuous Integration/Contínuous Delivery)</i> agregará ao FMA práticas consolidadas nas organizações.
Ciclo Trimestral de Entrega Contínua.	<i>Continuous Release</i>	E5	O processo atual passaria uma visão de que a entrega contínua estaria condicionada ao fim de cada trimestre. Como a entrega da <i>release</i> não está condicionada a nenhum <i>timebox</i> , o processo foi renomeado para Ciclo Trimestral de Revisão da Entrega Contínua.
Incluir a Melhoria Contínua do Time.	<i>Continuous Release</i>	E3	Formalizar um processo de melhoria contínua em nível de time de projeto para tornar transparente no FMA esse procedimento.
Incluir a Organização no FMA.	Organização	E4	Contemplar a relação do FMA com a organização para demonstrar como o FMA se relaciona com a organização e como o <i>feedback</i> e as mudanças planejadas se comunicam com a organização.
Práticas OKR	Organização	E1 e E5	Incluir os <i>Objectives and Key Results (OKR)</i> na definição de metas estratégicas, pois OKR também monitora o andamento e entrega das metas trimestralmente.

Fonte: elaborado pelo autor, 2023.

Conforme a Tabela 2, os participantes também fizeram recomendações de ajustes no FMA visando ampliar a relevância e a aplicabilidade do *framework* nas organizações. Na seção 2, o FMA foi novamente apresentação com as recomendações implementadas no *framework* conforme a Tabela 2 para se obter o de acordo dos participantes.

4.1 Aplicabilidade do FMA

O primeiro ponto apresentado no FG aos entrevistados foi em relação à aplicabilidade do FMA. A aplicabilidade refere-se em como o FMA opera e a sua aplicabilidade nas organizações para a condução de seus projetos, sejam estes projetos conduzidos por um ou mais times de projeto sob iniciativas diferentes ou em nível escalado com múltiplos times de projeto. Assim, o FMA não prescreve em como a organização deve utilizá-lo e cabe a mesma essa decisão. Dessa forma, como o FMA é aplicável em empresas orientadas ou não em projetos,

conforme a Tabela 2, E4 destacou a importância de destacar a flexibilidade do FMA em diversos tipos de organizações. Para E5, o FMA pode ser aplicado em organizações que estiverem em diferentes estágios de maturidade em gestão de projetos. Nesse sentido que E5 sugeriu que o FMA pode ser aplicado em organizações que desejarem apenas o FMA em nível de time com as etapas de *Continuous Discovery*, *Continuous Delivery* e *Continuous Release*. E5 ainda considerou a aplicação do nível *Continuous Initiate* e *Continuous Improvement* para as organizações com maior nível maturidade em gestão de projetos. Nesse sentido também que E4 sugeriu que o FMA pode ser parte aplicado em uma organização cliente (demandante do projeto) e em uma organização fornecedora (fábrica de software). Assim a organização cliente aplicaria o FMA em nível de *Continuous Initiate* e *Continuous Improvement*, enquanto a organização fornecedora adotaria as etapas de *Continuous Discovery*, *Continuous Delivery* e *Continuous Release*.

4.2 Métricas Ágeis do FMA

O FMA oferece um processo contínuo de medições para o monitoramento, identificação de mudanças e oportunidades de melhoria e revisão de estratégias de produto, projeto e organizacional. Conforme a Tabela 2, E1 reforçou que o FMA precisa categorizar as métricas para facilitar a identificação e a compreensão das métricas. Além disso, E2 e E3 colocaram a necessidade de rastreabilidade das métricas para seja possível identificar a origem da geração da métrica. As sugestões de categorização e rastreabilidade das métricas do FMA proporcionará mais visibilidade e entendimento sobre as métricas geradas em cada etapa.

4.3 *Continuous Initiate*

A iniciação em projetos é uma etapa crucial, pois é nessa etapa onde concepções de projeto e estratégias e metas são alinhadas aos escopos de projeto. Nesse sentido que FMA considera em seu fluxo etapas de portfólio e programa com medições contínuas para um *feedback* rápido e contínuo. As métricas visam compreender o estado atual das metas, estratégias, itens de portfólio e de programa.

Conforme a Tabela 2, E1 entende que deve haver a visão de produto em nível de Programa, assim como o FMA considera a visão de portfólio. Assim, E1 destaca a importância da visão de produto vai auxiliar no engajamento dos times por compreender o produto que será desenvolvido, além de ajudar no mapeamento de dependências para os times envolvidos na etapa de *Initiate* para o Programa.

Conforme sugestão de E4 a inclusão da palavra *Continuous* dará o sentido de ciclo contínuo que o *framework* pretende oferecer como um fluxo contínuo de *feedback* e aprendizado.

Já E2 destacou a importância do fluxo da etapa de *Continuous Improvement* retornar para etapa de *Continuous Initiate* com o *feedback* e planos de ação para melhorias e ajustes nesta etapa do FMA. Nesse caso, uma observação foi adicionada para representar que o fluxo é opcional, ou seja, quando houver mudanças a serem transmitidas para a etapa de *Continuous Improvement*.

4.4 *Continuous Discovery*

O FMA enfatiza que o uso da descoberta como uma oportunidade de aprendizado e entendimento sobre o produto de *software* e oferece um processo simples para realizar a descoberta e detalhamento de requisitos de negócio e requisitos através de um *backlog* bem estruturado e um ciclo de *inception* e UX/UI. Além disso, as métricas realizam a rastreabilidade da saúde do *backlog* e sobre o estado dos itens do *backlog*. Assim, dada a relevância da etapa de *Continuous Discovery*, a Tabela 2 apresenta que E3 entende que deve haver o planejamento da iteração para representar o lote de entrega da iteração corrente e isso oferecerá maior visibilidade através de um evento no FMA.

E2 destacou a importância do fluxo da etapa de *Continuous Improvement* retornar para etapa de *Continuous Discovery* com o *feedback* e planos de ação para melhorias e ajustes nesta etapa do FMA. Nesse caso, uma observação foi adicionada para representar que o fluxo é opcional, isto é, quando houver mudanças a serem transmitidas para a etapa de *Continuous Discovery*.

Conforme sugestão de E4 a inclusão da palavra *Continuous* dará o sentido de ciclo contínuo que o *framework* pretende com uma visão de fluxo contínuo de *feedback* e aprendizado em todas as etapas do FMA.

Dado que o FMA pode ser utilizado com as melhores práticas ágeis, E1 sugeriu de incluir práticas como *Design Sprint*, *Lean Inception* e/ou *Design Thinking* durante a etapa de *Continuous Discovery* e potencializando o fluxo de processos e medições do FMA.

4.5 *Continuous Delivery*

A etapa de *Continuous Delivery* do FMA possui processos de desenvolvimento baseado em testes, integração e entrega contínua, além de testes de aceitação de usuário. As métricas nessa etapa apresentam resultados medidos em tempo de execução do ciclo de desenvolvi-

mento e medidas de desempenho e dados empíricos sobre as entregas realizadas. Assim, a Tabela 2 apresenta E2 que destacou a importância do fluxo da etapa de *Continuous Improvement* retornar para etapa de *Continuous Delivery* com o *feedback* e planos de ação para melhorias e ajustes nesta etapa do FMA. Nesse caso, uma observação foi adicionada para representar que o fluxo é opcional, ou seja, quando houver mudanças a serem transmitidas para a etapa de *Continuous Delivery*. Conforme sugestão de E4 a inclusão da palavra *Continuous* dará o sentido de ciclo contínuo que o *framework* pretende.

Dado que o FMA pode ser utilizado com as melhores práticas ágeis para que possam potencializá-lo, E1 sugeriu de incluir práticas como *Scrum Framework*, Método *Kanban*, *Extreme Programming* e *DevOps (Continuous Integration/Continuous Delivery)* durante a etapa de *Continuous Delivery*. Neste mesmo sentido que E5 sugeriu de incluir a prática *DevOps (Continuous Integration/Continuous Delivery)* durante a etapa de *Continuous Delivery*. Estas sugestões como adições potencializarão o FMA com as práticas, há muito tempo, consolidadas no mercado e nas organizações.

4.6 Continuous Release

O fluxo de *Continuous Release* visa o lançamento contínuo de um *software* de valor com lançamentos a qualquer momento e revisões trimestrais do agregado dos lançamentos do trimestre. As métricas, por sua vez, identificam oportunidades com base nos lançamentos realizados, além de servir como *baseline* entre o planejado versus realizado nas etapas de *Initiate* e *Discovery*. Assim, Tabela 2 destaca E5 com a preocupação com o processo Ciclo Trimestral de Entrega Contínua na etapa de *Release* que passaria uma visão de que a entrega contínua estaria condicionada ao fim de cada trimestre. Assim, E5 trouxe importância de a entrega da *release* não estar condicionada a nenhum *timebox* e o processo foi renomeado para Ciclo Trimestral de Revisão da Entrega Contínua. Assim, o FMA se torna mais evidente que o processo de entrega contínua não precisa estar ligado a um *timebox*, isto é, a entrega pode ocorrer em qualquer momento do projeto dentro do trimestre corrente.

Apesar da possibilidade de o FMA ser utilizado com o *Framework Scrum*, o Método *Kanban* ou qualquer prática ágil, E3 sugeriu incluir um processo de melhoria contínua em nível de time de projeto.

Conforme sugestão de E4 a inclusão da palavra *Continuous* para a etapa de *Release* dará o sentido de ciclo contínuo de *feedback* e aprendizado que o *framework* pretende.

E2 destacou a importância do fluxo da etapa de *Continuous Improvement* retornar para etapa de *Continuous Release* com o *feedback* e planos de ação para melhorias e ajustes nesta

etapa do FMA. Nesse caso, uma observação foi adicionada para representar que o fluxo é opcional, ou seja, quando houver mudanças a serem transmitidas para a etapa de *Continuous Release*.

4.7 Continuous Improvement

O *Continuous Improvement* é uma etapa que visa exclusivamente obter as métricas de todas as etapas e realizar um processo de planejamento, execução e revisão de ações para a mudança plena e contínua em nível organizacional e/ou em nível de projeto. Conforme a Tabela 2, E1 e E2 colocaram que seria interessante padronizar a nomenclatura com a palavra *continuous* nas demais etapas, principalmente com a proposta de E2 da etapa de *Continuous Improvement* retornar com as métricas e *feedback* para as demais etapas.

E2 destacou a importância do fluxo da etapa de *Continuous Improvement* retornar para cada etapa com o *feedback* e planos de ação para melhorias e ajustes em cada etapa do FMA. E3 sugeriu a adição de linhas pontilhadas para os retornos para representar que nem sempre haverá mudanças na determinada etapa, ou seja, opcionais ou por demanda.

4.8 Organização

A Tabela 2 apresenta E4 que colocou a importância do desenho do FMA contemplar a sua relação com a organização para demonstrar como o FMA se relaciona com a organização e como o *feedback* e as mudanças planejadas comunicam com a organização. Esse fluxo do FMA para a Organização ajudará a mesma no redesenho de suas estratégias e metas organizacionais, pois poderá trabalhar com um cenário novo de *feedback* e melhorias identificadas através do fluxo contínuo de *feedback* e aprendizado que as métricas proporcionam através de seus dados.

Conforme as orientações de E1 e E5 com relação a utilização de práticas ágeis através do FMA, os *Objectives and Key Results* (OKR) podem ser utilizados na definição de metas estratégicas, pois OKR também monitora o andamento e entrega das metas trimestralmente. Assim, a utilização de OKR potencializará o FMA com uma prática consolidada em grandes organizações como o Google.

4.9 Feedback dos Participantes

Os resultados obtidos foram apresentados aos participantes através da seção 2 do FG. Os participantes reconheceram que todas as alterações sugeridas foram atendidas e que o FMA oferece contribuições importantes aos praticantes no campo da agilidade e da gestão ágil de projetos.

5 ANÁLISE E DISCUSSÃO

Mediante a aplicação da pesquisa através da FG com os praticantes em gestão ágil de projetos, a seção 1 contou com 5 participantes que apresentaram suas sugestões de ajuste do modelo do FMA apresentado pelo entrevistador e, culminando na seção 2, foram apresentadas as alterações conforme as sugestões da seção 01 e que foram totalmente aplicadas e absorvidas ao FMA após a análise realizada pelo entrevistador e facilitador.

Conforme a Figura 2, as sugestões foram aplicadas no modelo do FMA. O FMA inicia com a Organização como um agente importante ao funcionamento do FMA, pois é a responsável pela geração de insumos necessários para o FMA, tais como a definição de projetos e metas estratégicas ao longo do ano ou do trimestre corrente. As etapas do FMA foram renomeadas com a inclusão da palavra *continuous* nas etapas de *Initiate*, *Discovery*, *Delivery* e *Release* para assim promover uma visão de fluxo contínuo. Além disso, o *feedback* de *Continuous Improvement* para a Organização com métricas e plano de mudanças para fortalecer a revisão de projetos e metas estratégicas.

Para a etapa de *Continuous Initiate*, a Figura 2 apresenta essa fase sendo alimentada pela Organização que mantém e revisa novos insumos para os próximos trimestres de projeto. Além disso e, em nível de Programa, foi incluída o retorno de *Continuous Improvement* e a Visão de Produto após o mapeamento de Dependências entre Times do Programa. A visão de produto fornecerá um entendimento compartilhado com todos os times sobre o que se pretende desenvolver no trimestre.

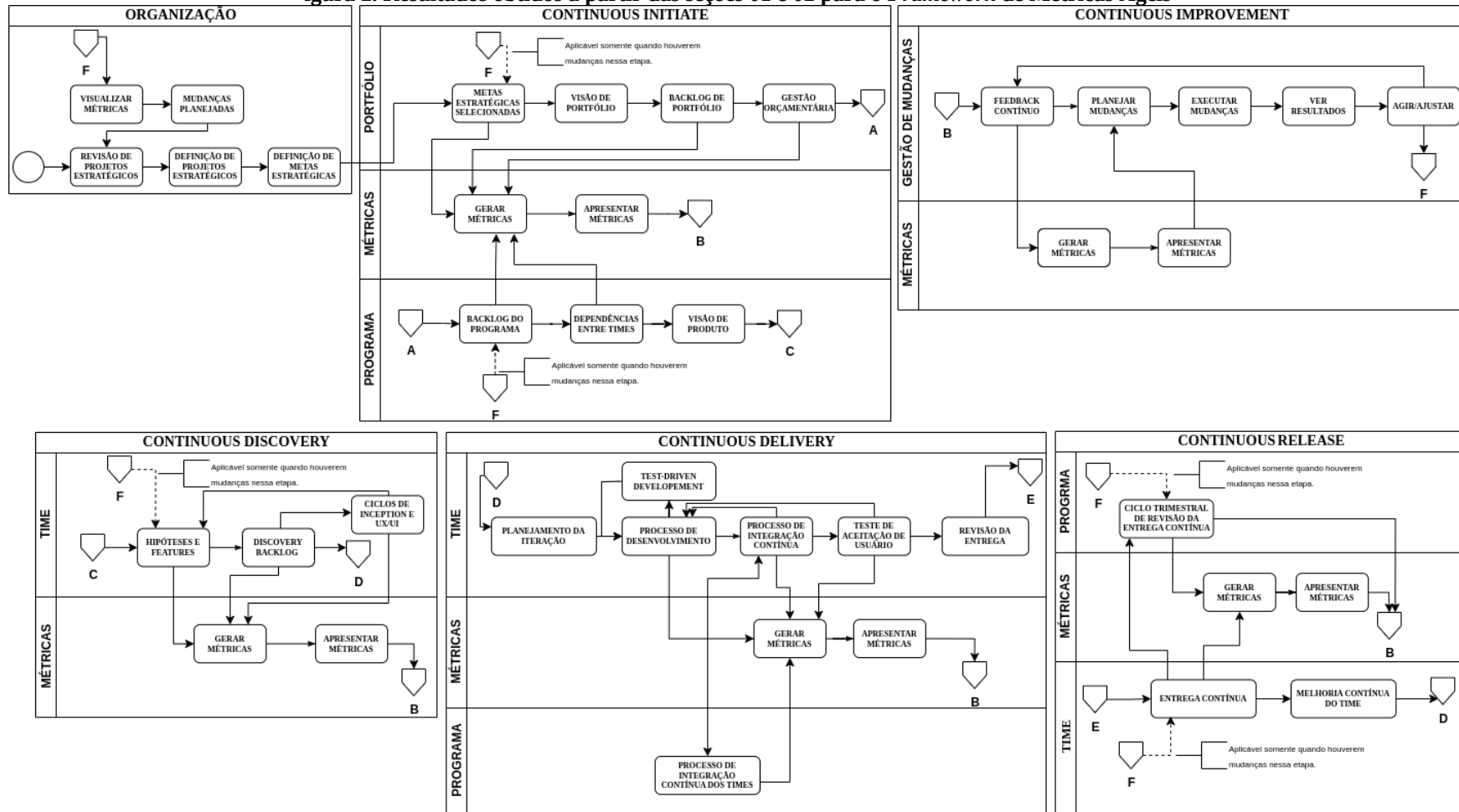
A Figura 2 também apresenta o retorno de *Continuous Improvement* na etapa de *Continuous Delivery* que pode trazer métricas de clientes e mercado sobre os requisitos entregues, teorias e hipóteses validadas que podem refletir no artefato do *Discovery Backlog*.

Para *Continuous Release*, a Figura 2 apresenta mudanças significativas para melhorar a visibilidade do FMA para uma abordagem de CI/CD do *DevOps*. Assim, *Continuous Release* sofreu mudanças para representar que os times realizam entregas contínuas sem a imposição de algum *timebox* específico, ou seja, cada time entrega diariamente,

semanalmente e assim por diante. Nesse sentido que o processo de Ciclo Trimestral de Entrega Contínua passa um sentimento de que a entrega ocorrerá somente a cada trimestre. Assim, a mudança para Ciclo Trimestral de Revisão da Entrega Contínua, mostra uma visão de que o processo de Entrega Contínua segue com entregas a qualquer momento e o Ciclo Trimestral de Revisão da Entrega Contínua apresenta uma revisão das entregas feitas no trimestre que é uma importante ferramenta de gestão e revisão de métricas. Por fim, em *Continuous Release* teve a inclusão da Melhoria Contínua do Time para que o time faça um evento de reflexão e lições aprendidas.

Por fim, os praticantes não sugeriram mudanças na etapa de *Continuous Improvement*. Desse modo, a Figura 2 não apresenta mudanças nessa etapa.

figura 2. Resultados obtidos a partir das seções 01 e 02 para o *Framework* de Métricas Ágeis



Fonte: elaborado pelo autor, 2023.

A Figura 2 apresenta uma evolução do FMA em relação a Figura 1 para trazer melhorias importantes para a visibilidade do FMA como uma ferramenta de gestão ágil de projetos com entrega, *feedback*, planos de mudança e métricas em todas as etapas de maneira contínua e com visibilidade. As alterações e contribuições destacadas pelos participantes do *focus group* contidas na Tabela 2 trouxeram maior transparência, visibilidade e robustez no FMA no tocante ao processo de validação de hipóteses contínua, lançamento contínuo, *feedback* contínuo e melhoria contínua. Além disso, as contribuições identificaram oportunidades de implementação do FMA para diferentes tipos de organização.

6 CONCLUSÃO

A pesquisa por meio do FG foi utilizada como técnica de análise e validação de dados. Assim, a amostra foi composta por 05 profissionais com experiência no gerenciamento de projetos ágeis e no processo de tomada de decisões por meio do uso de métricas ágeis que constam relacionados na Tabela 1.

Após a seção 1, os resultados das alterações e contribuições sugeridas pelos participantes foram estruturados na íntegra na Tabela 2. Assim, contribuições como padronizar a palavra *continuous* em todas as etapas, categorização e rastreabilidade das métricas, aplicabilidade do FMA em diferentes tamanhos de organização, posicionamento no projeto (cliente e fornecedor) e níveis de maturidade da organização. Além disso, a Tabela 2 trouxe novos processos para artefatos e eventos importantes, como a Visão de Produto, Planejamento da Iteração e Melhoria Contínua do Time. Como o FMA é um *framework* para *feedback* e melhoria contínua baseada em métricas, a sugestão de retornar *Continuous Improvement* para todas as etapas do FMA potencializa a melhoria contínua e o uso de métricas para revisão do FMA, do produto em desenvolvimento e das estratégias organizacionais.

A Tabela 2 também trouxe sugestões importantes para potencializar o uso do FMA a partir de práticas ágeis conhecidas e consolidadas, tais como *Scrum*, *Kanban*, *OKR*, *Design Thinking*, *DevOps*, entre outras. A Seção 2 foi utilizada para reunir novamente os participantes para apresentação das alterações e contribuições com objetivo de se obter a validação deles, onde todos concordaram com as alterações realizadas.

Esta pesquisa contribui tanto praticamente quanto academicamente para o campo do desenvolvimento de *software* e gestão de projetos. No nível prático, fornece *insights* sobre o uso de práticas ágeis como *Scrum* e *Kanban*, bem como outras práticas como *OKR*, *Design Thinking* e *DevOps*, para melhorar o uso de métricas ágeis nas organizações. Também oferece

uma perspectiva sobre o papel da liderança em projetos ágeis, enfatizando sua importância na execução bem-sucedida do projeto. Assim, espera-se que esse *framework* de métrica possa suportar com uma ferramenta que ofereça uma visão consolidada de métricas ágeis para apoiar a gestão em diversos níveis de avaliação de metas estratégicas, ideias e hipóteses, escopo de produto, gestão de projeto e gestão ágil de produto.

Academicamente, esta pesquisa contribui para o corpo existente de conhecimento, fornecendo uma revisão sistemática da literatura sobre liderança em projetos ágeis. Também apresenta um estudo de síntese estruturado sobre os benefícios e desafios do uso do *Kanban* em engenharia de *software*. Além disso, explora o potencial da engenharia ágil orientada a modelos usando cadeias de ferramentas *DevOps* e discute a revisão histórica e os trabalhos futuros do *DevOps*. Além disso, espera-se que esse trabalho de tese possa contribuir com a evolução de estudos acadêmicos no desenvolvimento da agilidade na gestão de projetos nas diferentes organizações, na agilidade de negócios (*business agility*) e na medição em diferentes etapas do projeto ágil.

As limitações desse estudo estão em torno da pesquisa sobre o custo para a implementação do FMA em uma organização, como também mensurar o custo de eventos, papéis e artefatos do *framework*. Como trabalhos futuros, a sugestão seria de aplicar na prática o FMA em organizações diferentes para validar o *framework*.

REFERÊNCIAS

- Azanha Neto, J. D. S., Penha, R., da Silva, L. F., & Scafuto, I. C. (2022). The importance of leadership in agile projects: systematic literature review. *Research, Society and Development*, 11(5), e44511528117-e44511528117.
- Budacu, E. N., e Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, 22(4), 70-79.
- Creswell, John W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications. Cap. 5 – 6*
- Da Silva, L. F., Russo, R. D. F. S. M., & De Oliveira, P. S. G. (2018). Quantitativa ou qualitativa? um alinhamento entre pesquisa, pesquisador e achados em pesquisas sociais. *Revista Pretexto*, 30-45.
- Dorninger, B., & Ziebermayr, T. (2021). Software-Qualitätssicherung im Maschinen-und Anlagenbau: automatisierte Bewertung der technischen Qualität von SPS-Code. *e & i Elektrotechnik und Informationstechnik*, 138(6), 315-320.
- Dos Santos, P. S. M., Beltrão, A. C., de Souza, B. P., e Travassos, G. H. (2018). On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *Journal of Software Engineering Research and Development*, 6(1), 1-29.
- Gokarna, M., & Singh, R. (2021, February). DevOps: a historical review and future works. In *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 366-371). IEEE.

- Greenleaf, R. K. (1998). *The power of servant-leadership*. Berrett-Koehler Publishers.
- Huijgens, H., Lamping, R., Stevens, D., Rothengatter, H., Gousios, G., & Romano, D. (2017, August). Strong agile metrics: mining log data to determine predictive power of software metrics for continuous delivery teams. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering* (pp. 866-871).
- Jyothi, V. E., e Rao, K. N. (2017). Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Engineering Research and Science*, 4(3), 237085.
- Ries, E. (2011). *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.
- Silva, C. R., & Braga, J. L. (2023). Gestão do conhecimento potencializando a gestão de riscos de projetos de desenvolvimento de software ágil utilizando o SCRUM. *Fronteiras da Representação do Conhecimento*, 1(1), 116-140.
- Spears, L. C. (2010). Character and servant leadership: Ten characteristics of effective, caring leaders. *The journal of virtues & leadership*, 1(1), 25-30.
- Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante, 2019.
- Süß, J. G., Swift, S., & Escott, E. (2022). Using DevOps toolchains in Agile model-driven engineering. *Software and Systems Modeling*, 21(4), 1495-1510.
- Paragano, C. (2021). Agile Contracting, come gestire un contratto. *PROJECT MANAGER (IL)*, (2021/46).
- Stewart, D. W., & Shamdasani, P. (2017). Online focus groups. *Journal of Advertising*, 46(1), 48-60.
- Thiollent, M. (2022). *Metodologia da pesquisa-ação*. Cortez editora.
- Morgan, D. L., & Krueger, R. A. (1993). When to use focus groups and why.
- Zwaanswijk, M., & van Dulmen, S. (2014). Advantages of asynchronous online focus groups and face-to-face focus groups as perceived by child, adolescent and adult participants: a survey study. *BMC research notes*, 7(1), 1-7.
- Van Dierendonck, D. (2011). Servant leadership: A review and synthesis. *Journal of management*, 37(4), 1228-1261.

7 PROCEDIMENTOS METODOLÓGICOS PARA PRODUTO TECNOLÓGICO

A natureza da presente pesquisa se caracteriza como uma pesquisa aplicada e dirigida à solução de problema específico. Uma atitude consciente e crítica com relação aos procedimentos e paradigmas de pesquisa podem gerar avanços importantes às áreas de conhecimento (Silva, Russo, e Oliveira, 2018). Assim, na próxima seção, será apresentada o desenho da pesquisa como resultado do desenvolvimento dessa pesquisa de tese resultante de quatro estudos realizados e publicados.

7.1 DESENHO DA PESQUISA

O desenho da pesquisa que será apresentado a seguir foi dividido, segundo o modelo do TAL, em um desenho do *Framework* de Métricas Ágeis (FMA) com cinco figuras contendo: a) fluxo da etapa de *Initiate* para o portfólio e programa (atualmente conhecido também tribo), e o relacionamento dessa etapa com a Organização; b) fluxo da etapa de *Discovery* para times (*squads*) de projeto; c) fluxo da etapa *Delivery* para programa (tribo) e times de projeto; d) fluxo da etapa de *Release* para o programa e times de projeto e; e) o desenho segue além das etapas do modelo TAL e apresenta o fluxo para a etapa de *Continuous Improvement* para a gestão de mudanças que retroalimentará o FMA com as mudanças necessárias mediante as métricas apresentadas nas etapas do TAL. Assim, as etapas do FMA foram definidas, a partir do Estudo 4, por *Continuous Initiate*, *Continuous Discovery*, *Continuous Delivery*, *Continuous Release* e *Continuous Improvement*.

Na sequência, serão apresentadas as cerimônias, artefatos e papéis que envolvem e norteiam todo o processo do FMA, além de seu relacionamento com outros *frameworks* e práticas ágeis consolidadas nas empresas como o *Framework Scrum*, Método *Kanban*, *Design Thinking*, entre outros.

O FMA é um *framework* que opera de forma integrada com times de projetos para o mesmo produto ou produtos que operam dentro da mesma iniciativa ou perspectiva de negócio que entrega valor para um ou mais clientes ou mercados onde produtos e serviços operam. Dessa forma, o FMA trabalha em períodos fundamentais de entrega (*timebox*) como os períodos quinzenais (nível de time) e trimestrais (níveis de programa e portfólio). Esse modelo de entregas quinzenais e trimestrais sincronizados é uma abordagem para a gestão ágil de projetos e para a agilidade de negócios (*business agility*), pois permitem que as entregas

sejam monitoradas e revisadas em ciclos curtos e permanentes, permitindo que seja decidido pelos *stakeholders* em mudar a estratégia do produto ou serviço, ou persistir com a estratégia para validar novas premissas, ideias e hipóteses de negócio.

Conforme a terminologia do modelo TAL, as iniciativas dentro de um projeto de desenvolvimento de *software* são compreendidas por *features*, os múltiplos projetos inter-relacionados ou não como programa. Assim, o FMA utilizará as terminologias como *Features* e Programas.

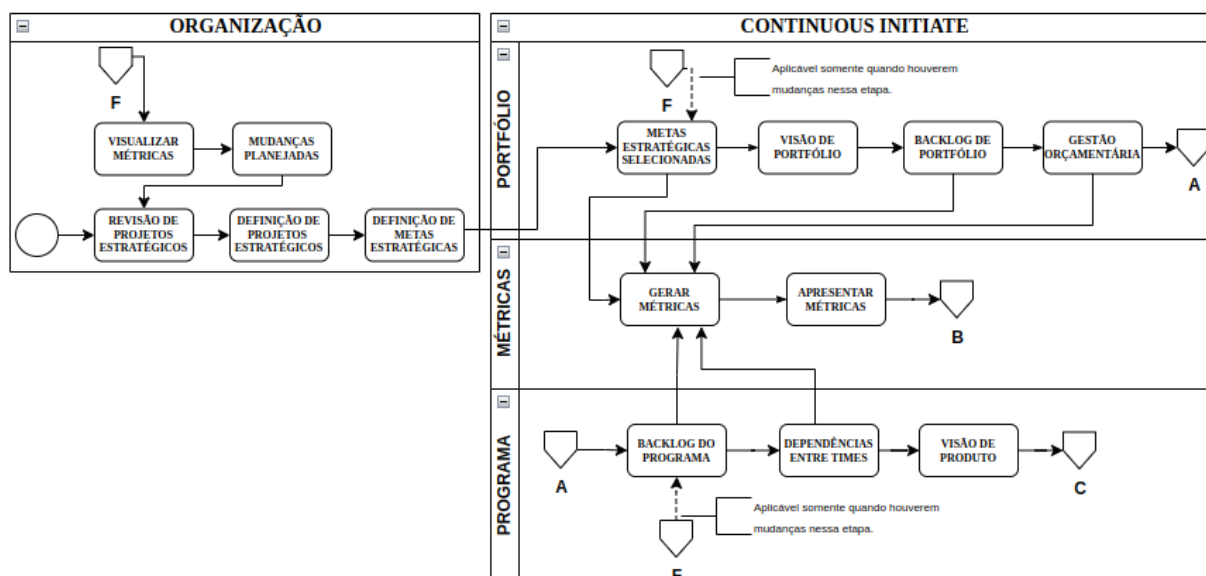
A seguir as etapas do FMA serão discutidos separadamente para uma maior compreensão e depois será apresentado o desenho completo. Além disso, combinado a cada parte do FMA serão apresentados os papéis relacionados com cada etapa para um melhor entendimento dos papéis interessados.

7.2 CONTINUOUS INITIATE

O desenho da pesquisa representado pela Figura 2 apresenta o fluxo da etapa de *Initiate* do TAL adaptado para *Continuous Initiate* para o portfólio, programa e métricas. Assim, a Figura 2 inicia com as Metas e Projetos Estratégicos que são definidos e revisados pela Organização durante os ciclos trimestrais e que alimenta a etapa de *Continuous Initiate*. Tais projetos e metas podem estar relacionadas com mudanças organizacionais necessárias que seriam suportadas através de *softwares* corporativos ou por uma organização de projetos de *software* que constrói serviços ou produtos de *software* para o mercado, tais como sistemas embarcados, sistemas corporativos, sistemas web, cliente-servidor e aplicações móveis. Uma vez definidas as metas e projetos, *Continuous Initiate* inicia as Metas Estratégicas Seleccionadas e a Visão de Portfólio contém as metas e projetos aprovados para um ou mais ciclos trimestrais de projetos. Assim, metas e projetos aprovados são organizados e agrupados em *features* e, posteriormente na etapa de *Continuous Discovery*, as *features* agrupam as histórias de usuário para as entregas quinzenais.

O *Backlog* do Portfólio, por sua vez, seleciona todos as *features* aprovadas para o próximo ciclo trimestral e ordenado ou priorizado para que sejam executados conforme o valor de negócio a ser entregue primeiro. Assim, essas *features* são estimadas pela Gestão Orçamentária com os custos para a execução de cada *feature* no trimestre para um conjunto de entregáveis do projeto amarradas em cada *feature*. Assim, cada time de projeto terá um conjunto de *features* orçadas e planejadas para serem trabalhadas no próximo trimestre.

Figura 2 - Fluxo da etapa *Continuous Initiate* para a proposta de *Framework* de Métricas Ágeis (FMA)



Fonte: elaborado pelo autor, 2023.

Nos processos de Metas Estratégicas, *Backlog* de Portfólio e Gestão Orçamentaria são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente. Posteriormente serão apresentadas as métricas sugeridas para esses processos em um tópico separado.

O fluxo segue para o nível de Programa onde se constrói o *Backlog* do Programa que será trabalhado ao longo do ciclo trimestral do projeto de desenvolvimento por todos os times ágeis de projeto envolvidos. Esse nível Programa também pode ser conhecido por Tribo por algumas organizações que contém um conjunto de times ágeis de projeto e atendem a uma certa iniciativa ou produto. Além disso, os times ágeis de projeto envolvidos nas *features* do Programa se reúnem através do processo de Dependências entre Times para fazer o mapeamento de todas as dependências técnicas e/ou funcionais, os riscos envolvidos e para organizarem e priorizarem as *features* a serem desenvolvidas primeiro no trimestre. Além disso, o mapeamento de dependências é uma oportunidade para a identificação de riscos e oportunidades que possam atrapalhar o trabalho das equipes dentro do Programa. O resultado desse mapeamento de dependências culminará na Visão de Produto que fornecerá um entendimento compartilhado sobre os macros objetivos, o que se pretende e o que não se pretende com o produto. Além disso, essa Visão de Produto fornecerá aos times de projeto envolvidos no Programa um maior engajamento na direção do desenvolvimento de *software* centrado no cliente e entregando o valor esperado por ele.

Nos processos de *Backlog* do Programa e Dependências entre Times são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente. Posteriormente serão apresentadas as métricas sugeridas para esses processos em um tópico separado.

A Tabela 2 apresenta os papéis e frentes de trabalho relacionadas com a etapa de *Continuous Initiate*. As frentes de trabalho envolvidas nessa etapa são Portfólio e Programa.

Tabela 2- Papéis do Framework de Métricas Ágeis (FMA) na etapa de *Continuous Initiate*

Papel	Frente de Trabalho	Etapa do FMA
<i>Agile Metrics Owner</i> (AMO)	Portfólio/Programa	<i>Continuous Initiate</i>
<i>Agile Portfolio Owner</i> (APO1)	Portfólio	
<i>Agile Program Owner</i> (APO2)	Programa	
<i>Agile Architecture Owner</i> (AAO)		
<i>Agile Business Manager</i> (ABM)		

Fonte: elaborado pelo autor, 2023.

O *Agile Metric Owner* (AMO) é o responsável por manter e apresentar nas devidas cerimônias as métricas que foram coletadas durante a fase de *Continuous Initiate*. O *Agile Portfolio Owner* (APO1) é o responsável em receber as estratégias da organização e, através delas, manter o *Backlog* do Portfólio e questões orçamentárias do Portfólio. O *Agile Program Owner* (APO2) é o responsável pela orientação das demandas dentro do Programa e o *Agile Business Manager* (ABM) é o responsável pela orientação do negócio dentro do Programa buscando planejar e definir produtos e serviços para clientes. O *Agile Architecture Owner* (AAO) é o responsável pela definição e desenho de arquitetura de *software* numa visão corporativa e de arquitetura de soluções no Programa. Assim, detalhamentos de arquitetura de *software* serão preocupações na etapa de *Continuous Delivery*.

7.3 CONTINUOUS DISCOVERY

O desenho da pesquisa representado pela Figura 3 apresenta a etapa de *Discovery* do modelo TAL adaptado para *Continuous Discovery* que inicia com as Hipóteses e *Features* selecionadas para o ciclo trimestral do projeto de desenvolvimento de *software*. Essas *features* formarão o *Discovery Backlog* para o trimestre que deverá ser priorizado com base no valor e nas dependências mapeadas conforme a etapa de *Continuous Initiate* anterior a essa etapa.

Assim, inicia-se na etapa de *Continuous Discovery* os Ciclos de *Inception* e UX/UI (*User Experience/User Interface*) para detalhamento dos requisitos em Histórias de Usuário (HU), jornadas do usuário, *personas* e interfaces de usuário que trarão novas descobertas sobre o produto. Esses resultados poderão influenciar em priorização ou até o descarte de Hipóteses e *Features* e no *Discovery Backlog*. Em ambos os processos, são definidos os artefatos orgânicos que mudam conforme o conhecimento sobre o problema e necessidade de usuário aumentam, o que poderá resultar na mudança de priorização dos itens do *Discovery Backlog* ou até mesmo a eliminação do item por não fazer mais sentido seguir com ele.

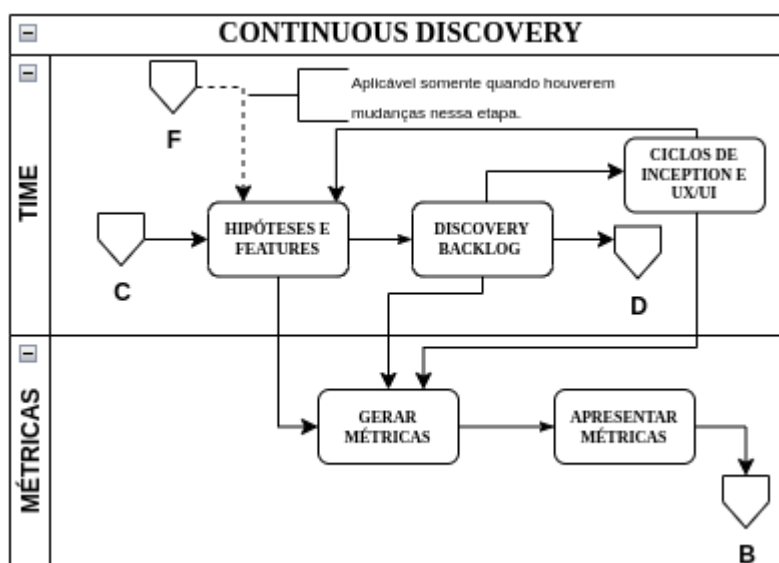


Figura 3 - Fluxo da etapa *Continuous Discovery* para a proposta de Framework de Métricas Ágeis (FMA)
Fonte: elaborado pelo autor, 2023.

Nos processos de Hipóteses, *Features*, *Discovery Backlog* e Ciclos de *Inception* e UX/UI são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente. Posteriormente serão apresentadas a métricas por etapa.

A Tabela 3 apresenta os papéis e frentes de trabalho relacionadas com a etapa de *Continuous Discovery*. A frente de trabalho envolvida nessa etapa é o Time de Projeto.

Tabela 3- Papéis do Framework de Métricas Ágeis (FMA) na etapa de *Continuous Discovery*

Papel	Frente de Trabalho	Etapa do FMA
Agile Business Owner (ABO)	Time de Projeto	<i>Continuous Discovery</i>
Agile Coach (AC)		
Agile Software Developer (ASD)		

Fonte: elaborado pelo autor, 2023.

O *Agile Business Owner* (ABO) é o responsável pela gestão do *Delivery Backlog* com a escrita das histórias de usuário e sua priorização com base no valor. Além disso, o ABO deve suportar os *Agile Software Developers* (ASDs) em questões de negócio durante o trimestre corrente. O *Agile Coach* (AC) é o responsável pela facilitação, treinamento e aplicação das práticas ágeis e por manter e apresentar nas devidas cerimônias as métricas que foram coletadas em nível de Time de Projeto e Programa. Os ASDs são os responsáveis pelo desenvolvimento do *software* conforme a orientação e priorização de negócios feita pelo ABO a cada ciclo de desenvolvimento. Além disso, os ASDs serão os responsáveis por explorar o detalhamento emergente da arquitetura de *software* e de soluções desenhada inicialmente em nível de Programa pelo AAO.

7.4 CONTINUOUS DELIVERY

O desenho da pesquisa representado pela Figura 4 apresenta a etapa de *Delivery* do modelo TAL adaptado pra *Continuous Delivery* que se inicia com o Planejamento da Iteração que é um processo que inclui a cerimônia de planejamento das HUs previstas para a próxima iteração. O fluxo da Figura 4 segue com Processo de Desenvolvimento que possui todas as técnicas e boas práticas de Engenharia de *Software* para as especificações técnicas, qualidade de código-fonte, desenvolvimento de *software*, testes de unidade, integrados e testes de qualidade (Sommerville, 2007; Pressman, 2021). Nesse sentido que um ciclo de desenvolvimento orientado ao teste é definido entre o *Test-Driven Development* (TDD) e o Processo de Desenvolvimento. O TDD é uma técnica oriunda do *Extreme Programming* (XP) que busca realizar um ciclo de desenvolvimento orientado ao teste e refatoração para otimizar o código-fonte e a qualidade do mesmo (Bakhtiary, Gandomani & Salajegheh 2020). O TDD ajudará os ASDs no desenvolvimento de *software* com mais qualidade desde a construção. O XP é uma prática de desenvolvimento ágil muito difundida e suas práticas são altamente utilizadas em projetos ágeis de desenvolvimento de *software* (Bakhtiary, Gandomani & Salajegheh 2020).

O fluxo da Figura 4 segue com o Processo de Integração Contínua que visa a integração os arquivos de código-fonte gerados pelos desenvolvedores de um ou mais Times de Projeto em um servidor que controla as ramificações e versões do *software* em desenvolvimento através de uma aplicação servidora para esse fim (Dorninger & Ziebermayr, 2021). Em caso da integração com dois ou mais times, temos o Processo de Integração Contínua dos Times. Assim, tanto o Processo de Integração Contínua e quanto o Processo de

Integração Contínua dos Times são práticas utilizadas por *DevOps* (Dorninger & Ziebermayr, 2021) que incluem, além da integração do código-fonte, os testes automatizados com *feedback* rápido aos desenvolvedores sobre os testes de unidade e produzindo métricas importantes sobre o *feedback*.

O fluxo segue na Figura 4 com os Testes de Aceitação do Usuário que visa a demonstração e manuseio da versão pelos clientes e *stakeholders* que é uma importante etapa para obter *feedback* e métricas para decisão entre refutar ou continuar com a direção do projeto durante o ciclo quinzenal de desenvolvimento do produto.

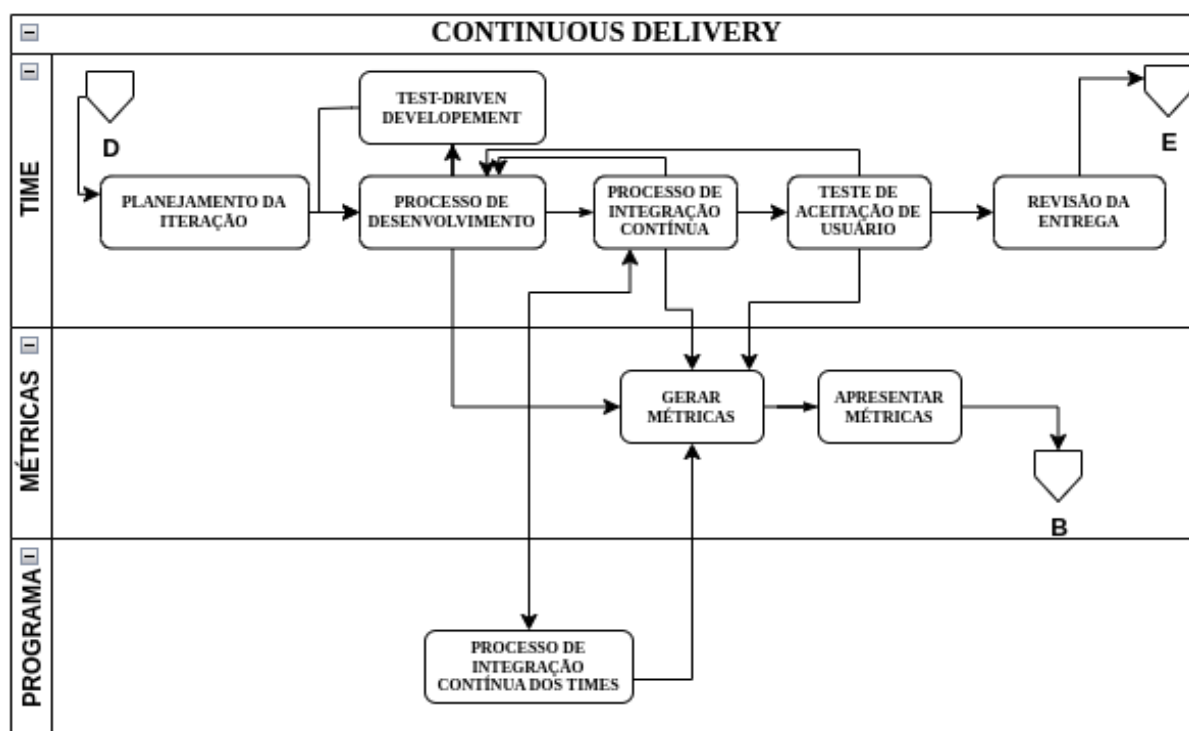


Figura 4 - Fluxo da etapa *Continuous Delivery* para a proposta de *Framework* de Métricas Ágeis (FMA)
Fonte: elaborado pelo autor, 2023.

Nos processos de Processo de Desenvolvimento, Processo de Integração Contínua, Teste de Aceitação de Usuário e Processo de Integração Contínua dos Times são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente. Mais adiante serão apresentadas as métricas sugeridas por etapa.

A Tabela 4 apresenta os papéis e frentes de trabalho relacionadas com a etapa de *Continuous Delivery*. A frente de trabalho envolvida nessa etapa é do Time de Projeto.

Tabela 4- Papéis do Framework de Métricas Ágeis (FMA) na etapa de *Continuous Delivery*

Papel	Frente de Trabalho	Etapa do FMA
<i>Agile Business Owner (ABO)</i>	Time de Projeto	<i>Continuous Delivery</i>
<i>Agile Coach (AC)</i>		
<i>Agile Software Developer (ASD)</i>		

Fonte: elaborado pelo autor, 2023.

O papel do ABO é responsável pela gestão do *Delivery Backlog* com priorização e valor, além da eliminação ou criação dos novos itens conforme o conhecimento sobre o cliente, produto e mercado emerge. Além disso, o ABO deve suportar os ASDs em questões de negócio durante o trimestre corrente. O AC é o responsável pelas práticas ágeis e por manter e apresentar nas devidas cerimônias as métricas que foram coletadas em nível de Time de Projeto e Programa. Os ASDs são responsáveis pelo desenvolvimento do *software* conforme a orientação e priorização de negócios feita pelo ABO a cada ciclo de desenvolvimento.

7.5 CONTINUOUS RELEASE

O fluxo segue na Figura 5 com a etapa de *Release* do modelo TAL adaptado para *Continuous Release*. O processo inicia com o processo de Entrega Contínua que inclui a entrega de uma versão (*release*) do *software* funcional em condições de ser publicado em Produção ou liberado ao mercado para os clientes. A entrega contínua também é uma prática do *DevOps* que visa a entrega rápida e contínua para que, através da utilização rápida pelos clientes, possa obter o *feedback* mais rapidamente possível (Dorninger & Ziebermayr, 2021). Ao finalizar um ciclo de desenvolvimento, cada Time de Projeto se reúne para realizar a Melhoria Contínua do Time que busca identificar os temas de melhorias, aprendizado e sucesso para mapear um plano de ação a ser perseguido pelo Time de Projeto. Assim, o Ciclo Trimestral de Revisão da Entrega Contínua visa revisar, apresentar os resultados e gerenciar as entregas realizadas durante o trimestre corrente para que uma versão do MVP ou melhorias incrementais ou corretivas sejam disponibilizadas aos clientes em produção ou no mercado.

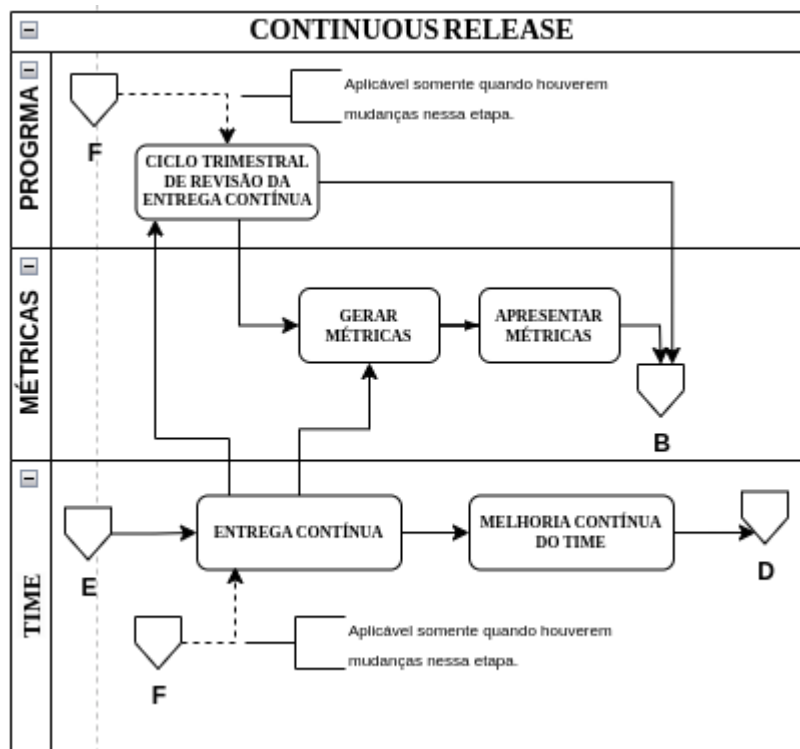


Figura 5 - Fluxo da etapa Continuous Release para a proposta de Framework de Métricas Ágeis (FMA)
 Fonte: elaborado pelo autor, 2023.

Nos processos de Processos de Entrega Contínua e do Ciclo Trimestral de Revisão de Entrega Contínua são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente, e serão apresentadas mais adiante com as métricas propostas pelo FMA por etapa.

A Tabela 5 apresenta os papéis e frentes de trabalho relacionadas com a etapa de Continuous Delivery. As frentes de trabalho envolvidas nessa etapa são o Portfólio, Programa e Time de Projeto.

Tabela 5- Papéis do Framework de Métricas Ágeis (FMA) na etapa de Continuous Release

Papel	Frente de Trabalho	Etapa do FMA
Agile Metrics Owner (AMO)	Portfólio/Programa	Continuous Release
Agile Program Owner (APO1)	Programa	
Agile Business Manager (ABM)		
Agile Architecture Owner (AAO)		
Agile Business Owner (ABO)	Time de Projeto	
Agile Coach (AC)		
Agile Software Developer (ASD)		

Fonte: elaborado pelo autor, 2023.

O APO2 é o responsável pela orquestração dos times de projeto envolvidos nas *features* dentro do Programa. Assim, o APO2 deve manter reuniões durante o trimestre para apoiar dos ABOs e ACs em impedimentos que possa atrapalhar a execução dos trabalhos pelas equipes. Os ABOs de cada time são os responsáveis pela liberação da *release* para clientes e *stakeholders* para que seja medido o valor esperado e o ABM deve acompanhar esse processo de liberação e colaborar com a facilitação e remoção de problemas. Os ACs devem facilitar o evento de liberação com o APO2 e trabalharem combinados com o AMO para apresentarem as métricas relacionadas a liberação realizada até o momento com objetivo de planejarem os próximos passos e revisarem os resultados. Os ASDs devem estar disponíveis para serem acionados para quaisquer problemas e/ou ações necessárias durante a liberação.

7.6 CONTINUOUS IMPROVEMENT

O fluxo segue na Figura 6 com a etapa de *Continuous Improvement* é uma etapa que visa manter as métricas geradas em todas as etapas do TAL e promover a melhoria contínua em cada etapa e nas ações da organização que podem auxiliar na revisão de metas estratégicas para um ou mais ciclos trimestrais de desenvolvimento. O processo inicia com o *Feedback* Contínuo que obtêm as métricas geradas por todas as etapas do TAL que alimenta o processo Planejar Mudanças que visa o mapeamento e planejamento de mudanças necessárias dentro ou após o trimestre corrente. O processo segue com o Executar Mudanças que visa a execução das mudanças planejadas pela organização com objetivo de mudar a rota de um ou mais projetos, programa ou itens do portfólio. Além disso, mudanças estruturais, de pessoas, processos e de gestão podem ser necessárias conforme forem identificadas e planejadas. O processo segue observando os resultados alcançados com as mudanças executadas em Ver Resultados. Assim, dependendo os resultados obtidos, serão necessárias as ações para seguir em frente com a mudança ou ajustar o plano e retornar ao início do processo do *Continuous Improvement* e conforme pode ser visto na Figura 6.

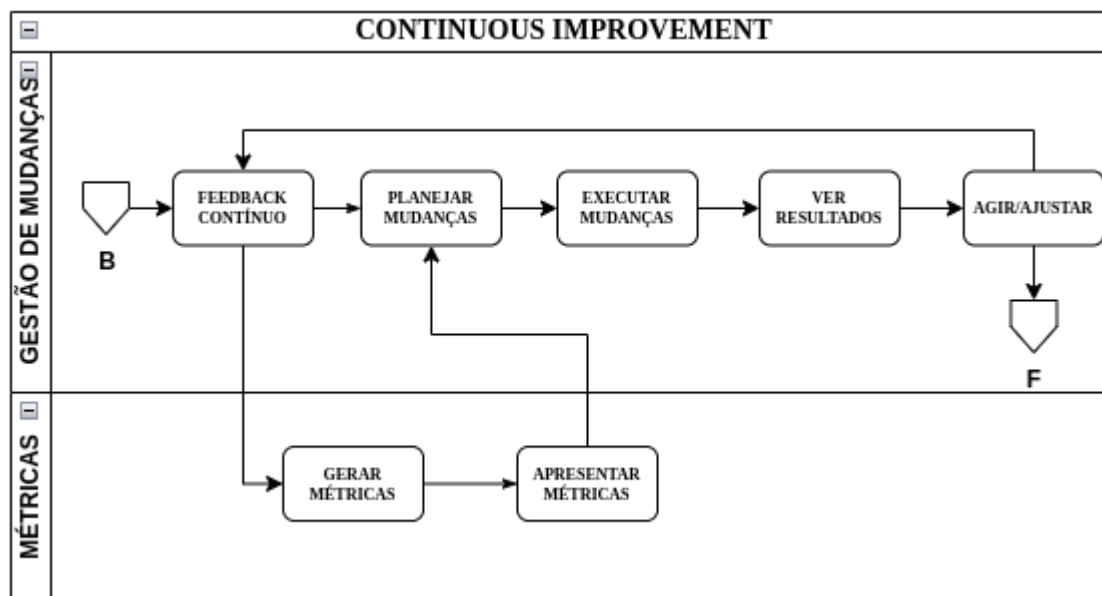


Figura 6 - Fluxo da etapa *Continuous Improvement* para a proposta de *Framework* de Métricas Ágeis (FMA)

Fonte: elaborado pelo autor, 2023.

Nos processos de *Feedback* Contínuo e Planejar Mudanças são coletados os dados de cada um para a geração e apresentação das métricas através dos processos Gerar Métricas e Apresentar Métricas, respectivamente. Posteriormente serão apresentadas as métricas sugeridas para esses processos em um tópico separado.

Assim, a Figura 6 apresenta uma lacuna importante no que tange o uso de métricas ágeis que extrapolam a necessidade de medir somente durante as etapas de *Continuous Discovery*, *Continuous Delivery* e *Continuous Release*, pois na etapa de *Continuous Initiate* que se concentram as metas e estratégias organizacionais para um novo ciclo de desenvolvimento de um produto, manter um produto ou sistemas corporativos existentes, ou para mudar a organização para um novo cenário. Assim, tais metas estratégicas são direcionadas ao portfólio, programas e times de projeto. Nesse sentido que é importante que o processo envolva a medição de todas as etapas do FMA, mas com igual nível de importância pela liderança. Além disso, é importante que o FMA forneça um processo de *feedback* contínuo a fim de viabilizar as métricas em todos os níveis do FMA e permitir que, com esses dados coletados, a organização possa unir pessoas para refletirem sobre esses dados e planejar mudanças na direção do *Backlog* do Portfólio atual e das metas estratégicas que visam alimentar o próximo *Backlog* do Portfólio. Além disso, métricas importantes serão coletadas para que stakeholders possam avaliar a relação entre o orçamento planejado e executado a cada trimestre e as entregas previstas e entregas realizadas.

A Tabela 6 apresenta os papéis e frentes de trabalho relacionadas com a etapa de *Continuous Delivery*. As frentes de trabalho envolvidas nessa etapa são o Portfólio, Programa e Time.

Tabela 6- Papéis do *Framework* de Métricas Ágeis (FMA) para a etapa de *Continuous Improvement*

Papel	Frente de Trabalho	Etapa do FMA
<i>Agile Metrics Owner (AMO)</i>	Portfólio/Programa	<i>Continuous Improvement</i>
<i>Agile Portfolio Owner (APO1)</i>	Portfólio	
<i>Agile Business Manager (ABM)</i>	Programa	
<i>Agile Program Owner (APO2)</i>		
<i>Agile Business Owner (ABO)</i>	Time de Projeto	
<i>Agile Coach (AC)</i>		
<i>Agile Software Developer (ASD)</i>		

Fonte: elaborado pelo autor, 2023.

O *Continuous Improvement* é uma etapa de colaboração entre todos os envolvidos nos papéis definidos pelo FMA. Somente o AMO deve ser o responsável de obter e apresentar as métricas de todo os ciclos trimestrais realizados até o momento das cerimônias. Os demais devem colaborar e facilitar para que as estratégias de negócios, projeto e operacional sejam revistas para o próximo ciclo trimestral com objetivo de manter o cliente no centro das decisões e entregar o melhor valor possível no melhor tempo. Os ASDs devem definir um representante para participar do *Continuous Improvement* visto que não é necessário a participação de todos. Isso torna a reunião mais enxuta, mais simples de gerenciar e otimiza a comunicação. Assim, cada papel deve sair com um plano de mudanças sob a sua responsabilidade e prazo de entrega alinhado com a sua frente de trabalho e etapa do FMA.

A Tabela 7 a seguir apresenta as responsabilidades de cada papel conforme o tipo de mudança a ser planejada e executada. É importante destacar que as estratégias apresentadas na Tabela 7 são alguns cenários que podem ser tratados como mudanças no FMA, entretanto, as mudanças não se limitam aos cenários da Tabela 7.

Tabela 7- Estratégias e papéis do *Framework* de Métricas Ágeis (FMA) para a etapa de *Continuous Improvement*

Estratégia no <i>Continuous Improvement</i>	Papel	Etapa do FMA
Revisar/ajustar/criar métricas	<i>Agile Metrics Owner</i>	Todas as Etapas
Metas estratégicas de negócios/organizacionais	<i>Agile Portfolio Owner</i> <i>Agile Business Manager</i>	<i>Continuous Initiate</i>
Iniciativas do programa	<i>Agile Program Owner</i>	<i>Continuous Initiate</i>

Iniciativas do produto	<i>Agile Business Owner</i>	<i>Continuous Discovery</i>
Iniciativas sobre agilidade	<i>Agile Program Owner</i> <i>Agile Coach</i> <i>Agile Metrics Owner</i>	Todas as Etapas
Iniciativas de desenvolvimento de <i>software</i>	<i>Agile Software Developer</i>	<i>Continuous Delivery</i>
Iniciativas de integração contínua e <i>release</i>	<i>Agile Metrics Owner</i> <i>Agile Software Developer</i>	<i>Continuous Release</i>
Revisão do processo de melhoria contínua.	<i>Agile Metrics Owner</i> <i>Agile Program Owner</i> <i>Agile Coach</i>	<i>Continuous Improvement</i>

Fonte: elaborado pelo autor, 2023.

Com objetivo de contribuir tecnicamente com a etapa de *Continuous Improvement* e, conforme a Tabela 7, os ASDs precisam definir um representante para participar do processo de melhoria contínua. Isso contribui e ajuda na condução da reunião, na definição do plano e otimiza a comunicação mantendo as pessoas-chave na reunião. Assim, a Tabela 7 apresenta de maneira substancial as iniciativas organizacionais onde as melhorias possam ser aplicadas para rever todo o processo que envolve o FMA, o processo ágil, a estrutura organizacional, papéis e as estratégias organizacionais. Nesse sentido, as ações de melhorias podem ser aplicadas em qualquer área em que as pessoas que estiverem colaborando durante o *Continuous Improvement* definirem como importantes para o momento.

7.7 FRAMEWORK DE MÉTRICAS

A Figura 8 apresenta o fluxo do processo integrado entre todas as etapas do FMA. Assim, por se tratar de um fluxo contínuo de metas organizacionais, gestão orçamentária, desenvolvimento de produto, medição e *feedback* contínuos, o fluxo do FMA não tem uma data fim definida em projetos de escopo fechado como previsto na gestão de projetos tradicional. Isto é, o FMA deve respeitar as estratégias de negócio para um ciclo de vida do produto de *software*. Esse ciclo de vida é variável dependendo do avanço da tecnologia, do mercado onde o *software* atua, do perfil de seus clientes e do tipo de tecnologia (*web*, *mobile*, computação embarcada etc.).

A Figura 8 apresenta todas as etapas do FMA que podem ser adotadas em sua completude ou através de um conjunto de etapas em duas partes separadas. Essa flexibilidade de adotar o FMA em partes dependerá do propósito de cada organização e como ela deseja monitorar e realizar a gestão ágil de seus projetos. Assim, como discutido no Estudo 4, a

Tabela 8 apresenta como o FMA pode ser dividido em partes como o Nível Essencial e o Nível Completo, e qual o propósito da organização para cada nível.

Tabela 8- Níveis de adoção do Framework de Métricas Ágeis (FMA) em dois níveis de propósitos organizacionais

Nível do FMA	Nome do Nível	Partes do FMA	Propósito da Organização em Cada Nível
1	Nível Essencial	- <i>Continuous Discovery</i> - <i>Continuous Delivery</i> - <i>Continuous Release</i>	Realizar a gestão do projeto com foco em escopo a partir de histórias de usuário descobertas diretamente entre o time de projetos e as áreas de negócios da organização. Dessa maneira, a organização entrega as metas e projetos diretamente para a área de negócios ou demandante que entrega o escopo em ciclos quinzenais para o time de projetos. Além disso, a área demandante que é responsável pela gestão orçamentária do projeto.
2	Nível Completo	- Nível Essencial - <i>Continuous Initiate</i> - <i>Continuous Improvement</i>	Realizar a gestão do projeto desde a iniciação com gestão de portfólio e metas, orçamento e programa. A gestão de escopo é feita pelo FMA através do <i>Continuous Initiate</i> para o Nível Essencial com ações de melhoria contínua em nível organizacional do projeto através do <i>Continuous Improvement</i> .

Fonte: elaborado pelo autor, 2023.

Além da organização que pode implementar o FMA em partes separadas e evoluindo do nível essencial para o completo, a flexibilidade do FMA vai além disso. Assim, a Tabela 9 apresenta como o FMA pode ser adotado em partes por uma única organização e em partes por organizações diferentes que operam na mesma iniciativa e no mesmo projeto ágil.

Tabela 9- Níveis de adoção do Framework de Métricas Ágeis (FMA) em diferentes modelos organizacionais do Projeto Ágil

Nível do FMA	Nome do Nível	Modelo Organizacional do Projeto Ágil	Propósito da Organização em Cada Nível
1	Nível Essencial	Organização orientada a projetos internos.	Realizar a gestão do projeto no Nível Essencial sem dependências externas de infraestrutura e de pessoas externas gerenciadas por fornecedores. A organização possui toda a estrutura e pessoas para conduzir por si própria seus projetos. A organização é a única responsável pela gestão de métricas.
2	Nível Completo	Organização orientada a projetos internos.	Realizar a gestão do projeto no Nível Completo sem dependências externas de infraestrutura e de pessoas externas gerenciadas por fornecedores. A organização possui toda a estrutura e pessoas para conduzir por si própria seus projetos. A organização é a única responsável pela gestão de métricas.
1	Nível	Organização cliente que terceiriza a	Organização que detêm as metas e projetos

	Essencial	parte da gestão ágil projetos para outra organização fornecedora.	estratégicos, mas a gestão do Nível Essencial é terceirizada para outra organização fornecedora orientada a projetos.
		Organização fornecedora de serviços de gestão ágil de projetos contratada pela Organização cliente.	Organização fornecedora que foi terceirizada pela Organização cliente. A Organização fornecedora realiza a gestão do projeto no Nível Essencial. A organização fornecedora é a única responsável pela gestão de métricas.
2	Nível Completo	Organização cliente que terceiriza a parte da gestão ágil projetos para outra organização fornecedora.	Organização que detêm as metas e projetos estratégicos, e realiza a gestão do Nível Completo nas etapas de <i>Continuous Initiate</i> e <i>Continuous Improvement</i> , ficando as demais etapas terceirizadas para outra organização fornecedora orientada a projetos. A organização cliente é a única responsável pela gestão de métricas nessas etapas.
		Organização fornecedora de serviços de gestão ágil de projetos contratada pela Organização cliente.	Organização fornecedora que foi contratada pela Organização cliente. A Organização fornecedora realiza a gestão do projeto no Nível Completo nas etapas de <i>Continuous Discovery</i> , <i>Continuous Delivery</i> e <i>Continuous Release</i> . A organização fornecedora é a única responsável pela gestão de métricas nessas etapas.

Fonte: elaborado pelo autor, 2023.

A Figura 8 apresenta o mesmo fluxo da Figura 7, mas com uma forma visual diferente, apresentando acompanhado dos *loops* para representar diversos ciclos de iterações sendo executados dentro das etapas, além dos papéis e artefatos em cada etapa do FMA.

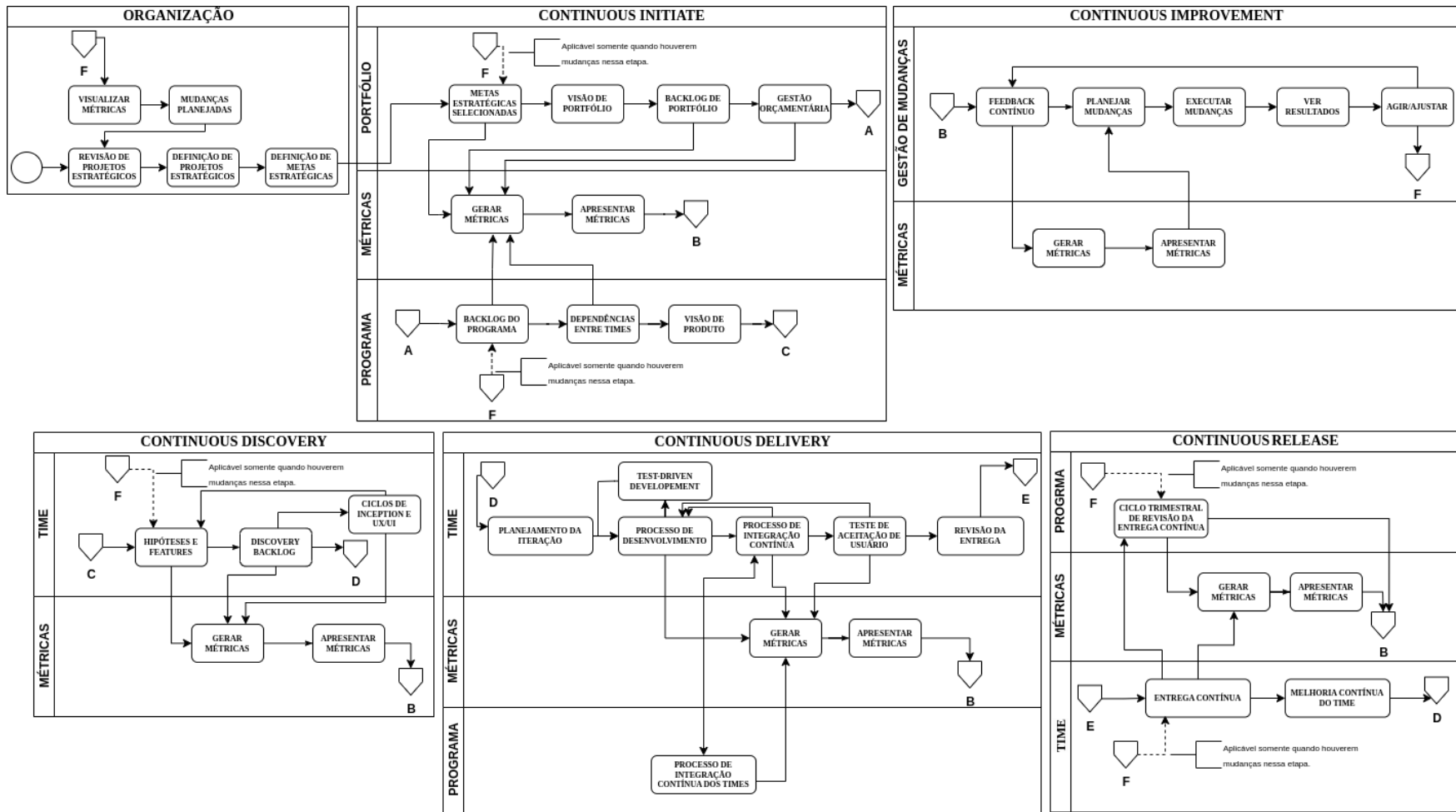


Figura 7 - Fluxo de processo do Framework de Métricas Ágeis (FMA)

Fonte: elaborado pelo autor, 2023.

A Figura 8 apresenta o fluxo do processo integrado com todos os papéis envolvidos entre todas as etapas do FMA.

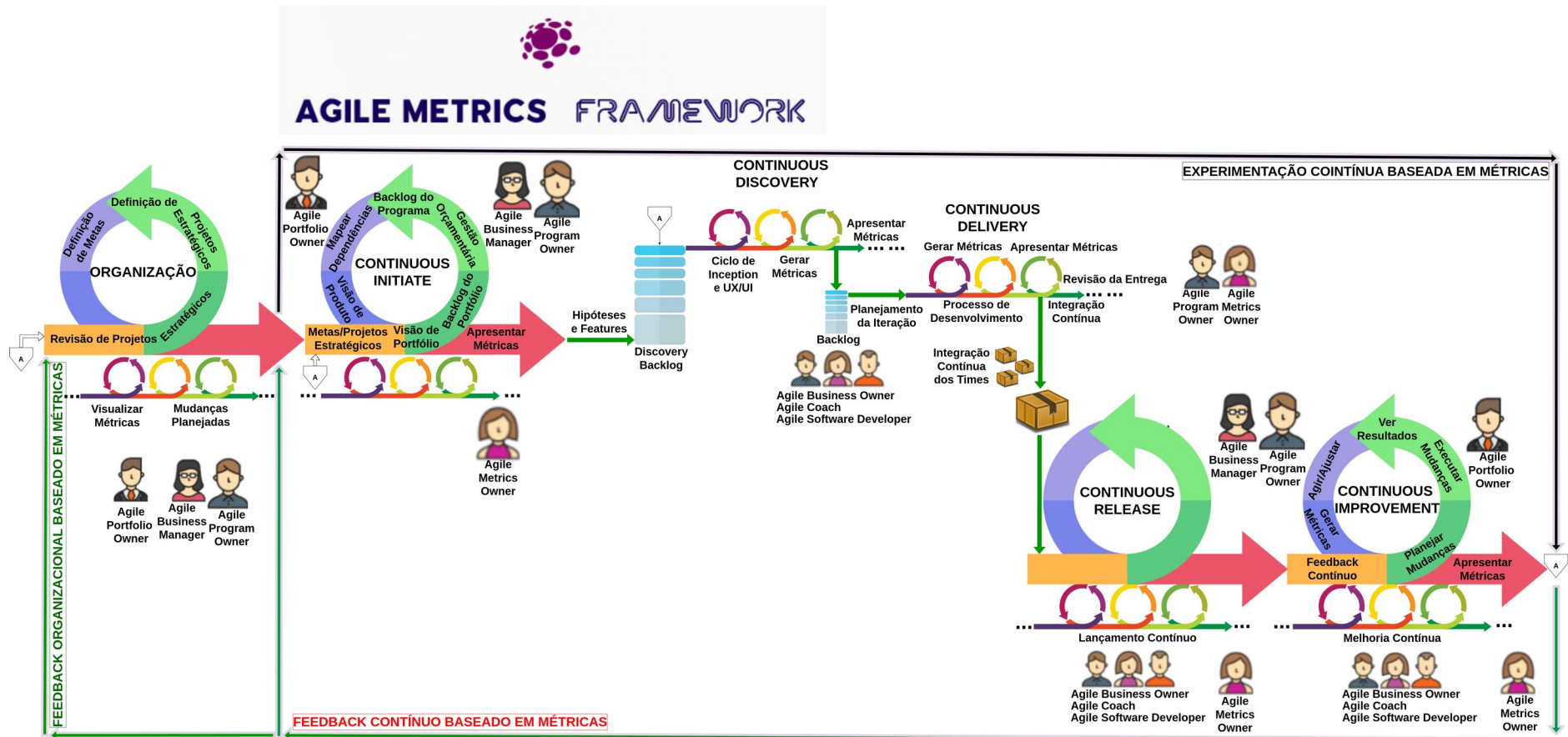


Figura 8 - Fluxo detalhado do Framework de Métricas Ágeis (FMA)

Fonte: elaborado pelo autor, 2023.

Como o FMA é um *framework*, a organização e execução do fluxo de trabalho do FMA necessita de cerimônias para organizar a gestão e tornar transparente o processo de a inspeção e adaptação de seu próprio processo e sobre a evolução do produto de *software* construído pelo projeto. Assim, a Tabela 10 apresenta as cerimônias para cada etapa do FMA que correspondem ao *Continuous Initiate*, *Continuous Discovery*, *Continuous Delivery*, *Continuous Release* e *Continuous Improvement*. Além disso, a Tabela 10 apresenta cada cerimônia com a periodicidade e frente de trabalho associados.

Tabela 10- Cerimônias do Framework de Métricas Ágeis (FMA)

Cerimônia	Periodicidade	Frente de Trabalho	Etapa do FMA
Planejamento de Portfólio	Semanal	Portfólio	<i>Continuous Initiate</i>
Planejamento de Orçamento	Semanal	Portfólio	<i>Continuous Initiate</i>
Revisão de Métricas	Trimestral	Portfólio	<i>Continuous Initiate</i>
Planejamento do <i>Backlog</i> do Programa	Quinzenal	Programa	<i>Continuous Initiate</i>
Reunião de Dependências e Riscos do Programa	Quinzenal	Programa	<i>Continuous Initiate</i>
Refinamento do <i>Discovery Backlog</i>	Semanal	Time de Projeto	<i>Continuous Discovery</i>
Planejamento do <i>Discovery Backlog</i>	Quinzenal	Time de Projeto	<i>Continuous Discovery</i>
Revisão de Métricas	Quinzenal	Time de Projeto	<i>Continuous Discovery</i>
Revisão de Métricas	Diária	Time de Projeto	<i>Continuous Delivery</i>
Sincronização entre <i>Business Owners</i>	Semanal	Programa/Time de Projeto	<i>Continuous Delivery</i>
Sincronização entre <i>Agile Coaches</i>	Semanal	Programa/Time de Projeto	<i>Continuous Delivery</i>
Reunião de Integração Contínua	Semanal	Time de Projeto	<i>Continuous Delivery</i>
Reunião de Integração Contínua dos Times	Quinzenal	Time de Projeto	<i>Continuous Delivey</i>
Revisão de Métricas	Quinzenal	Time de Projeto	<i>Continuous Delivey</i>
Reunião de Entrega Contínua	Mensal	Time de Projeto	<i>Continuous Release</i>
Revisão de Métricas	Mensal	Time de Projeto	<i>Continuous Release</i>
Reunião de Entrega Contínua	Trimestral	Programa	<i>Continuous Release</i>
Revisão de Métricas	Trimestral	Programa	<i>Continuous Release</i>
Planejamento de Mudanças	Trimestral	Gestão de Mudanças	<i>Continuous Improvement</i>
Revisão de Métricas	Trimestral	Gestão de Mudanças	<i>Continuous Improvement</i>
Revisão dos Resultados de Mudanças	Trimestral	Gestão de Mudanças	<i>Continuous Improvement</i>

Fonte: elaborado pelo autor, 2023.

Assim, conforme a Tabela 10, as Reuniões de Métricas (RM) acontecem em frequências diária, semanal, quinzenal e trimestral em todas as etapas do FMA. Assim, a RM cobrirá a necessidade de revisão das métricas, acompanhamento das *features* e HUs do projeto na forma constante e ativa para avaliação do andamento, melhorias a serem identificadas e *feedback* contínuo.

As reuniões de Planejamento de Portfólio e Planejamento de Orçamento são cerimônias da etapa de *Continuous Initiate* onde o APO1 e ABM colaboram entre si e com membros da organização para elaborarem o *Backlog* do Portfólio para o próximo trimestre. O AMO facilitará as decisões apresentando as métricas do trimestre corrente com objetivo de auxiliar nas decisões do APO1 e ABM sobre as estratégias do *Backlog* do Portfólio.

As cerimônias do Planejamento do *Backlog* do Programa e Reunião de Dependências e Riscos do Programa acontecem em nível de Programa onde os APO1, APO2, ABM colaboram para elaborar o *Backlog* do Programa, identificando as dependências de negócios e possíveis riscos com a visão trimestral dos itens de maior valor que foram priorizados em portfólio.

As cerimônias do Refinamento do *Discovery Backlog* e o Planejamento do *Discovery Backlog* ocorrem semanal e quinzenalmente, respectivamente. O Refinamento do *Discovery Backlog* visa antecipar a discussão entre ABO, AC e ASDs sobre os próximos requisitos a serem desenvolvidos com objetivo de dirimir eventuais dúvidas ou necessidades de detalhamento de negócio e/ou técnica. É importante que o ABO e o ASDs tenham o entendimento sobre e em quais condições uma HU está pronta para o próximo ciclo de desenvolvimento. Essa tratativa é uma lista de itens conhecidos como pré-requisitos ou *definition of ready* (DoR) para cada HU. A antecipação da discussão pode ocorrer sobre um ou dois ou mais ciclos de desenvolvimento futuros para reduzir os riscos de uma HU chegar no ciclo de desenvolvimento com pendências de negócio e/ou técnicas. O Planejamento do *Discovery Backlog* visa reunir o ABO, AC e ASDs sobre o planejamento de itens para o próximo ciclo de desenvolvimento que se inicia a seguir.

As cerimônias de Sincronização entre *Business Owners* (SBO) e Sincronização entre *Agile Coaches* (SAC) buscam reunir semanalmente os ABOs e ACs, respectivamente, no sentido de alinhar o andamento das iterações dentro do trimestre com a possibilidade também de sinalizar eventuais riscos e impedimentos que estejam bloqueando o desenvolvimento das entregas da iteração em andamento e de iterações futuras. Além disso, essas cerimônias são facilitadas pelo APO2 que deve colaborar tais impedimentos e riscos com o APO1 e ABM no

sentido de escalar problemas que os ABOs e os ACs não estejam conseguindo resolver em seu nível operacional de atuação no FMA.

As cerimônias de Reunião de Integração Contínua (RIC) e Reunião de Integração Contínua dos Times (RICT) são momentos importantes para que os times mantenham a integração dos códigos-fonte gerados pelos Times de Projeto. O RIC é realizado semanalmente por cada time individualmente como um evento, entretanto, cada ASD de cada time pode fazer a integração contínua a qualquer momento para não criar uma dependência de integração somente no evento. Assim, cada ASD alinha junto com o seu Time de Projeto o momento para realizar a revisão de código-fonte e integração contínua. Para o RICT é realizado quinzenalmente em conjunto entre um ASD representante de cada time do programa com o apoio do AAO e a facilitação do AC. Como cada ASD pode desenvolver em diferentes camadas na arquitetura de *software*, o processo de integração é sempre um trabalho complexo e desafiante no sentido da correção e ajustes de código-fonte durante o desenvolvimento de *software* integrado. Dessa forma, a RIC é um encontro semanal para alinhamento e execução da integração. O RICT é um encontro quinzenal visando o último trabalho de integração no final da iteração atual. Esses dois encontros são ricas oportunidades para tornar transparentes os problemas, o aprendizado e uma forma de disseminar o conhecimento adquirido e emergente.

Com as integrações realizadas de forma contínua pelos times, a cerimônia de Reunião de Entrega Contínua é realizada em período mensal e trimestralmente. Os lançamentos ou *releases* podem ser liberadas a qualquer momento pelos times ou no limite do trimestre para uma versão estável com objetivo de obter o *feedback* de clientes e do mercado, além de ser objeto de verificação de hipóteses pelos ABO e APM sobre as novas implementações recém-lançadas.

As cerimônias de Planejamento de Mudanças e Revisão dos Resultados de Mudanças fazem parte do processo de *Continuous Improvement* que visa o planejamento das mudanças necessárias conforme as métricas apresentadas em todas as etapas do FMA e a revisão dos resultados e ações necessárias, respectivamente.

Nos ciclos ou iterações de desenvolvimento do FMA, tanto o *Framework Scrum* (FS) quanto o Método *Kanban* pode ser utilizado para aplicar práticas de gestão ágil de projetos. A utilização dessas práticas ou não sempre ficará sob escolha dos Times de Projeto e da Organização. Assim, a Tabela 11 apresenta o relacionamento entre as cerimônias do FMA com o FS, objetivos e os relacionamentos com as etapas do FMA.

Tabela 11- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com as Cerimônias do Framework Scrum (FS)

Cerimônia FMA	Cerimônia FS	Objetivo	Etapa do FMA
Revisão de Métricas Planejamento do <i>Discovery Backlog</i>	<i>Sprint Planning</i>	Planejar o próximo ciclo de desenvolvimento.	<i>Continuous Discovery</i>
Revisão de Métricas	<i>Daily Meeting</i>	Apresentar as métricas diárias do desenvolvimento.	<i>Continuous Delivery</i>
Revisão de Métricas Reunião de Entrega Contínua	<i>Sprint Review</i>	Apresentar as métricas do ciclo de desenvolvimentos relacionados ao produto.	<i>Continuous Delivery</i> <i>Continuous Release</i>
Revisão de Métricas Planejamento de Mudanças Revisão Resultados de Mudanças	<i>Sprint Retrospective</i>	Apresentar as métricas do ciclo de desenvolvimentos relacionados ao produto e trabalho relacionado ao time.	<i>Continuous Delivery</i> <i>Continuous Improvement</i>

Fonte: elaborado pelo autor, 2023.

Durante o ciclo de desenvolvimento do FMA, o FS pode ser utilizado para a gestão ágil de projetos de desenvolvimento de *software*. Isto é, o FS poderá potencializar os benefícios do FMA, assim com a utilização do quadro *Kanban* para dar mais transparência e controle de fluxo de trabalho combinado ao FMA e o FS. É importante destacar que o quadro *Kanban* é uma ferramenta de gestão visual de processo e o Método *Kanban* é um método de gestão de projetos com papéis e cerimônias (Dos Santos *et al.*, 2018). Assim, a Tabela 11 demonstra que as cerimônias do FMA possuem relacionamento direto com as cerimônias do FS e, essencialmente os eventos são semelhantes, apesar de que o FMA busca apoiar a liderança na gestão ágil de projetos e na agilidade organizacional, não apenas em escopo operacional (*discovery* e *delivery*) do projeto ágil. Além disso, as revisões de métricas acompanhadas dos eventos correspondentes, devem apoiar as cerimônias onde há decisões com base em dados históricos e tendências sobre o que fazer a seguir.

Assim como a Tabela 11 acima, a Tabela 12 apresenta o relacionamento entre as cerimônias do FMA com o Método *Kanban* (MK), objetivos e o relacionamento com as etapas do FMA. O MK poderá ser utiliza para potencializar os benefícios do uso do FMA.

Tabela 12- Relacionamento das Cerimônias do Framework de Métricas Ágeis (FMA) com as Cerimônias do Método Kanban (MK)

Cerimônia FMA	Cerimônia MK	Objetivo	Etapa do FMA
Revisão de Métricas Planejamento do <i>Discovery Backlog</i>	<i>Replenishment Meeting</i>	Refinar os próximos ciclos de desenvolvimento.	<i>Continuous Discovery</i>

Revisão de Métricas	<i>Kanban Meeting</i>	Apresentar as métricas diárias do desenvolvimento.	<i>Continuous Delivery</i>
Reunião de Entrega Contínua Revisão de Métricas	<i>Delivery Planning</i>	Planejar e formalizar as entregas das <i>releases</i> .	<i>Continuous Delivery</i>
Revisão de Métricas Reunião de Entrega Contínua Planejamento de Mudanças Revisão Resultados de Mudanças	<i>Strategy Review</i>	Apresentar as métricas do ciclo de desenvolvimentos relacionados ao produto e trabalho referente ao time.	Todas as Etapas
Revisão de Métricas	<i>Operations Review</i>	Revisão do sistema ou o fluxo de trabalho.	<i>Continuous Discovery</i> <i>Continuous Delivery</i>
Revisão de Métricas Reunião de Entrega Contínua Revisão Resultados de Mudanças	<i>Service-delivery Review</i>	Verificar se estão entregando as demandas de acordo com as expectativas de clientes.	<i>Continuous Delivery</i> <i>Continuous Improvement</i>
Revisão de Métricas Reunião de Dependências e Riscos do Programa	<i>Risk Review</i>	Verificar os Riscos inerentes ao fluxo de trabalho que podem comprometer a entrega.	Todas as Etapas

Fonte: elaborado pelo autor, 2023.

Através da Tabela 12, observa-se um relacionamento de cerimônias com formatos e objetivos em comum entre o FMA e as revisões de métricas se fazem necessárias também no MK quando adotado nas etapas de *Continuous Discovery* e *Continuous Delivery* do FMA. Assim, as métricas previstas no MK podem ser mantidas, mas deve-se observar as métricas nos demais níveis do FMA, como Programa e Portfólio.

Com objetivo de lidar com as mudanças necessárias na organização e nos projetos ágeis e realizar a inspeção e adaptação baseadas em métricas, a Tabela 13 apresenta o relacionamento entre as cerimônias do FMA com o Método PDCA (*Plan-Do-Check-Act*), objetivos e o relacionamento com as etapas do FMA. O PDCA é um método iterativo de gestão de mudanças composto por quatro etapas, sendo que pode ser utilizado para melhoria contínua e o controle de processos de negócios, mudanças estruturais em organizações, serviços e produtos (Isniah, Purba e Debora, 2020). O processo do FMA é inspirado no PDCA no que se refere ao *Continuous Improvement*, mas também pode ser considerado uma *Big Retrospective* em nível organizacional do projeto que coleta dados de métricas constantemente.

Tabela 13- Relacionamento das Cerimônias do *Framework* de Métricas Ágeis (FMA) com os processos do PDCA

Cerimônia FMA	Cerimônia PDCA	Objetivo	Etapa do FMA
Revisão de Métricas Planejamento de Mudanças	Planejar (<i>Plan</i>)	Planejar mudanças a serem experimentadas pela próxima etapa.	<i>Continuous Improvement</i>
Revisão de Métricas	Executar (<i>Do</i>).	Execução de ações relacionadas as mudanças planejadas com objetivo de validar hipóteses e soluções.	<i>Continuous Improvement</i>
Revisão de Métricas Revisão dos Resultados de Mudanças	Verificar (<i>Check</i>)	Monitorar e verificar os resultados de mudanças planejadas	<i>Continuous Improvement</i>
Revisão de Métricas	Agir/Ajustar (<i>Act/Adjust</i>)	Criar padrões e executar mudanças definitivas.	<i>Continuous Improvement</i>

Fonte: elaborado pelo autor, 2023.

Como os projetos nascem de metas e estratégias organizacionais, o FMA já contempla essa necessidade em seu fluxo de processo. Dessa forma, a adoção de técnicas para a gestão de metas poderá ser utilizada combinado ao FMA para potencializá-lo. Assim, a Tabela 14 apresenta o relacionamento entre as cerimônias do FMA com as etapas do *Objectives Key-Results* (OKR), objetivos e o relacionamento com as etapas do FMA. O OKR é um *framework* para gestão ágil de desempenho através da definição de objetivos (metas) que passam ser mensurados através do planejamento de resultados-chave (Stray *et al.*, 2023; Vedal *et al.*, 2021). Além disso, o OKR permite criar metas em nível estratégico, gerencial e operacional com revisões semanais e trimestrais (Stray *et al.*, 2023; Vedal *et al.*, 2021). Assim, esse modelo de trimestral do OKR está em sincronia com o fluxo de trabalho que opera também de maneira trimestral do FMA.

Tabela 14- Relacionamento das Cerimônias do *Framework* de Métricas Ágeis (FMA) com as etapas do *Objectives Key-Results* (OKR)

Cerimônia FMA	Etapas OKR	Objetivo	Etapa do FMA
Revisão de Métricas Planejamento de Mudanças Revisão dos Resultados de Mudanças	OKRs Estratégicos	Planejar, definir e validar os OKRs Estratégicos.	<i>Continuous Initiate</i>
Planejamento do <i>Backlog</i> do Programa Reunião de Dependências do Programa	OKRs dos Times (Programa)	Planejar, definir e validar os OKRs dos Times (Programa).	<i>Continuous Initiate</i>
Revisão de Métricas Sincronização entre <i>Business Owners</i> Sincronização entre <i>Agile Coaches</i>	Monitorar OKRs	Monitoramento semanal e análise dos OKRs	<i>Continuous Delivery</i>

Reunião de Entrega Contínua Revisão de Métricas	Revisar OKRs	Revisão Trimestral dos OKRs	<i>Continuous Release</i>
Planejamento de Mudanças Revisão de Métricas Revisão dos Resultados de Mudanças	Lições Aprendidas	Retrospectiva e Lições Aprendidas	<i>Continuous Improvement</i>

Fonte: elaborado pelo autor, 2023.

A definição de papéis é importante para a execução do FMA nas organizações para a condução das cerimônias e garantir as entradas e saídas de artefatos esperados. Assim, a Tabela 15 apresenta os papéis do FMA, responsabilidade, frentes de trabalho e o relacionamento de cada papel nas etapas do FMA. Além disso, a Tabela 15 apresenta como esses papéis colaboram entre si para que o desenvolvimento de *software* em escala organizacional ou por iniciativas de projetos isolados e que possa ocorrer seguindo os princípios de agilidade em projetos e agilidade organizacional.

Tabela 15- Os papéis do *Framework* de Métricas Ágeis (FMA) e o relacionamento de cada papel nas etapas do FMA

Papel	Responsabilidade	Frente de Trabalho	Etapa do FMA
<i>Agile Metrics Owner</i>	Responsável por manter, coletar e apresentar as métricas nas cerimônias.	Portfólio/Programa	<i>Continuous Initiate</i>
<i>Agile Portfolio Owner</i>	Responsável pela interface com executivos para as estratégias de negócios e de produtos. Além disso, mantém a gestão do <i>Backlog</i> de Portfólio e da Gestão Orçamentária.	Portfólio	
<i>Agile Business Manager</i>	Responsável pela gestão de negócios de um ou mais produtos dentro do Programa.	Programa	
<i>Agile Program Owner</i>	Responsável pela condução de projetos de um ou mais produtos dentro do Programa.		
<i>Agile Architecture Owner</i>	Responsável pela arquitetura de <i>software</i> em nível organizacional do Programa.		
<i>Agile Business Owner</i>	Responsável pela gestão de negócios de um produto dentro do Time, ou seja, é o responsável pelo <i>Discovery Backlog</i> .	Time	<i>Continuous Discovery</i>
<i>Agile Coach</i>	Responsável em treinar, facilitar e orientar o time nas práticas ágeis na etapa de <i>Discovery</i> . Além disso, deve manter, coletar e apresentar as métricas nas cerimônias.		
<i>Agile Software Developer</i>	Responsável pela colaboração durante o refinamento do <i>Discovery Backlog</i> .		
<i>Agile Business Owner</i>	Responsável em apoiar os <i>Softwares Developers</i> sobre questões relacionadas os itens de negócios <i>Discovery Backlog</i> que estão em desenvolvimento na etapa de <i>Delivery</i> .	Time	<i>Continuous Delivery</i>

<i>Agile Coach</i>	Responsável em treinar, facilitar e orientar o time nas práticas ágeis na etapa de <i>Delivery</i> . Além disso, deve manter, coletar e apresentar as métricas nas cerimônias.			
<i>Agile Software Developer</i>	Responsáveis em atingir a meta de desenvolvimento dos itens comprometidos com o <i>Business Owner</i> e gerar um incremento de <i>software</i> funcional.			
<i>Agile Program Owner</i>	Responsável pela condução do Ciclo Trimestral de Entrega Contínua de um ou mais produtos dentro do Programa.	Programa	<i>Continuous Release</i>	
<i>Agile Business Manager</i>	Responsável em apoiar o <i>Business Owner</i> , <i>Program Owner</i> e <i>Software Developers</i> sobre questões relacionadas sobre questões ligadas a <i>Release</i> e a sua liberação.			
<i>Agile Architecture Owner</i>	Responsável pela arquitetura de <i>software</i> em nível organizacional do Programa.			
<i>Agile Metrics Owner</i>	Responsável por manter, coletar e apresentar as métricas nas cerimônias identificando e facilitando decisões sobre oportunidades de melhorias e novas estratégias de processo e negócio.	Portfólio/Programa		
<i>Agile Business Owner</i>	Responsável em apoiar os <i>Softwares Developers</i> e trabalhar com o <i>Business Manager</i> sobre questões relacionadas sobre questões ligadas a <i>Release</i> e a sua liberação.	Time		
<i>Agile Coach</i>	Responsável em treinar, facilitar e orientar o time nas práticas ágeis na etapa de <i>Release</i> . Além disso, deve apoiar o <i>Metrics Owner</i> com as métricas.			
<i>Agile Software Developer</i>	Responsável pela liberação de uma <i>release</i> integrada com outros times de desenvolvimento envolvidas no Programa.			
<i>Agile Metrics Owner</i>	Responsável por manter, coletar e apresentar as métricas nas cerimônias identificando e facilitando decisões sobre oportunidades de melhorias e novas estratégias de processo e negócio.	Portfólio/Programa		<i>Continuous Improvement</i>
<i>Agile Portfolio Owner</i>	Responsável por identificar oportunidades de melhorias e novas estratégias de processo e negócio e escalar em nível executivo/organizacional.	Portfólio		
<i>Agile Business Manager</i>	Responsável por identificar oportunidades de melhorias de negócio e produto, além de escalar melhorias necessárias em nível de Programa e Portfólio.	Programa		
<i>Agile Program Owner</i>	Responsável por identificar oportunidades de melhorias de processo de gestão ágil de projetos em nível de Programa e Time, além de escalar melhorias necessárias em nível de Programa e Portfólio.			
<i>Agile Architecture Owner</i>	Responsável pela arquitetura de <i>software</i> em nível organizacional do Programa.			
<i>Agile Business Owner</i>	Responsável por identificar oportunidades de melhorias de negócio e produto, além de escalar melhorias necessárias em nível de Time e Programa.	Time		

<i>Agile Coach</i>	Responsável por identificar oportunidades de melhorias de processo de gestão ágil de projetos em nível de Time, além de escalar melhorias necessárias em nível de Time e Programa.		
<i>Agile Software Developer</i>	Responsável por identificar oportunidades de melhorias de processo de desenvolvimento ágil de software.		

Fonte: elaborado pelo autor, 2023.

Como desde a organização, o FMA possui entradas e saídas esperadas. Essas entradas e saídas possuem artefatos necessários para alimentar as etapas e prover as métricas necessárias para o propósito do FMA de ser uma ferramenta de processo e gerencial para apoiar a liderança corporativa. Nesse sentido que a Tabela 16 apresenta os artefatos, etapas e o objetivo de cada artefato previsto por etapa do FMA. Assim, foram definidos os artefatos que devem ser mantidos em cada etapa do FMA e esses artefatos são elementos importantes para a realização da gestão do escopo em nível de Portfólio, Programa e de Time de Projeto que operam em escala ou não. Além disso, os artefatos são ferramentas para promover a transparência, inspeção e adaptação (empirismo) do que está sendo feito em cada etapa do FMA. Por fim, os artefatos relacionados na Tabela 16 são bases de dados para a geração de métricas em todas as etapas do FMA.

Tabela 16- Artefatos por Etapas do *Framework* de Métricas Ágeis (FMA)

Artefato	Etapa FMA	Objetivo
Visão de Portfólio	<i>Continuous Initiate</i>	Prover a visão do portfólio do trimestre corrente e do próximo.
<i>Backlog</i> de Portfólio	<i>Continuous Initiate</i>	Prover o <i>backlog</i> do portfólio do trimestre corrente e do próximo.
<i>Backlog</i> do Programa com Dependências Mapeadas	<i>Continuous Initiate</i>	Prover o <i>backlog</i> do programa com as dependências de negócio e/ou técnicas mapeadas entre os times envolvidos no programa.
Métricas Geradas	<i>Continuous Initiate</i>	Prover métricas ágeis.
Hipóteses e <i>Features</i>	<i>Continuous Discovery</i>	<i>Backlog</i> que possui todas as hipóteses e <i>features</i> identificadas com base no portfólio do programa.
<i>Discovery Backlog</i>	<i>Continuous Discovery</i>	<i>Backlog</i> que possui todas as histórias de usuário do programa no trimestre atual.
Métricas Geradas	<i>Continuous Discovery</i>	Prover métricas ágeis.
<i>Backlog</i>	<i>Continuous Delivery</i>	<i>Backlog</i> que possui todas as histórias de usuário da iteração atual.
Incremento de <i>Software</i>	<i>Continuous</i>	Incremento liberado e potencialmente capaz de ser utilizado

	<i>Delivery</i>	pelos clientes
Métricas Geradas	<i>Continuous Delivery</i>	Prover métricas ágeis.
Incrementos de <i>Software</i> Integrados	<i>Continuous Delivery</i>	Incrementos liberados, integrados e potencialmente capaz de serem utilizados pelos clientes.
Métricas Geradas	<i>Continuous Release</i>	Prover métricas ágeis.
<i>Release</i> Mensal	<i>Continuous Release</i>	Versão liberada mensalmente ou conforme demanda.
<i>Release</i> Trimestral	<i>Continuous Release</i>	Versão liberada trimestralmente ou conforme demanda.
Plano de Mudanças	<i>Continuous Improvement</i>	Documento contendo todas as mudanças identificadas para o trimestre atual e/ou anterior.
Métricas Geradas	<i>Continuous Improvement</i>	Prover métricas ágeis.

Fonte: elaborado pelo autor, 2023.

A geração de métricas em um *framework* com as dimensões do FMA necessita de um controle de onde a métrica é gerada e possa ser visualização ao longo dele. Dessa forma, a rastreabilidade das métricas no FMA podem ajudar na compreensão da origem na métrica e o impacto dela no futuro, principalmente quando não se olha para ela com a devida atenção. Além disso, a rastreabilidade de métricas ajuda a estender ou modificar a métrica para refletir um cenário novo no projeto ágil. Assim, a Tabela 15 apresenta as métricas do FMA com a marcação de quais etapas do FMA as métricas são geradas (MG – Métricas Geradas) e as etapas onde as métricas são apenas visualizadas (MV – Métricas Visualizadas). Além disso, a Tabela 15 considera a Categoria das Métricas para categorização do foco da medição. Assim, a Tabela 15 é um instrumento capaz de fornecer junto com o FMA a rastreabilidade da origem da métrica e em quais etapas do *framework* possam ser visualizadas. Essa rastreabilidade é um importante instrumento para compreender onde o resultado positivo ou negativo da métrica foi gerado para que as pessoas saibam onde tomar decisões sobre melhorias e ações no projeto.

Tabela 17- Métricas rastreáveis por Etapas e Categorização como ferramenta gerencial do Framework de Métricas Ágeis (FMA). Medida Gerada (MG); Métrica Visualizada (MV)

Métricas	Categoria das Métricas	Etapas do FMA				
		<i>Continuous Initiate</i>	<i>Continuous Discovery</i>	<i>Continuous Delivery</i>	<i>Continuous Release</i>	<i>Continuous Improvement</i>
Percentual de Metas	Monitoramento	MG/MV	MV	MV	MV	MV

Estratégicas Entregues	Estratégico					
Total de Itens em <i>Discovery</i> versus em <i>Delivery</i>	Monitoramento Operacional	MV	MG/MV	MG/MV	MV	MV
Eficiência do Fluxo	Fluxo de Trabalho	MV	MV	MG/MV	MV	MV
Frequência de <i>Deploy</i> da <i>Release</i>	Entrega Contínua	MV	MV	MV	MG/MV	MV
<i>Net Promoter Score</i>	<i>Feedback</i> do Cliente	MV	MV	MV	MV	MG/MV
<i>Throughput</i>	Entrega Contínua	MV	MV	MG/MV	MV	MV
Motivo da Escolha do Produto versus Nível de Satisfação	<i>Feedback</i> do Cliente	MV	MV	MV	MV	MG/MV
Quantidade de Defeitos Escapados por MVP	<i>Feedback</i> do Cliente	MV	MV	MV	MV	MG/MV
Total de Hipóteses Validadas e Aceitas	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Total de Hipóteses Validadas e Rejeitadas	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Percentual de Itens do <i>Backlog</i> do Portfólio Entregues	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Percentual de Itens do <i>Backlog</i> do Programa Entregues	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Agregação de valor	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
<i>Lead Time</i> - Comprometimento Programa	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Qualidade por Visão Empresarial ou Visão de Produto	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Orçamento dedicado à análise ou verificação do cliente	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Projetos orientados para o cliente	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Orçamento Estimado em Portfólio vs Custo Realizado da Entrega	Monitoramento Estratégico	MG/MV	MV	MV	MV	MV
Custo do atraso	Monitoramento Operacional	MG/MV	MV	MV	MV	MV
Custo hora real e hora planejada	Monitoramento Operacional	MG/MV	MV	MV	MV	MV
Capacidade, desempenho e maturidade do processo	Monitoramento de Processo	MG/MV	MV	MV	MV	MV

Esforço, desempenho e risco do projeto	Monitoramento Operacional	MG/MV	MV	MV	MV	MV
Total de Itens com Dependência e Vinculados a Riscos Identificados	Monitoramento Operacional	MG/MV	MV	MV	MV	MV
Total de Hipóteses Não Validadas	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Total de Itens de <i>Discovery</i> Descartados em <i>Delivery</i>	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Nível de Alinhamento da Funcionalidade	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Total de Itens Descartados em <i>Discovery</i>	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Quantidade de <i>commits</i>	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
<i>Product Backlog Rating</i>	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Avaliação da qualidade das histórias de usuário	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Total de Protótipos Funcionais validados por Teste A/B	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Envolvimento e Propriedade do Cliente	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Aplicabilidade e Rastreabilidade	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Clareza do Artefato	Monitoramento Operacional	MV	MG/MV	MV	MV	MV
Qualidade dos Eventos do <i>Scrum</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Impedimentos	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Sprint Burn Down</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Epic Burn Down</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Control Chart</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Reabertos	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Atribuição de <i>bugs</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Nível de Confiabilidade	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Nível de Eficiência e Eficácia	Monitoramento Operacional	MV	MV	MG/MV	MV	MV

Nível de Eficácia	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Nível de Integridade	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Nível de Manutenibilidade	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Nível de Portabilidade	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Complexidade por acoplamento	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Complexidade por coesão	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Chidamber-Kemerer</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Mean Time to Execution</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Source Lines of Code</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Coupling between Objects</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Response for a Class</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Weighted Methods per Class</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Complexidade ciclomática	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Qualidade do código-fonte	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Code smells</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Bugs</i> resolvidos por unidade de tempo	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Componentes reutilizáveis	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Status</i> desenvolvimento	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Percentual de Revisão de Código-fonte Aprovada e Reprovada	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Percentual de Revisão de Código-fonte Não Realizada	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Percentual de Problemas na Integração Contínua	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Tempo Médio de Fila	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Limite de WIP Médio	Monitoramento Operacional	MV	MV	MG/MV	MV	MV

<i>Work Item Age</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Elegibilidade do incremento	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Média de <i>Work Item Age</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Lead Time</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Cycle Time</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Velocity</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Média de <i>Cycle Time</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Elegibilidade do incremento	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Cobertura dos cenários de teste e histórias de usuário	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Quantidade de Defeitos por Iteração	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Total de Cenários de Teste Executados e Não Executados	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Total de Cenários de Teste Automatizados e Manuais	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Média de <i>Throughput</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Moda de <i>Throughput</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
Média do <i>Lead Time (Little Law)</i>	Monitoramento Operacional	MV	MV	MG/MV	MV	MV
<i>Lead Time</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Crescimento nas vendas de novos produtos	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Entrega antecipada e frequente de software	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Aspectos relacionados ao esforço	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Qualidade por <i>Release</i> ou Incremento	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Percentual de Problemas na Integração Contínua dos Times	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Probabilidade de Prazo de Entrega (<i>Forecasting</i>)	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Tempo Médio de	Monitoramento	MV	MV	MV	MG/MV	MV

Resolução de <i>Bugs</i>	de Lançamento					
<i>Throughput versus</i> Objetivo da Iteração.	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Média do <i>Lead Time</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
<i>Throughput versus</i> <i>RoadMap</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Moda de <i>Throughput</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Média de <i>Throughput</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
Média de <i>Lead Time</i>	Monitoramento de Lançamento	MV	MV	MV	MG/MV	MV
<i>Previsão por Planning</i> <i>Poker</i>	Monitoramento de Lançamento	MV	MV	MV	MV	MG/MV
Qualidade por <i>Sprint</i>	Monitoramento Operacional	MV	MV	MV	MV	MG/MV
<i>Throughput versus</i> Objetivos da Iteração.	Monitoramento Operacional	MV	MV	MV	MV	MG/MV
Receita por Usuário	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV
Percentual de Retenção de Clientes do Produto	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV
Percentual de Novos Clientes do Produto	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV
Percentual de Retenção de Clientes por MVP	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV
Percentual de Novos Clientes por MVP	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV
Tempo Médio de Resolução de <i>Bugs</i> por MVP	<i>Feedback</i> de Cliente	MV	MV	MV	MV	MG/MV

Fonte: elaborado pelo autor, 2023.

O FMA é um *framework* de métricas que foram atribuídas em cada etapa do FMA para medir e dar visibilidade sobre o estado das decisões e produto ao longo do fluxo. Dado que o FMA é uma proposta flexível, novas métricas podem ser adicionadas ou modificadas para atender a um mercado específico ou a organização. Assim, a Tabela 18 apresenta as métricas do FMA com o devido detalhamento das Métricas do FMA com a referência dos estudos 2 e 4 realizados nessa pesquisa, principais interessados com os papéis do FMA, etapas do FMA e descrição de cada métrica. É importante salientar que as métricas apresentadas na Tabela 18 são sugeridas para o FMA, mas nada impede da liderança e da organização de

definir suas próprias métricas, desde que mantenha a estrutura do FMA com os papéis, cerimônias e artefatos.

Tabela 18- Detalhamento das Métricas do FMA com a referência dos estudos realizados nessa pesquisa, principais interessados, etapas e descrição.

Etapa do FMA	Interessados/Papéis do FMA	Métricas do FMA	Referência	Descrição das Métricas
Continuous Initiate	Agile Metrics Owner Agile Portfolio Owner Agile Business Manager Agile Program Owner	Percentual de Metas Estratégicas Entregues	Estudo 4	Percentual de metas estratégicas entregues no final do trimestre anterior.
		Percentual de Itens do <i>Backlog</i> do Portfólio Entregues	Estudo 4	Percentual de itens de <i>Backlog</i> do Portfólio entregues no final do trimestre anterior.
		Percentual de Itens do <i>Backlog</i> do Programa Entregues	Estudo 4	Percentual de itens de <i>Backlog</i> do Programa entregues no final do trimestre anterior.
		Agregação de valor	Estudo 2	Medir o retorno do investimento como uma forma relevante de medir a agregação de valor
		<i>Lead Time</i> - Comprometimento Programa	Estudo 4	Medição do <i>Lead Time</i> desde o comprometimento do Programa até a entrega da <i>release</i> de itens do trimestre atual.
		Qualidade por Visão Empresarial ou Visão de Produto	Estudo 2	Medir o nível de qualidade do software alinhado com a Visão Empresarial ou Visão de Produto.
		Orçamento dedicado à análise ou verificação do cliente	Estudo 2	Medir os orçamentos com dedicação de análise e verificação do cliente.
		Projetos orientados para o cliente	Estudo 2	Medir o número de projetos criados e orientados para o cliente.
		Orçamento Estimado em Portfólio vs Custo Realizado da Entrega	Estudo 4	Medir a relação entre o custo estimado versus realizado da <i>release</i> .
		Custo do atraso	Estudo 2	Custo de atraso de entregas do projeto durante a rodada da <i>Sprint</i> .
Custo hora real e hora planejada	Estudo 2	Custo hora real e planejada de entregas do projeto durante a rodada da <i>Sprint</i> .		

		Capacidade, desempenho e maturidade do processo	Estudo 2	Medir a capacidade, desempenho e maturidade do processo ágil.
		Esforço, desempenho e risco do projeto	Estudo 2	Medir o esforço, desempenho e risco do projeto ágil.
		Total de Itens com Dependência e Vinculados a Riscos Identificados	Estudo 4	Medir o total de itens com dependências e que estejam vinculados com os riscos identificados durante a etapa de <i>Initiate</i> /Programa.
Continuous Discovery	Agile Business Owner Agile Coach Agile Software Developer	Total de Itens em <i>Discovery</i> versus em <i>Delivery</i>	Estudo 4	Medir o balanceamento de itens fluindo por <i>discovery</i> e <i>delivery</i> .
		Total de Hipóteses Não Validadas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> não validadas em <i>delivery</i> e <i>release</i> .
		Total de Hipóteses Validadas e Rejeitadas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> e validadas e rejeitadas em <i>delivery</i> e <i>release</i> .
		Total de Hipóteses Validadas e Aceitas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> validadas e aceitas em <i>delivery</i> e <i>release</i> .
		Total de Itens de <i>Discovery</i> Descartados em <i>Delivery</i>	Estudo 4	Medir o número de itens em <i>discovery</i> que foram descartados em <i>delivery</i> .
		Nível de Alinhamento da Funcionalidade	Estudo 2	Medir o nível de alinhamento das funcionalidades com as estratégias organizacionais.
		Total de Itens Descartados em <i>Discovery</i>	Estudo 4	Medir o número de itens em <i>discovery</i> que foram descartados em <i>discovery</i> .
		Quantidade de <i>commits</i>	Estudo 2	Quantidade de novos códigos-fonte publicados.
		<i>Product Backlog Rating</i>	Estudo 2	Medir o nível de prioridade do item do <i>backlog</i> .
		Avaliação da qualidade das histórias de usuário	Estudo 2	Determina a qualidade da redação das histórias de usuário, a definição de prioridade e o risco das histórias de usuário, os relacionamentos de histórias de usuário com fontes de requisitos,

				requisitos não-funcionais e versões de produtos.
		Total de Protótipos Funcionais validados por Teste A/B	Estudo 4	Medir o total de Protótipos Funcionais validados por Teste A/B para validação de hipóteses de negócio e/ou UX/UI.
		Envolvimento e Propriedade do Cliente	Estudo 2	Medir o envolvimento e propriedade do artefato produzido pelo projeto.
		Aplicabilidade e Rastreabilidade	Estudo 2	Medir a aplicabilidade e rastreabilidade do artefato produzido pelo projeto.
		Clareza do Artefato	Estudo 2	Medir a clareza do artefato produzido pelo projeto.
Continuous Delivery	Agile Business Owner Agile Coach Agile Software Developer	Eficiência do Fluxo	Estudo 2	Medir quantos itens do <i>backlog</i> são entregues por ciclo.
		Qualidade dos Eventos do <i>Scrum</i>	Estudo 2	Medir a qualidade dos eventos do <i>Scrum</i> , como <i>Sprint Plannig</i> , <i>Daily</i> , <i>Sprint Review</i> e <i>Sprint Retrospective</i> .
		Impedimentos	Estudo 2	Impedimentos mapeados durante a rodada da <i>Sprint</i>
		<i>Sprint Burn Down</i>	Estudo 2	Métrica sobre pontos de história no <i>Sprint</i>
		<i>Epic Burn Down</i>	Estudo 2	Métrica sobre pontos de história no Épico
		<i>Control Chart</i>	Estudo 2	Medir o <i>Cycle Time</i> (ou <i>Lead Time</i>) na <i>Sprint</i>
		Reabertos	Estudo 2	Medir o número de tarefas ou incidentes reabertos. Itens reabertos podem aumentar o <i>time to market</i> .
		Atribuição de <i>bugs</i>	Estudo 2	Métrica sobre atribuição da pessoa mais indicada para a resolução do problema e reduzindo o <i>time to market</i> do produto.

		Nível de Confiabilidade	Estudo 2	Medir o nível de confiabilidade em termos de qualidade de software.
		Nível de Eficiência e Eficácia	Estudo 2	Medir o nível de eficiência em termos de qualidade de software.
		Nível de Eficácia	Estudo 2	Medir o nível de eficácia em termos de qualidade de <i>software</i> .
		Nível de Integridade	Estudo 2	Medir o nível de integridade em termos de qualidade de software.
		Nível de Manutenibilidade	Estudo 2	Medir o nível de manutenibilidade em termos de qualidade de <i>software</i> .
		Nível de Portabilidade	Estudo 2	Medir o nível de portabilidade do software com outras arquiteturas de <i>software</i> .
		Complexidade por acoplamento	Estudo 2	Medir a complexidade por acoplamento que define o número de recursos de <i>software</i> por solução.
		Complexidade por coesão	Estudo 2	Medir a complexidade por coesão que define o número de coesão intra componentes de <i>software</i> .
		<i>Chidamber-Kemerer</i>	Estudo 2	Análises de complexidade de programas orientados a objetos em Java.
		<i>Mean Time to Execution</i>	Estudo 2	Medir o desempenho do código com o tempo médio de execução
		<i>Source Lines of Code</i>	Estudo 2	Medir o número de linhas do código-fonte do programa.
		<i>Coupling between Objects</i>	Estudo 2	Representa o número de classes acopladas a uma determinada classe.
		<i>Response for a Class</i>	Estudo 2	Número total de métodos que potencialmente podem ser executados em resposta a uma

			mensagem recebida por um objeto de uma classe.
	<i>Weighted Methods per Class</i>	Estudo 2	Medir a soma da complexidade dos métodos em uma classe.
	Complexidade ciclomática	Estudo 2	Medir a complexidade ciclomática (complexidade do código-fonte).
	Qualidade do código-fonte	Estudo 2	Medir a qualidade do código-fonte.
	<i>Code smells</i>	Estudo 2	Medir potenciais problemas em código-fonte.
	<i>Bugs</i> resolvidos por unidade de tempo	Estudo 2	Métrica que olha para a quantidade de <i>bugs</i> resolvidos por unidade de tempo.
	Componentes reutilizáveis	Estudo 2	Medir o número de componentes de software reutilizáveis dentro ou entre projetos.
	<i>Status</i> desenvolvimento	Estudo 2	Fornecer um rastreamento que mostre o status do desenvolvimento de produtos de <i>software</i> .
	Percentual de Revisão de Código-fonte Aprovada e Reprovada	Estudo 4	Medir o percentual de revisão de código-fonte aprovada e reprovada. A revisão auxilia na redução de <i>bugs</i> , melhora a qualidade e ajuda no aprendizado.
	Percentual de Revisão de Código-fonte Não Realizada	Estudo 4	Medir o percentual de revisão de código-fonte não realizada.
	Percentual de Problemas na Integração Contínua	Estudo 4	Medir o percentual de problemas identificados durante o processo de Integração Contínua de cada time.
	Tempo Médio de Fila	Estudo 4	Medir o tempo médio de fila nos estágios de desenvolvimento de <i>software</i> .
	Limite de WIP Médio	Estudo 4	Medir o limite médio de trabalho em progresso.
	<i>Work Item Age</i>	Estudo 4	Tempo do requisito nos estágios de

			desenvolvimento.
	Elegibilidade do incremento	Estudo 2	Avalia o impacto das práticas de gerenciamento de projetos nos dois primeiros <i>Sprints</i> .
	Média de <i>Work Item Age</i>	Estudo 4	Média de <i>Work Item Age</i> (Tempo do requisito nos estágios de desenvolvimento).
	<i>Lead Time</i>	Estudo 2	Medir a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente.
	<i>Cycle Time</i>	Estudo 2	Medir o tempo para realizar um item do <i>backlog</i> em etapas dentro da iteração.
	<i>Velocity</i>	Estudo 2	Medir quantos pontos de história foram feitos por <i>Sprint</i>
	Média de <i>Cycle Time</i>	Estudo 4	Média de <i>Cycle Time</i> (Medir o tempo para realizar um item do <i>backlog</i> em etapas dentro da iteração).
	Elegibilidade do incremento	Estudo 2	Avalia o impacto das práticas de gerenciamento de projetos nos dois primeiros <i>Sprints</i> .
	Cobertura dos cenários de teste e histórias de usuário	Estudo 2	Cobertura dos cenários de teste e histórias de usuário que auxiliaram no desempenho do processo de teste.
	Quantidade de Defeitos por Iteração	Estudo 4	Medir a quantidade de defeitos identificados nos testes por Iteração.
	Total de Cenários de Teste Executados e Não Executados	Estudo 4	Medir o total de cenários de teste executados e não Executados por iteração.
	Total de Cenários de Teste Automatizados e Manuais	Estudo 4	Medir o total de cenários de teste automatizados e manuais por iteração.
	<i>Throughput</i>	Estudo 4	Medir a taxa de vazão do time por iteração.

		Média de <i>Throughput</i>	Estudo 4	Medir a média da taxa de vazão do time por iteração.
		Moda de <i>Throughput</i>	Estudo 4	Medir a moda da taxa de vazão do time por iteração.
		Média do <i>Lead Time (Little Law)</i>	Estudo 4	Medir a média do <i>Lead Time</i> (entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente).
<i>Continuous Release</i>	<i>Agile Program Owner Agile Metrics Owner Agile Business Manager Agile Business Owner Agile Coach Agile Software Developer</i>	<i>Lead Time</i>	Estudo 4	Medir a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente
		Crescimento nas vendas de novos produtos	Estudo 2	Medir o aumento nas vendas de novos produtos.
		Entrega antecipada e frequente de software	Estudo 2	Medir a relevância da produtividade entregando software com antecedência e frequência.
		Aspectos relacionados ao esforço	Estudo 2	Medir a relevância o esforço de implantação de software que resolva os problemas de clientes.
		Qualidade por <i>Release</i> ou Incremento	Estudo 2	Medir o nível de qualidade do software por <i>Release</i> ou incremento.
		Percentual de Problemas na Integração Contínua dos Times	Estudo 4	Medir o percentual de problemas identificados durante o processo de Integração Contínua dos times.
		Probabilidade de Prazo de Entrega (<i>Forecasting</i>)	Estudo 4	Medir a probabilidade de prazo de entrega dos itens distribuídos pelo planejamento do <i>roadmap</i> .
		Tempo Médio de Resolução de <i>Bugs</i>	Estudo 4	Medir o tempo médio dos times em resolução de <i>bugs</i> na <i>release</i> .
		Frequência de <i>Deploy</i> da <i>Release</i>	Estudo 4	Medir a frequência com que os times realizam o <i>deploy</i> da <i>release</i> .
		<i>Throughput versus</i> Objetivo da Iteração.	Estudo 4	Compara a taxa de vazão <i>versus</i> os objetivos de entrega da Iteração.

		Média do <i>Lead Time</i>	Estudo 4	Medir a média do <i>Lead Time</i> (entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente).
		<i>Throughput versus RoadMap</i>	Estudo 4	Compara a taxa de vazão <i>versus o planejamento do roadmap</i> .
		Moda de <i>Throughput</i>	Estudo 4	Calcula a moda da taxa de vazão.
		Média de <i>Throughput</i>	Estudo 4	Calcula a média da taxa de vazão.
		<i>Throughput</i>	Estudo 4	Medir a taxa de vazão.
		Média de <i>Lead Time</i>	Estudo 4	Medir a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente.
Continuous Improvement	Agile Metrics Owner Agile Portfolio Owner Agile Business Manager Agile Program Owner Agile Business Owner Agile Coach Agile Software Developer	<i>Net Promoter Score</i>	Estudo 2	Medir o nível de satisfação dos clientes em relação do produto recebido e sua experiência com ele.
		<i>Previsão por Planning Poker</i>	Estudo 2	Medir o nível de compreensão e assertividade da estimativa.
		Qualidade por <i>Sprint</i>	Estudo 2	Medir o nível de qualidade do software por <i>Sprint</i> .
		<i>Throughput versus</i> Objetivos da Iteração.	Estudo 4	Compara a taxa de vazão <i>versus</i> os objetivos de entrega da Iteração.
		<i>Throughput versus RoadMap</i>	Estudo 4	Compara a taxa de vazão <i>versus o planejamento do roadmap</i> .
		Moda de <i>Throughput</i>	Estudo 4	Calcula a moda da taxa de vazão.
		Média de <i>Throughput</i>	Estudo 4	Calcula a média da taxa de vazão.
		Média de <i>Lead Time</i>	Estudo 4	Medir a média do <i>Lead Time</i> (entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente).

		Total de Hipóteses Não Validadas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> não validadas em <i>delivery</i> e <i>release</i> .
		Total de Hipóteses Validadas e Rejeitadas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> e validadas e rejeitadas em <i>delivery</i> e <i>release</i> .
		Total de Hipóteses Validadas e Aceitas	Estudo 4	Medir se hipóteses definidas em <i>discovery</i> validadas e aceitas em <i>delivery</i> e <i>release</i> .
		Total de Itens de <i>Discovery</i> Descartados em <i>Delivery</i>	Estudo 4	Medir o número de itens em <i>discovery</i> que foram descartados em <i>delivery</i> .
		Total de Itens Descartados em <i>Discovery</i>	Estudo 4	Medir o número de itens em <i>discovery</i> que foram descartados em <i>discovery</i> .
		Motivo da Escolha do Produto <i>versus</i> Nível de Satisfação	Estudo 4	Comparação entre as pesquisas do motivo da escolha do produto <i>versus</i> o nível de satisfação.
		Receita por Usuário	Estudo 4	Medir o valor da receita obtida por usuário do <i>software</i> .
		Percentual de Retenção de Clientes do Produto	Estudo 4	Medir o percentual de retenção de clientes do <i>software</i> .
		Percentual de Novos Clientes do Produto	Estudo 4	Medir o percentual de novos clientes do <i>software</i> .
		Percentual de Retenção de Clientes por MVP	Estudo 4	Medir o percentual de retenção de clientes por MVP.
		Percentual de Novos Clientes por MVP	Estudo 4	Medir o percentual de novos clientes por MVP.
		Tempo Médio de Resolução de <i>Bugs</i> por MVP	Estudo 4	Medir o tempo médio de resolução de <i>bugs</i> por MVP
		Quantidade de Defeitos Escapados por MVP	Estudo 4	Medir a quantidade de defeitos escapados por MVP

Fonte: elaborado pelo autor, 2023.

Na próxima seção, serão apresentados os temas relacionados com a avaliação do produto tecnológico, resultados e contribuições, assim como a intenção do registro de propriedade intelectual fruto desse trabalho de pesquisa de tese.

8 PRODUTO TÉCNICO TECNOLÓGICO

Esta tese propôs como produto tecnológico uma patente de invenção em linha com os estudos e resultados empíricos apresentados. A patente de invenção tem como objetivo ser responsável por orientar a liderança corporativa na gestão de projetos ágeis. O produto tecnológico é classificado de acordo com os critérios estabelecidos pelo GT 06 – Qualis tecnológico de 2016, conforme estabelecido pela Portaria CAPES 171/2018. O objetivo principal desta portaria era “desenvolver uma metodologia de avaliação da produção técnica e tecnológica que pudesse ser aplicada a todas as áreas de avaliação” (CAPES, 2019b). A avaliação do produto técnico tecnológico se baseia em cinco critérios: aderência, impacto, aplicabilidade, inovação e complexidade.

Em se tratando da **aderência**, o produto tecnológico gerado por esta tese se enquadra na linha de pesquisa 2 “Gerenciamento de Projetos”, vinculado ao Programa de Pós-Graduação em Gestão de Projetos – PPGP UNINOVE. A presente tese também está alinhada com outras pesquisas de seu orientador e vinculadas aos projeto eixo “Projetos Ágeis e Híbridos”. Sendo assim, a aderência deste produto tecnológico é alta para o programa de pós-graduação ao qual está vinculado e sua respectiva linha de pesquisa. Isso se deve à sua aplicabilidade na gestão de projetos e à conexão direta com a área de conhecimento das ciências sociais aplicadas.

A avaliação do **impacto** está ligada às alterações que os produtos técnicos tecnológicos provocam no meio em que são implementados (CAPES, 2019). Para avaliar adequadamente os produtos propostos, é necessário compreender por que eles foram criados e qual é o principal objetivo de sua aplicação, o que permite perceber em que domínio as alterações são detectadas (CAPES, 2019). O produto tecnológico gerado é de uso gerencial e pode ser aproveitado por qualquer organização ativa no setor de tecnologia da informação, que é um dos líderes na adoção de abordagens ágeis. O verdadeiro valor do produto, entretanto, reside na maneira como suporta e melhora o gerenciamento de projetos que segue princípios ágeis. Portanto, ele tem o potencial não apenas para influenciar, mas também para elevar o padrão das práticas ágeis no gerenciamento de projetos em todo o setor. Assim, para essa tese, o impacto é considerado alto uma vez que o produto gerado tem potencial de ser utilizado por qualquer organização do segmento de tecnologia da informação como suporte ao processo de gerenciamento de projetos sob práticas ágeis.

A **aplicabilidade**, que significa o quão facilmente um produto tem potencial de ser usado para cumprir os objetivos para os quais foi criado (CAPES, 2019), está diretamente associada à extensão de sua aplicabilidade e ao seu potencial de replicação (CAPES, 2019). O produto tecnológico desta tese demonstra aplicabilidade em várias esferas, desde organizações de TI consolidadas até *startups* e pequenas e médias empresas. Com sua facilidade de uso, potencial de replicação e relevância direta para as práticas de gerenciamento de projetos ágeis, este produto pode ser uma ferramenta valiosa para otimizar processos de negócios e promover uma cultura ágil nessas empresas. Assim, a aplicabilidade para o produto tecnológico é denominada média, uma vez que o modelo ainda não adotado, até esse momento, em ambiente produtivo.

O critério de **inovação** é definido como a capacidade de criar ou alterar algo (CAPES, 2019). Segundo a CAPES, a avaliação da inovação apenas considera a geração de novos conhecimentos, sem levar em conta usabilidade, complexidade, impacto ou quaisquer outros aspectos da produção. O produto resultante desta tese recebe uma classificação de inovação média, pois, embora derive de conceitos já estabelecidos, tem um potencial de aplicabilidade considerável. A solução desenvolvida para a classe de problemas pode ser relevante para outros indivíduos que enfrentam desafios similares durante o controle e monitoramento de seus projetos sob as práticas ágeis ou em uma fase de transição das práticas tradicionais para as práticas ágeis, demonstrando assim seu potencial de replicabilidade.

A **complexidade** é definida como ‘uma característica ligada à variedade de participantes, laços e conhecimentos necessários para a criação e evolução dos produtos’ (CAPES, 2019). De acordo com a abordagem da CAPES (2019), o sistema sugerido abrange o desenvolvimento com a união de diversos tipos de conhecimento e a participação de vários atores.

Neste contexto, foi reivindicado a autoria de uma patente de invenção sob o título “Método para controlar e monitorar projetos de *software*” como produto tecnológico desta tese. Na Figura 9, consta parte do comprovante do depósito da patente de invenção, cujo depósito foi realizado no dia 11/12/2023 no Instituto Nacional da Propriedade Industrial (INPI).



Pedido nacional de Invenção, Modelo de Utilidade, Certificado de Adição de Invenção e entrada na fase nacional do PCT

Número do Processo: BR 7

Dados do Depositante (71)

Depositante 1 de 2

Nome ou Razão Social: JOSE DA SILVA AZANHA NETO

Figura 9 - Pedido de Patente de invenção

Fonte: Elaborado pelo autor, 2023.

O produto resultante desta tese é apresentado como complexidade alta, uma vez que, até o momento desta pesquisa, é o único produto tecnológico que abrange todas as etapas do *The Agile Landscape – Deloitte* (TAL), que correspondem as etapas de *Initiate, Discovery, Delivey e Release* visa medir todos os aspectos do projeto desde a concepção de uma ideia ou hipótese até a avaliação e *feedback* do produto de *software* contemplado pelo cliente.

9 CONCLUSÃO

Ao longo dos anos o ambiente organizacional apresenta grandes desafios onde a gestão de projetos está presente com o papel fundamental na condução de projetos da organização entorno dela mesma ou de seus produtos e serviços. A liderança tem um papel fundamental na gestão de projetos ágeis, mas ainda existem desafios a serem explorados, pois a liderança está presente nos níveis operacionais do projeto, inclusive como liderança servidora no mesmo nível. Os projetos ágeis são medidos em nível operacional por métricas ágeis, mas com a contraposição da liderança que controla projetos a partir de métricas tradicionais, o que abriu uma lacuna de pesquisa que foi explorada nesse trabalho de pesquisa de tese.

Assim, a tese focou no objetivo de propor um *framework* de métricas ágeis para orientar a liderança na gestão de projetos ágeis para responder à questão de pesquisa desta tese, que compreendia em saber “Como as métricas ágeis podem orientar a liderança

corporativa na gestão de projetos ágeis?”. Como a pesquisa seguiu os procedimentos metodológicos baseados em estudos, o Estudo 1 seguiu uma abordagem de RSL com o objetivo de identificar como os estudos sobre as práticas ágeis de gestão de projetos abordam a liderança. O Estudo 2 foi outra RSL, onde obteve-se o mapeamento de artigos que abordaram métricas em todo o ciclo de vida de um projeto tradicional e ágil. O Estudo 3 identificou como está o estado da base de patentes em relação aos modelos propostos para processos e *frameworks* de métricas ágeis. A realização do Estudo 3 demonstrou a lacuna identificada por esse trabalho de pesquisa de tese e deu origem a versão inicial do *framework* de métricas ágeis para a liderança corporativa. Por fim, o Estudo 4 seguiu com a realização de entrevistas com a abordagem do *focus group* para ajustes, melhorias e *feedback* sobre a proposição do *framework* de métricas ágeis por meio dos praticantes em gestão ágil de projetos em diversos segmentos da TI em empresas e instituições.

Como resultados, verificou-se que ainda existem obstáculos para implementar a abordagem ágil nas organizações, mas que práticas baseadas na liderança ágil e na liderança servidora podem auxiliar no desenvolvimento ágil nos projetos. Nesse sentido que a comunicação, governança, cultura, liderança, processos e pessoas precisam estar engajadas no processo de transformação ágil nas organizações e a liderança deve apoiar nesse processo. Isto é, implementar a agilidade vai muito além da criação e desenvolvimento de times de projetos apenas para rodar o *Framework Scrum* ou o Método *Kanban* e aplicar métricas operacionais para se tornar ágil.

Além disso, percebeu-se nos resultados que as métricas tradicionais focaram em apresentar a métrica do *Earned Value Management* (EVM) com a adoção de melhorias na métrica, como *Earned Value Management/Earned Schedule* (EVM/ES) e o *Earned Duration Management* (EDM) como inspiração. Nas métricas ágeis, percebeu-se que as métricas como *Lead Time*, *Cycle Time*, relacionadas à qualidade de *software* e o *Velocity* são as mais citadas nas publicações, e se concentraram no *discovery* e no *delivery* do projeto ágil. Assim, esses resultados denotam a existência de uma lacuna no uso pela liderança corporativa de métricas tradicionais e ágeis de projetos que atendam todas, ou partes interligadas, das etapas de um ambiente ágil de projetos como as etapas de *Initiate*, *Discovery*, *Delivery* e *Release* sugerido pelo modelo do TAL.

Referente aos resultados da pesquisa sobre a base de patentes realizado pelo Estudo 3, os dados sobre as patentes publicadas foram extraídos pelas bases PATENTSCOPE, ESPACENET e LENS. Dessa maneira, não foram encontradas soluções relevantes sobre

processos ou *frameworks* para métricas ágeis nas bases de patente, o que indicou uma oportunidade para desenvolvimento dessa pesquisa nesse sentido e este trabalho apresenta o FMA como uma ferramenta para auxílio a liderança corporativa em projetos ágeis através de métricas ágeis em níveis de Portfólio, Programa e Times de Projeto.

O *focus group* realizado por meio do Estudo 4 trouxe algumas melhorias e ajustes sugeridos pelos praticantes em gestão de projetos ágeis convidados e atuantes em diferentes organizações e setores da TI. Dentre as contribuições dos praticantes ao FMA, pode-se citar as melhorias no processo do *Continuous Release* na comunicação da entrega da *release*, retorno de *feedback* e métricas de *Continuous Improvement* para todas as etapas do FMA, inclusão de categorização e rastreamento de métricas. Além disso, incluiu-se a estrutura de entrada de metas e projetos estratégicos da organização como interface ao FMA, a inclusão de alguns processos em *Continuous Discovery* e *Continuous Delivery* para declarar alguns eventos do FMA. No tocante aos papéis, sugeriu-se a criação do papel de arquitetura de soluções e de software em nível corporativo, destacando-se que o detalhamento da arquitetura é emergente e sob a gestão do Time de Projeto. Por fim, destaca-se a possibilidade de adoção em partes do FMA e por diferentes organizações (cliente e fornecedor), entre outras contribuições. Isto é, pode-se implementar o FMA somente com o *Continuous Discovery*, *Continuous Delivery* e *Continuous Release* considerando uma abordagem exclusivamente operacional do projeto. Assim, considerando uma abordagem de gestão de projetos ágeis em nível organizacional, pode-se incluir as partes do *Continuous Initiate* e *Continuous Improvement*.

A versão final do FMA trouxe um *framework* mais robusto para lidar com necessidades da liderança corporativa no controle e monitoramento do desempenho do projeto por meio das métricas ágeis em todas as etapas do projeto em nível operacional e organizacional. Isto é, o FMA pode auxiliar na medição de projetos desde a iniciação em nível de portfólio e programa, *discovery*, *delivery*, *release* e na melhoria contínua. Assim, um processo para melhoria contínua seria medindo com o planejando ações para a melhoria contínua do projeto e produto. Além disso, o FMA fomenta o planejamento de mudanças entorno da revisão constante de metas e projetos estratégicos da organização, de produtos e serviços oferecidos aos seus clientes internos e externos.

Como contribuição teórica, a realização desta pesquisa procurou aprofundar na discussão do uso de métricas ágeis contemplando todas as etapas do *The Agile Landscape – Deloitte (TAL)*, que correspondem as etapas de *Initiate*, *Discovery*, *Delivey* e *Release*, frente a

relevância do tema destacada durante toda a realização desta pesquisa. Vale lembrar que até a realização desta pesquisa nenhum processo ou *framework* de métricas ágeis, tanto em bases de dados acadêmicas quanto patentária apresentava tal solução.

Assim, com a realização desta pesquisa, espera-se contribuir para apresentar as organizações o FMA como um *framework* robusto e flexível na adoção de outras métricas e na sua implementação, e que seja capaz de fornecer as métricas ágeis em todas as etapas do projeto para a liderança corporativa e equipes envolvidas nos projetos. Além de fornecer métricas de projeto, espera-se que as métricas de portfólio, programa, *discovery* e *delivery*, release, *feedback* de clientes, *stakeholders* e do mercado possam auxiliar a liderança corporativa e a equipe do projeto na tomada de decisão sobre o projeto ágil através de monitoramento, planejamento, ações e revisões de melhoria contínua. Assim, espera-se também que o FMA auxilie a liderança corporativa e a equipe de projeto a obter o *feedback* sobre a condução do projeto e sobre a avaliação de clientes e do mercado sobre o produto ou serviços o mais cedo possível. Nesse sentido que o FMA fornece uma visibilidade de métricas em todas as etapas para que pessoas e organizações possam tomar decisões o mais cedo possível entre refutar ou perseverar na estratégia organizacional ou na estratégia de produto ou serviço.

Para a prática, o *framework* do FMA poderá ser encapsulado em um aplicativo vinculado as ferramentas de mercado de gestão de projetos com o objetivo de coletar as métricas durante as etapas de concepção, descoberta, desenvolvimento e lançamento do produto ao mercado ou na produção. Para a teoria, espera-se apresentar um conjunto de métricas ágeis que controle e monitore o ciclo de vida dos projetos ágeis, desde a etapa de concepção (*Continuous Discovery*), pela fase de desenvolvimento (*Continuous Delivery*) e a fase de entrega contínua (*DevOps*), o que até o momento é um avanço inédito nas práticas gerenciais de projetos ágeis.

Além das contribuições para a academia e para a prática, o produto tecnológico desenvolvido nesta pesquisa de tese está alinhado aos esforços em direção aos Objetivos de Desenvolvimento Sustentável (ODS) estabelecidos pela ONU. O produto tecnológico descreve um sistema inovador de gerenciamento de projetos que visa fortalecer a cooperação global para o desenvolvimento sustentável das organizações. A invenção é projetada para maximizar a eficiência e eficácia das colaborações multinacionais, promove especificamente o Objetivo de Desenvolvimento Sustentável (ODS) 17 da ONU - Parcerias para as Metas. Esta ferramenta de gerenciamento de projetos pode permitir que organizações ao redor do

mundo gerencie projetos de forma eficiente, valorizando a partilha de conhecimentos, a transferência de tecnologia e a mobilização de recursos.

De modo geral é possível considerar que os objetivos da pesquisa foram alcançados, bem como respondida à questão por meio da realização dos estudos acima citados. O *framework* de métricas ágeis para a liderança corporativa se apresenta como uma ferramenta útil para apoio à gestão de projetos por meio da melhor adoção das métricas ágeis em todas as etapas do projeto em nível operacional e organizacional.

Como limitação dessa pesquisa, não foram abordadas pesquisas entorno do custo para implantação do FMA ou partes dele. Nesse sentido, esse trabalho de pesquisa de tese não abordou o custo para a aplicação dos artefatos, papéis e para adoção do FMA nos níveis essencial e completo, separadamente. Além disso, o FMA não foi testado juntamente as principais práticas de mercado como *Scrum*, Método *Kanban*, OKR e *DevOps*. Nesse sentido que, para trabalhos futuros, uma pesquisa poderia ser conduzida nas organizações com objetivo de se obter uma avaliação de custo com a adoção do FMA e avaliar o comportamento do mesmo com a adoção de práticas mais conhecidas no mercado como o *Scrum*, *Kanban*, OKR e *DevOps*. Por fim, o FMA poderia ser testado com a adoção de novas métricas ágeis para avaliar o fluxo, o rastreamento de métricas e as reuniões de avaliação de métricas para analisar o comportamento do FMA diante de novas métricas sugeridas pela organização ou pela liderança corporativa.

10 APÊNDICE

10.1 THE IMPORTANCE OF LEADERSHIP IN AGILE PROJECTS: SYSTEMATIC LITERATURE REVIEW

A IMPORTÂNCIA DA LIDERANÇA EM PROJETOS ÁGEIS: REVISÃO
SISTEMÁTICA DA LITERATURA

LA IMPORTANCIA DEL LIDERAZGO EN PROYECTOS ÁGILES: REVISIÓN
SISTEMÁTICA DE LA LITERATURA

Received: 03/15/2022 | Reviewed: 03/22/2022 | Accept: 04/06/2022 | Published: 04/12/2022

José da Silva Azanha Neto
ORCID: <https://orcid.org/0000-0001-9602-8283>
Universidade Nove de Julho, Brazil
E-mail: jose.azanha@uni9.edu.br

Renato Penha
ORCID: <https://orcid.org/0000-0002-1431-2860>
Universidade Nove de Julho, Brazil

E-mail: renato.penha@uni9.pro.br
Luciano Ferreira da Silva
ORCID: <https://orcid.org/0000-0001-6482-8729>
Universidade Nove de Julho, Brazil
E-mail: luciano.ferreira@uni9.pro.br

Isabel Cristina Scafuto
ORCID: <https://orcid.org/0000-0002-6788-3325>
Universidade Nove de Julho, Brazil
E-mail: isabelscafuto@uni9.pro.br

ABSTRACT

In an environment marked by the presence of agile practices for project management, the leadership role needs to present competencies that involve changes concerning market conditions and business uncertainties. They are responses to rapid changes and agility as an organizational competence to deal with these environmental factors surrounding the life cycle of projects. The objective of this article was to identify how studies on agile project management practices approach leadership. As a methodological strategy, a Systematic Literature Review was adopted to assist in the mapping and evaluation of a specific intellectual structure to develop a body of knowledge. For data collection, the Web of Science and Scopus databases were used. Initially, 120 articles were found. The results showed that there are still obstacles to implementing the agile approach in the organizations, but that practices based on agile leadership and servant leadership can help agile development in the projects. The contribution to adopting agile leadership and servant leadership is to facilitate transforming traditional organizations into agile organizations.

Keywords: Agile; Agility; Agile leadership; Servant leadership; Software development.

RESUMO

Em um ambiente marcado pela presença das práticas ágeis para a gestão de projetos, o papel de liderança precisa apresentar competências que envolvam as mudanças em relação às condições de mercado e às incertezas de negócio. São respostas às rápidas mudanças e a agilidade como competência organizacional para lidar com esses fatores ambientais entorno do ciclo de vida dos projetos. O objetivo deste artigo foi identificar como os estudos sobre as práticas ágeis de gestão de projetos abordam a liderança. Como estratégia metodológica foi adotado uma Revisão Sistemática de Literatura para auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento. Para a coleta de dados foram utilizadas as bases Web of Science e Scopus. Inicialmente, foram encontrados 120 artigos. Os resultados apontaram que ainda existem entraves para o processo, mas que práticas pautadas na liderança ágil e a liderança servidora em projetos de desenvolvimento ágil ainda apresentam desafios importantes nas organizações estabelecidas como organizações ágeis e como organizações tradicionais.

Palavras-chave: Ágil; Agilidade; Liderança ágil; Liderança servidora; Desenvolvimento de software.

RESUMEN

Em um ambiente marcado pela presença das práticas ágeis para a gestão de projetos, o papel de liderança precisa apresentar competências que envolvam as mudanças em relação às condições de mercado e às incertezas de negócio. São respostas às rápidas mudanças e a agilidade como competência organizacional para lidar com esses fatores ambientais entorno do ciclo de vida dos projetos. O objetivo deste artigo foi identificar como os estudos sobre as práticas ágeis de gestão de projetos abordam a liderança. Como estratégia metodológica foi adotado uma Revisão Sistemática de Literatura para auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento. Para a coleta de dados foram utilizadas as bases Web of Science e Scopus. Inicialmente, foram encontrados 120 artigos. Os resultados apontaram que ainda existem entraves para o processo, mas que práticas pautadas na liderança ágil e a liderança servidora em projetos de desenvolvimento ágil ainda apresentam desafios importantes nas organizações estabelecidas como organizações ágeis e como organizações tradicionais.

Palavras-chave: Ágil; Agilidade; Liderança ágil; Liderança servidora; Desenvolvimento de software.

1. INTRODUCTION

Project agility is the ability of a project team to quickly adapt to changes, review the project's direction and deliver value to the customer in short cycles of planning, development, and continuous delivery (Sutherland, 2019; Conforto & Amaral, 2016). In the context of agile practices, organizations and customers benefited from delivering value as early as possible through the realization of projects (Lichtenthaler, 2020; Walter-Güpner, 2018).

To adopt agile practices, organizations need to be prepared concerning possible culture changes, management processes, as well as in the relationship of leaders with the team, customers, and stakeholders (Alqudah & Razali, 2017). Among the frameworks most adopted by organizations, Scrum is a framework for dealing with complex and adaptive problems,

characterized by values, roles, artifacts, and events (Colomo-Palacios *et al.*, 2012). Regarding roles and responsibilities, Scrum is formed by the Product Owner (PO), the Developers and the Scrum Master (SM). The PO is responsible for the leadership and management of products, stakeholders and customers. Developers lead the development and delivery of the functional quality software increment at the end of each Sprint (Sutherland, 2019). The SM plays the role of a servant leader and ensures the correct application of Scrum, as well as being a facilitator for the PO and Developers (Anwer *et al.*, 2017; Hron & Obwegeser, 2018).

The KanBan method has as main objective to present a panoramic view of the activities to be conducted for the entire team, with principles and values that can be adopted in any production line (Hoda & Murugesan, 2016). The Kanban method is widely used in software development projects. Although the method does not prescribe roles and you can start work with it the way the organization operates (Majchrzak & Stilger, 2017), Kanban suggests that the organization has product, flow, and technical leadership (Dos Santos *et al.*, 2018).

When the leadership comes to roles and responsibilities of people in project management positions in both frameworks, Gjøystdal and Karunaratne (2020) highlight that there is no centrality of command about leadership. In this context, Pacheco *et al.* (2018) state that the leader needs to present new skills for the relationship with the team, customers, and stakeholders. Mergel (2016), and Shamim *et al.* (2016) highlight that in an environment of agile project management practices, the person in a leadership role, regardless of the framework, must ensure a collaborative environment, good relationship with the team, in addition to encouraging the ability of each one to act during the life cycle of the projects, thus giving rise to the term servant leadership.

Based on what was presented about the need for the leader to participate in the relationship with the team, customers, and stakeholders in an environment under agile practices, this research aims to answer the following question: "how studies on agile project management practices approach leadership?" The objective of this article was to identify how studies on agile project management practices approach leadership. To answer this question, a Systematic Literature Review - SLR was adopted as a methodological strategy. The justification for this choice is given by the role of SLR in helping to map and evaluate a specific intellectual structure to develop a body of knowledge (Tranfield *et al.*, 2003). In addition, SLR is a methodological procedure that uses literature as the main source of data (Sampaio & Mancini, 2007).

In the following section, the methodological procedures are presented. Subsequently, the results are presented and, finally, the final remarks are presented, which describe the findings and limitations of this research, as well indicate for future studies.

2. METHODOLOGY

The research presented here adopted an SLR as a method to understand the convergence of two relevant themes, which are agile practices and leadership. SLR differs from traditional narrative reviews by adopting a systematic scientific process that is replicable and transparent. In this sense, its use is also justified because it minimizes the bias

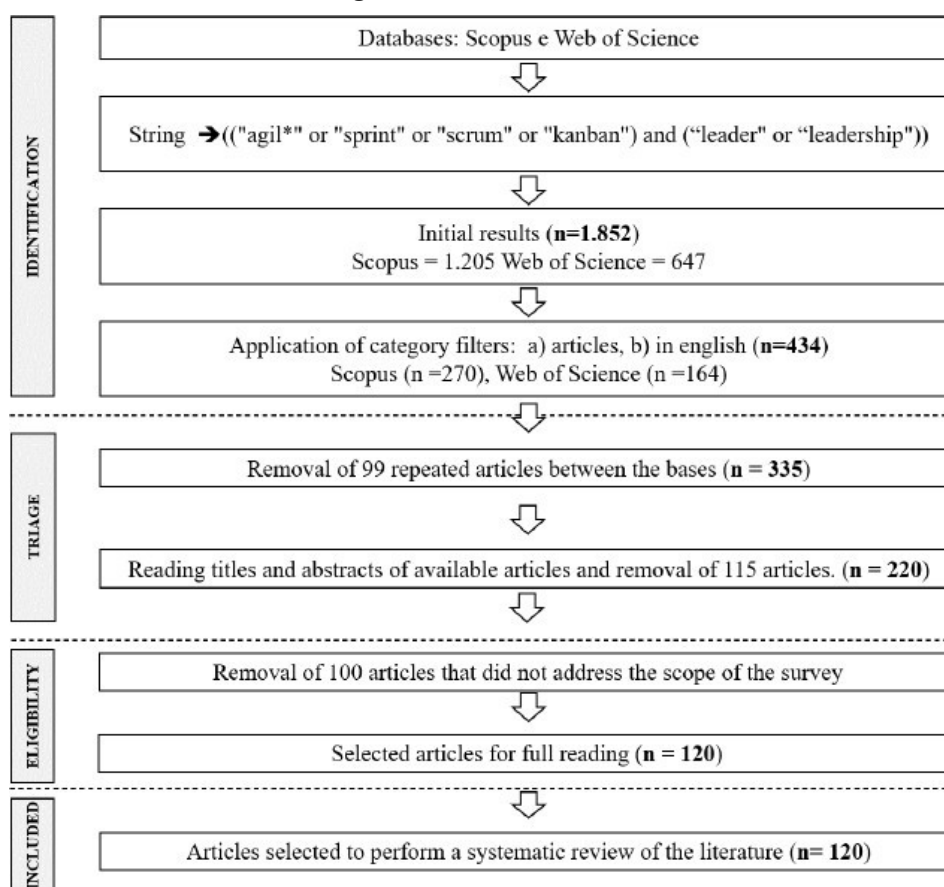
in the construction of a theoretical analysis corpus, as well as the possibility of building an audit trail of the decisions and procedures applied in this research (Cook *et al.*, 1997).

The procedures for carrying out this SLR followed six phases according to the prescriptions of Pollock and Berge (2018): (i) clarifying research goals and objectives; (ii) seeking relevant research; (iii) collecting data to incorporate in the analysis corpus; (iv) assess the quality of studies selected; (v) synthesizing the evidence from articles; (vi) interpret the findings after analysis process. The objective of the phases and activities presented is to guarantee the rigor and robustness that are aimed at in this type of research. Following this phase, the first step was motivated by the question that guides this research, which was: “how do studies on agile project management practices approach leadership?.” For this purpose, the academic databases Web of Science and Scopus were used as research sources, as they are one of the main databases for accessing research published in social sciences.

The string used to perform the searches was (((“agil*” or “sprint” or “scrum” or “kanban”) AND (“leader” or “leadership”))). The survey was conducted on May 1, 2021. The use of the Boolean operators “and” and “or,” in addition to the use of the symbol “*” allows for greater scope, and control in the construction of the research base. The operators are applied to consider the intersection of the two research areas studied. The use of the asterisk incorporates all variations of the word in the position after it is found. It is noteworthy that no temporal filter was applied so that it was possible to map all the production on the themes studied.

Therefore, after the first phase, which comprises the establishment of objectives, the researchers moved into the application of the search string. The results found in the first round underwent analysis and screening, as is shown in Figure 1, respecting the proposal by Pollock and Berge (2018), who present four steps for the elaboration of the analysis corpus.

Figure 1. Search results in data sources.



Source: adapted from Pollock and Berge (2018).

Figure 1 presents the four stages for the elaboration of the analysis corpus. The first step highlights the string used to search the databases selected. The second step is the screening of results to identify the articles corresponding to the research proposal. In the third stage, the eligibility criteria were applied, where the articles were evaluated according to the inclusion and exclusion criteria. And finally, the fourth stage constituted the corpus of analysis, resulting in the sample database of articles for the research.

For the composition of the database, some filters were applied during the initial research, such as the criterion of only articles in journals, excluding articles in congresses, books, among others. This exclusion was defined because these studies do not undergo a peer review in some cases, or even due to the maturity presented in working papers, in addition to mitigating the redundancy of works presented at congresses and published in journals. The areas delimited for this research were: “Management” and “Business” for Web of Science, “Business” for the Scopus database.

After the consolidation of the databases, repeated articles were removed to eliminate redundancy. Subsequently, the database was treated with the aid of Excel spreadsheets. This software allowed performing data analysis and presenting the results from the combination of quantitative information by frequency analysis, as well as qualitative information by categorizing the contents of the articles. This research phase also allowed us to present a relevant descriptive analysis of the study conducted.

From the reading of titles and abstracts in the screening phase to verify the eligibility of articles, some inclusion and exclusion criteria were established, highlighted in Figure 2. After verifying and reading the abstracts and introductions of the 345 previously selected articles, the final base was composed of 120 accepted articles that formed the research corpus.

Figure 2. Inclusion and Exclusion Criteria.

Inclusion Criteria	Reason for Inclusion
Articles that conceptualize the studied constructs	Allow addressing research purposes: understanding the constructs studied according to the articles selected.
Articles that address the relationships between the constructs studied	Allow contemplating the alignment of articles: understanding the interdependencies and relationships between the constructs.
Published articles	Offer greater rigor in the arguments and theoretical contributions studied.
Exclusion Criteria	Reason for Exclusion
Articles focusing on Quality, Marketing, Health, Finance, or other purposes outside the given constructs.	Exclude articles that are not focused on the questions that will offer insights to meet the research objectives.
Articles without a relevant theoretical foundation, or with a low relationship with the constructs.	One of the purposes of the study is to obtain future research perspectives, through theoretical knowledge existing in a structure, for which theoretical assumptions are prerequisites.

Source: Prepared by the authors (2021).

The next step was to read the 120 articles contained in the corpus of analysis, categorizing the contents in Excel spreadsheets to present a grouping of findings and comparison of categories. The activities applied in this phase are in line with the prescriptions of Pollock and Berge (2018) in phases (v) synthesizing the evidence and (vi) interpreting the findings. Although some quantitative treatments were applied, in this research, the qualitative analysis of the articles was prioritized to constitute a matrix that could represent the findings of this study.

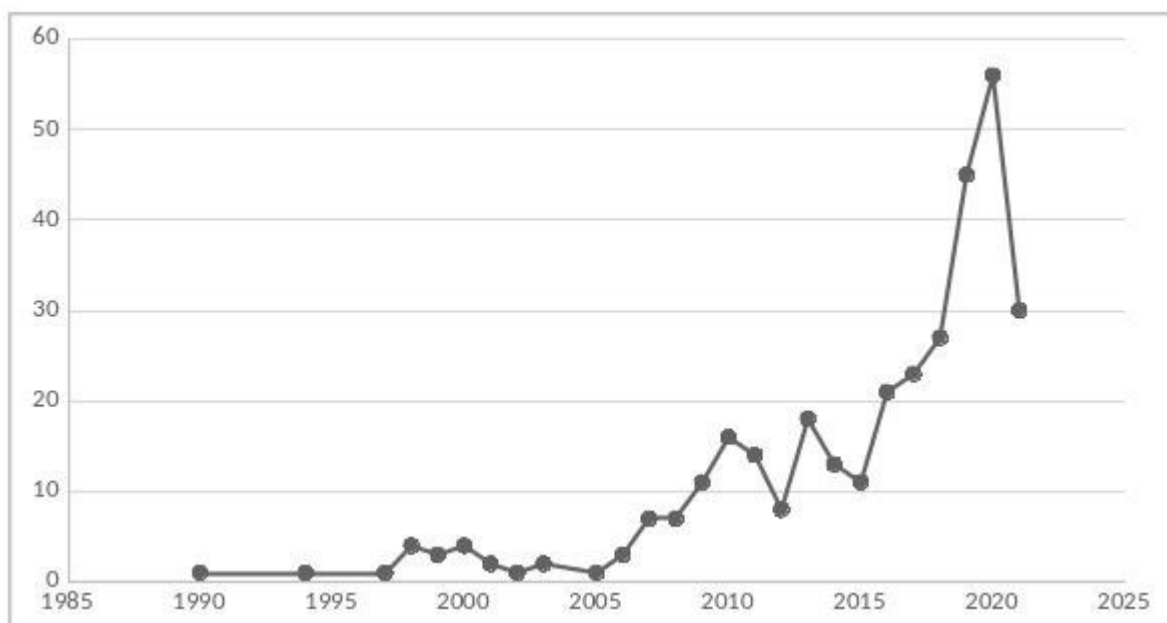
3. RESULTS AND DISCUSSION

In this section, the results of this SLR are presented after conducting the analysis processes. Initially, the mapping of the articles that constituted the corpus of analysis is presented. Following, the categories highlighted after an in-depth analysis of the contents of the articles are presented.

3.1 EVOLUTION OF ARTICLE PUBLISHED

The articles selected from the Web of Science and Scopus databases were screened and the analysis corpus consisted of 120 articles. The articles underwent a more in-depth analysis, which allowed us to understand the studies on the relationship between agile practices and leadership. The 120 articles analyzed are temporally situated between the years 1990 to May 2021, shown in Figure 3.

Figure 3. Temporal analysis of articles.



Source: Research data (2021).

As can be seen from the analysis of Figure 1, the publication of articles dealing with the relationship between agile practices and leadership remained low until 2005, with an average of 2 articles per year. Despite the drops in 2012 and 2015 to 8 and 11 articles respectively, the following years show a growth in research in which there is an intersection of these two themes. A relevant aspect is that the year 2021 is still being finalized at the time

of finalizing this article, but this year it already has 30 articles published until May, a number close to the average of 34.4% of the last five years.

After analyzing the temporal distribution of articles, we also sought to understand the frequency of articles per journal. Figure 4 shows the journals with more than two publications and the respective number of articles published on the topic, ordered by the largest number of publications and by the alphabetical order of the name of the journals.

Figure 4. Journals and number of publications on the subject.

#	Journals	Quantity
1	Harvard Business Review	8
2	Mis Quarterly Executive	7
3	California Management Review	5
4	Journal of Business Research	3
5	Journal of Leadership Studies	3
6	Management Decision	3
7	Benchmarking-An International Journal	2
8	Business Horizons	2
9	Industrial and Commercial Training	2
10	International Journal of Operations & Production Management	2
11	International Journal of Organizational Leadership	2
12	International Journal of Project Management	2
13	Journal of Asian Finance Economics and Business	2
14	Project Management Journal	2
15	Revista de Gestao e Projetos	2

Source: Research data (2021).

Regarding the number of articles published per journal, the journals considered most useful, according to research data, are: 'Harvard Business Review' with 8 articles; 'MIS Quarterly Executive' with 7 articles and "California Management Review" with 5 articles.

3.2 ANALYSIS AND DISCUSSION OF THE CATEGORIES FOUND

After mapping the articles that constituted the corpus of analysis, an in-depth analysis of the published contents was conducted. Reading and categorizing the contents led to the classification of articles into 4 categories concerning the object of this research. The categories are described in Figure 5.

Figure 5. Classification of articles into 4 categories.

Categories	Authors
Agile leadership	Wanasida <i>et al.</i> , 2021; Dabić <i>et al.</i> , 2021; Malik; Sarwar & Orr, 2021; Agarwal <i>et al.</i> 2021; Busse & Weidner, 2020; Kim, You & Hong, 2020; Holtzhausen & De Klerk, 2018; Srivastava & Jain, 2017; Hall & Rowland, 2016.
Agile Leadership Competency	Holweg & Maylor, 2018; Garcia & Russo, 2020; Perides, Barrote & Sbragia, 2021; Kaufmann, Kock & Gemünden, 2020; Gonçalves <i>et al.</i> , 2020; Anke & Ringeisen, 2021; Dries & Pepermans, 2012.
Leadership for Empowerment	Grass, Backmann & Hoegl, 2020; Kappelman <i>et al.</i> , 2019; Bäcklander, 2019; Parker, Holesgrove & Pathak, 2015.
Leadership in an Agile Organization	Tariq & Abonamah, 2021; Koch & Schermuly, 2020; Annosi <i>et al.</i> , 2020; Tronvoll <i>et al.</i> , 2020; Akkaya & Tabak, 2020; Vanharanta <i>et al.</i> , 2018; Lichtenthaler, 2020; Holbeche, 2019; Antonacopoulou <i>et al.</i> , 2019; Reitz <i>et al.</i> , 2020.

Source: Research data (2021).

The four categories found comprise an abstraction observed in the readings of the articles. It is worth mentioning that the initial analysis process of the articles was individual and, later, for the selection of the categories, the analysis of the researchers of this research was conducted. The categorization process was conducted by discussing and establishing a consensus on what was the greatest adherence of all articles in each of the constituted categories.

In the next section, the categories will be discussed to explain the findings of this research. The categories presented here do not only represent a way of explaining and organizing the contents studied, but also opportunities for creating research agendas.

3.2.1 Agile Leadership

Wanasida *et al.* (2021), Busse and Weidner (2020), and Hall and Rowland (2016) discuss the relationship of transformational leadership in the organization as important factors for performance, learning, and organizational agility. In this case, the leaders inspired and motivated their teams to create innovations and changes that will help the company to grow and develop towards organizational agility, especially in an environment of volatility, uncertainty, complexity, and ambiguity (VUCA). Dabić *et al.* (2021) demonstrate the role of entrepreneurial leaders with the intellectual agility that positively influences the organization's innovative capacity, and the innovation capacity involves both the project and the project teams. Malik, Sarwar, and Orr (2021) argue that agile practices of team

autonomy and agile communication contributed to empowerment, improving innovative behavior and project performance.

In sharing knowledge and project management practices such as Waterfall and agile, Agarwal *et al.* (2021) and Antonacopoulou *et al.* (2019) stated that culture for organizational learning and the flexible and collaborative organization supports knowledge sharing and the adoption of agile practices. This culture facilitates a vertical or distributed leadership style in a VUCA environment. To support collaborative leadership, Kim *et al.* (2020) suggested that, for the establishment of an agile culture in an organization, the exercise of servant leadership is important. For the authors, it is leadership that maximizes the potential of the organization's members based on respect, support, and rewards provided by the organization. Furthermore, for servant leadership in Scrum teams, Holtzhausen and De Klerk (2018) confirmed the importance of servant leadership skills when identifying and developing SM and Team Leaders, indicating the formal role of team leader in Scrum teams, and implementing agile practices effectively.

According to Parker, Holesgrove, and Pathak (2015), and Srivastava and Jain (2017), the role of servant leadership in self-organized and distributed teams can be challenging. In addition, they added that the multicultural influence of geographically distributed teams as a growing factor in organizations can influence the success of the self-organized team. These teams include task types and complexities, dynamics, and performance, which is an explicit definition of the self-organizing team and appropriate leadership models. In this sense, the authors proposed a framework for leadership to deal with self-organized, multicultural, and geographically distributed teams.

The articles in this category presented important discussions about the role of leadership in agile project management practices. There is an important involvement of agile leadership in the motivation for innovation and improvements aimed at leadership in an agile organization. An involvement of servant leadership, collaboration, and facilitation, supporting teams to self-organize and use flexible management techniques and agile frameworks.

3.2.2 Agile Leadership Competency

Holweg and Maylor (2018) highlighted that agile leadership can make important contributions to project management and that the use of tools, such as Lean and Theory of Constraints, can help organizations achieve greater value in projects. Since streamlining processes and reducing waste can reflect positively on the added value. Garcia and Russo (2020) identified the influence of the type of leadership on the performance of the project team, according to the methods applied in the management of software development projects. The authors identified three leadership styles, being transactional, transformational, and empowering leadership. Such leaders relate positively to the performance of the team. However, in this study, it was found that agile or predictive

project management methods did not influence the relationship between leadership and team performance.

Perides *et al.* (2021) brought an analysis of five competencies in agile or predictive management projects considered the most important, namely: integrity and personal reliability; Communication; teamwork; strategy and relationships; and team engagement. Close to these competencies, Dries and Pepermans (2012) published a list of leadership competencies that include: emotional agility; humor; efficient delegation; and quality of feedback. These competencies reflect in the team as they communicate openly because they have fun at work and have compassion for each other.

Regarding managerial practices, Gonçalves *et al.* (2020) showed concern in identifying the necessary competencies for managing projects that used the traditional and agile approach that require distinct team characteristics, especially in a technology-intensive context that requires high performance. The authors conducted a case study in an EdTech, which is an educational startup, with company employees who composed two different teams. The work concluded that the team is on the way to being a high-performance team, however, it is still necessary for members to incorporate more agile practices and agile leadership, understanding the real value they generated for customers, through the development of your products. In this sense, Anke and Ringeisen (2021) analyzed the competence requirements and leadership success criteria of project managers, who oversee agile software development teams.

In 2017, Kappelman *et al.* (2019) identified that agile software development ranks sixth in the top ten of the most important and hardest-to-find technical skills. Anke and Ringeisen (2021) presented several competencies focused on leadership, and how to motivate and promote people. The authors also brought the skills to interact and present adequate communication and conflict management to analyze and interpret the specialized knowledge in agile software development project management. The authors also identified that support and cooperation favor teamwork skills and that both conceptualizing and creating these relationships result in favoring openness. In addition, it was highlighted that organization and execution support resource management, and that adaptation and coping to deal with errors outlines an entrepreneurial and performance profile favoring software development.

The articles presented important discussions about leadership competencies in agile project management practices. The selected articles presented leadership styles such as transactional and transformational that positively influence the performance of those they lead. Competencies such as integrity, reliability, communication, teamwork, strategy, relationship, and engagement were identified in the surveys regardless of whether the projects are conducted in the agile or predictive model. Thus, these competencies are aligned with the competencies expected by the roles present in agile project models.

3.3 LEADERSHIP FOR EMPOWERMENT

Research by Grass, Backmann, and Hoegl (2020) developed a model of the continuous process of agile team innovation that revealed the importance of states of leadership empowerment and team adaptability. The authors found that empowerment is not a static state but emerges through interactions between various stakeholders. Thus, the involvement of leadership and staff in activities that have influences from the customer and the organizational environment caused the increase and decrease of empowerment. Hence, agile transformation and top management supportive behaviors played a significant role

that affected the empowerment dynamics and resulted in the adaptability of the team. The authors also revealed the dynamic role of empowerment and adaptability constructs for agile innovation processes, both to create conditions for empowerment and to foster adaptability.

For Bäcklander (2019), the conditions for empowerment can be the key balancing force in leadership in complex environments, such as in software development project environments. In this way, the authors showed that the work of agile coaches is important in enabling leaders to understand the context of others, supporting other leaders in establishing principles of group dynamics and teamwork. In addition, the work of agile coaches supports leadership in the transparency of conflicts, facilitating and encouraging constructive dialogue to balance leadership autonomy and the strategic alignment of organizations.

The leadership for empowerment presented in the articles in this category showed that the theme empowerment plays a key point in agile project management, which can influence teams to innovate and adapt throughout the project. Thus, leaders or agile coaches who foster empowerment in teams showed greater results from agile project management practices in organizations.

3.4 LEADERSHIP IN AN AGILE ORGANIZATION

Tariq and Abonamah (2021) discussed in their article that agility in organizations is especially derived from scope changes for customers, market changes, and technological changes, in addition to social, cultural, and competition development. This referred to an important ranking of business agility, where innovation, agility, adaptability and flexibility, and facilitation skills are critical for agile organizations.

Annosi *et al.* (2020), Akkaya and Tabak (2020), Vanharanta *et al.* (2018), and Holbeche (2019) sought to understand how agile methods can influence the way an agile organization learns, adapts, and changes. Therefore, the authors point out that the role of leadership in an agile organization develops with a collaborative, inspiring, trusting, resilient, and cooperative environment among employees and for the development of executive leaders. As suggested by Reitz *et al.* (2020), based on practical implications of the mindfulness method, which represents mindfulness of every movement, situation, and emotion.

Tronvoll *et al.* (2020) assessed the level of organizational agility and identified gaps between organizational needs and the needs of agile teams in this organizational learning process. In this sense, Tronvoll *et al.* (2020) suggested that the agile mindset is critical for transforming the organization's processes to move from planning to discovery, from data scarcity to shared data abundance, and from the hierarchy of jobs to the partnership.

Lichtenthaler (2020) highlighted those isolated applications of Design Thinking and Lean Startup in projects will not promote the benefits of agile innovation organizations. Thus, the authors found that executives are seeking the benefits of agility by jointly executing Design Thinking in front-end software development teams and Lean Startup in back-end software development teams.

From the perspective of an agile project management portfolio in organizations, Kaufmann *et al.* (2020) measured the agility capability of organizations. The research presented the recognition of the emerging strategy in organizations, that is, according to the empirical mentality of process control to make strategic decisions based on what is known.

Thus, the emerging strategy is positively linked to the success of the agile project management portfolio.

From the point of view of the adoption of practices by leadership in agile organizations, Koch and Schermuly (2020) studied the probability of attracting candidates and retaining employees because the organization performs project management using agility. The research showed a greater ability of organizations to retain and attract candidates in companies that adopt agile practices in project management.

The articles in this category showed that the topic of leadership in an agile organization plays a key role in leadership in agile project management and in the way, teams innovate and adapt throughout the project, and how leadership in the agile organization retains its talents. In this sense, the articles brought up topics such as business agility, innovation, agility, adaptability and flexibility, and facilitation skills. The role of leadership in an agile organization develops with a collaborative, inspiring, trusting, resilient and cooperative environment, in addition to the empirical mindset of process control of making strategic decisions. Thus, leaders, or agile coaches, who foster empowerment at various levels of leadership in teams showed greater results from agile project management practices and the strategic alignment of organizations.

The authors' articles showed that the organization's influence is fundamental to promote flexibility, adaptability, business agility, innovation for the agility of organizations that seek to deliver greater value to customers. In addition, organizations that promote a collaborative, inspiring, trusting, resilient, and cooperative environment among employees, attract and retain employees, and develop executive leaders. That is, they are agile organizations that learn, adapt, and change, especially those that understand agile methods and practices, such as Design Thinking and Lean Startup in conducting project management.

4. CONCLUSION

From the research conducted in the Web of Science and Scopus databases, SLR found articles that addressed aspects related to agile leadership, agile leadership competence, leadership for empowerment, and leadership in an agile organization, and addressed the relationships between the studied constructs.

The results showed that there are still obstacles to the process of applying agile practices, but activities that consider agile leadership and servant leadership in agile can improve this reach. In the same way, projects development in general still present challenges in organizations established as agile organizations and as traditional organizations as well. Feedback, communication, humor in pressure environments, delegation, autonomy, and flexibility of both executive leadership and leadership in self-organized and cross-functional teams are cited in some articles as fundamental for agile leadership.

The term servant leadership was introduced in the book by Greenleaf (1977), and he defines the term as a professional profile that seeks, first, to serve people and then to lead them. The servant-leader must create and maintain business process restructuring initiatives in the direction of agility to support agile projects (Holtzhausen & de Klerk, 2018; Anwer *et al.*, 2017; Hron & Obwegeser, 2018). Thus, from the point of view of "The Evidence-Based Management Guide" (EBM, 2020), servant leadership is not just about SM, the organization and top leadership must assume the role of servant leadership to support SM

in the transformation of the organizational environment around agility (Spiegler *et al.*, 2019; Walter-Güpner, 2018; Xu & Shen, 2018).

Regarding organizational competencies in agility, it is understood that the servant leader needs to consider competencies that involve changes in market conditions, business uncertainties, and adaptability in response to rapid changes and agility. These are organizational competencies to deal with these environmental factors surrounding the life cycle of information systems (Walter-Güpner, 2018). In this sense, the great challenge for traditional organizations is how to change the traditional mindset of control and monitoring in software development projects (Sutherland, 2019). This traditional management mindset applied to agile projects causes dysfunctions in decision making, generating financial impacts on projects, costs, and value delivered to customers (Lichtenthaler, 2020; Walter-Güpner, 2018). Servant leadership is a factual competency for the development of SM and Team Leaders, however, we did not identify the same competency as necessary for executive leadership in agile software development projects. Thus, research opportunities are evident to address the role of leadership in agile project management practices. There is an important involvement of agile leadership in the motivation for innovation and improvements aimed at the agile organization, in the involvement of servant leadership, supporting teams for self-organization, and use of management techniques and agile practices.

As found in the articles, the culture for organizational learning and the flexible and collaborative organization supports knowledge sharing and the adoption of agile methods. Thus, servant leadership supports the dissemination of agile thinking in organizations. In this sense, three leadership styles were identified, being transactional, transformational, and empowerment that did not influence the existing relationship between leadership and team performance in agile software development projects, or in the predictive model of project management.

Some articles investigated the importance of organizational learning to deal with constant changes in the business environment. The articles analyzed shown that the work of agile coaches is to disseminate the necessary skills and empower agile thinking to deal with unstable, and challenging organizational environments with the VUCA environment. In leadership empowerment, the authors showed that the theme empowerment plays a key role in agile project management, furthermore in the way teams innovate and adapt throughout the project. Thus, leaders or agile coaches who foster empowerment at various levels of leadership in teams showed greater results from agile project management practices and the strategic alignment of organizations.

Agile organizations are classified as organizations that learn and respond quickly to changes, always bent on delivering value to customers and businesses. In this sense, the use of tools such as Scrum, Design Thinking, Lean Startup, and Theory of Constraints can help organizations achieve greater value delivery in projects. This thinking also promotes entrepreneurship with the intellectual agility that positively influences the innovative capacity of the organization. The organization's influence is fundamental to promote flexibility, adaptability, business agility, innovation for the agility of organizations that seek to deliver greater value to customers. In addition, promoting a collaborative, inspiring, trusting, resilient and cooperative environment among employees, attracting and retaining employees, and developing executive leaders, make them agile organizations that learn, adapt, and change, and that understand the agile methods and practices.

This work presented limitations of the role of leadership in the theme of agile project management in organizations. Thus, articles were found that discussed the role of leadership in managing people and teams, in the influence of leadership in organizations

that consider themselves agile, in competencies found in agile leaders, and in how leadership can favor empowerment.

Future work could explore whether agile leadership contributes positively to the organization to remove organizational limitations and help in the delivery of project results in software development teams that use the Scrum Framework and the Method Kanban. Despite the limitations presented, it is expected that this study can contribute to the deepening of the discussion on the themes of Agile Practices and Agile Leadership, in addition to providing a base that informs the work conducted, journals, authors, and categories as a facilitating tool for the researchers.

REFERENCES

- Agarwal, U. A., Dixit, V., Nikolova, N., Jain, K., & Sankaran, S. (2021). A psychological contract perspective of vertical and distributed leadership in project- based organizations. *International Journal of Project Management*, 39(3), 249-258.
- Akkaya, B., & Tabak, A. (2020). The link between organizational agility and leadership: A research in science parks. *Academy of Strategic Management Journal*, 19(1), 1-17.
- Alqudah, M. K., & Razali, R. (2017). Key factors for selecting an Agile method: A systematic literature review. *International Journal on Advanced Science, Engineering and Information Technology*, 7(2), 526-537.
- Anke, S., & Ringeisen, T. (2021). Kompetenzanforderungen an Führungskräfte von agilen Softwareentwicklungsteams. Gruppe. Interaktion. *Organisation. Zeitschrift für Angewandte Organisationspsychologie (GIO)*, 52(1), 51-63.
- Annosi, M. C., Martini, A., Brunetta, F., & Marchegiani, L. (2020). Learning in an agile setting: A multilevel research study on the evolution of organizational routines. *Journal of Business Research*, 110, 554-566.
- Antonacopoulou, E. P., Moldjord, C., Steiro, T. J., & Stokkeland, C. (2019). The New Learning Organisation. *The Learning Organization*.
- Anwer, F., Aftab, S., Shah, S. M., & Waheed, U. (2017). Comparative analysis of two popular agile process models: Extreme Programming and Scrum. *International Journal of Computer Science and Telecommunications*, 8(2), 1-7.
- Bäcklander, G. (2019). Doing complexity leadership theory: How agile coaches at Spotify practise enabling leadership. *Creativity and Innovation Management*, 28(1), 42-60.
- Busse, R., & Weidner, G. (2020). A qualitative investigation on combined effects of distant leadership, organisational agility, and digital collaboration on perceived employee engagement. *Leadership & Organization Development Journal*.
- Colomo-Palacios, R., González-Carrasco, I., López-Cuadrado, J. L., & García-Crespo, Á. (2012). ReSySTER: A hybrid recommender system for Scrum team roles based on fuzzy and rough sets. *International Journal of Applied Mathematics and Computer Science*, 22, 801-816.
- Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model— A hybrid framework for technology-based companies. *Journal of Engineering and Technology Management*, 40, 1-14.

- Cook, D. J., Mulrow, C. D., & Haynes, R. B. (1997). Systematic reviews: synthesis of best evidence for clinical decisions. *Annals of internal medicine*, 126(5), 376-380. <https://doi.org/10.7326/0003-4819-126-5-199703010-00006>
- Dabić, M., Stojčić, N., Simić, M., Potocan, V., Slavković, M., & Nedelko, Z. (2021). Intellectual agility and innovation in micro and small businesses: The mediating role of entrepreneurial leadership. *Journal of Business Research*, 123, 683-695.
- Donald, L. A. N. G, & Rumsey, C. (2018). Business Disruption Is Here To Stay—What Should Leaders Do? *Quality-Access to Success*, 19, 35-40
- Dos Santos, P. S. M., Beltrão, A. C., de Souza, B. P., & Travassos, G. H. (2018). On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *Journal of Software Engineering Research and Development*, 6(1), 1-29.
- Dries, N., & Pepermans, R. (2012). How to identify leadership potential: Development and testing of a consensus model. *Human Resource Management*, 51(3), 361-385.
- Garcia, F. A. Z., & Russo, R. D. F. S. M. (2020). Liderança e desempenho da equipe de desenvolvimento de software: influência do tipo de gestão de projetos. *Revista Brasileira de Gestão de Negócios*, 21, 970-1005.
- Gjøystdal, S., & Karunaratne, T. (2020). Effect of Inadequate Self-Organized Teams in Agile Project Management: A Case Study From the Oil and Gas Industry. *International Journal of Information Technology Project Management (IJITPM)*, 11(3), 95-106.
- Gonçalves, L. C. C., de Oliveira, S. A. A., Pacheco, J. D. C. A., & Salume, P. K. (2020). Competências requeridas em equipes de projetos ágeis: um estudo de caso em uma Edtech. *Revista de Gestão e Projetos*, 11(3), 72-93.
- Grass, A., Backmann, J., & Hoegl, M. (2020). From Empowerment Dynamics to Team Adaptability: Exploring and Conceptualizing the Continuous Agile Team Innovation Process. *Journal of Product Innovation Management*, 37(4), 324-351.
- Greenleaf, R. K. (1979). Servant leadership: A journey into the nature of legitimate power and greatness. *Business Horizons*, 22(3), 91-92.
- Hall, R. D., & Rowland, C. A. (2016). Leadership development for managers in turbulent times. *Journal of Management Development*.
- Hoda, R., & Murugesan, L. K. (2016). Multi-level agile project management challenges: A self-organizing team perspective. *Journal of Systems and Software*, 117, 245-257.
- Holbeche, L. (2019). Designing sustainably agile and resilient organizations. *Systems Research and Behavioral Science*, 36(5), 668-677.
- Holtzhausen, N., & de Klerk, J. J. (2018). Servant leadership and the Scrum team's effectiveness. *Leadership & Organization Development Journal*
- Holtzhausen, N., & de Klerk, J. J. (2018). Servant leadership and the Scrum team's effectiveness. *Leadership & Organization Development Journal*.
- Holweg, M., & Maylor, H. (2018). Lean leadership in major projects: from “predict and provide” to “predict and prevent”. *International Journal of Operations & Production Management*.
- Hron, M., & Obwegeser, N. (2018). Scrum in practice: an overview of Scrum adaptations. In *Proceedings of the 51st Hawaii International Conference on System Sciences*.

- Kappelman, L., Johnson, V., Torres, R., Maurer, C., & McLean, E. (2019). A study of information systems issues, practices, and leadership in Europe. *European Journal of Information Systems*, 28(1), 26-42.
- Kaufmann, C., Kock, A., & Gemünden, H. G. (2020). Emerging strategy recognition in agile portfolios. *International Journal of Project Management*, 38(7), 429-440.
- Kim, T. W., You, Y. Y., & Hong, J. W. A Study on Effect of Servant Leadership and Perceived Organizational Support on Characteristics of Agile Organizational Culture. *Research in World Economy*, 11(2), 1-12.
- Koch, J., & Schermuly, C. C. (2020). Who is attracted and why? How agile project management influences employee's attraction and commitment. *International Journal of Managing Projects in Business*.
- Lichtenthaler, U. (2020). Agile innovation: the complementarity of design thinking and lean startup. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET)*, 11(1), 157-167.
- Lichtenthaler, U. (2020). Agile innovation: the complementarity of design thinking and lean startup. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET)*, 11(1), 157-167.
- Majchrzak, M., & Stilger, L. (2017). Experience report: Introducing Kanban into automotive software project. *e-Informatica Software Engineering Journal*, 11(1).
- Malik, M., Sarwar, S., & Orr, S. (2021). Agile practices and performance: Examining the role of psychological empowerment. *International Journal of Project Management*, 39(1), 10-20.
- Mergel, I. (2016). Agile innovation management in government: A research agenda. *Government Information Quarterly*, 33(3), 516–523.
- Pacheco, A., Marín-Raventós, G., & López, G. (2018). Designing a Technical Debt Visualization Tool to Improve Stakeholder Communication in the Decision-Making Process: A Case Study. In A.M. Tjoa, M. Raffai, P. Doucek, & N.M. Novak (Orgs.), *Research and Practical Issues of Enterprise Information Systems* (p. 15–26). Cham: Springer International Publishing.
- Parker, D. W., Holesgrove, M., & Pathak, R. (2015). Improving productivity with self-organised teams and agile leadership. *International Journal of Productivity and Performance Management*. 64 (1), 112-128.
- Perides, M. P. N., Barrote, E. B., & Sbragia, R. (2021). As competências de gestores de projetos que atuam com métodos ágeis e tradicionais: um estudo comparativo. *Revista de Gestão e Projetos*, 12(1), 11-38.
- Pollock, A., & Berge, E. (2018). How to do a systematic review. *International Journal of Stroke*, 13(2), 138-156. <https://doi.org/10.1177/1747493017743796>
- EBM (2020). The Evidence-Based Management Guide (2020). Measuring Value to Enable Improvement and Agility. Scrum.org. https://scrumorg-website-prod.s3.amazonaws.com/drupal/2020-12/EBM%20Guide%202020_1.pdf?nexus-file=https%3A%2F%2Fscrumorg-website-prod.s3.amazonaws.com%2Fdrupal%2F2020-12%2FEBM%2520Guide%25202020_1.pdf >.

- Sampaio, R. F., & Mancini, M. C. (2007). Systematic review studies: a guide for careful synthesis of the scientific evidence. *Brazilian Journal of Physical Therapy*, 11(1), 83-89. <https://doi.org/10.1590/S1413-35552007000100013>
- Shamim, S., Cang, S., Yu, H., & Li, Y. (2016). Management approaches for Industry 4.0: A human resource management perspective. 2016 IEEE Congress on Evolutionary Computation (CEC), 5309–5316.
- Spiegler, S. V., Heinecke, C., & Wagner, S. (2019). Leadership gap in agile teams: how teams and scrum masters mature. In *International Conference on Agile Software Development* (pp. 37-52). Springer, Cham.
- Srivastava, P., & Jain, S. (2017). A leadership framework for distributed self-organized scrum teams. *Team performance management: An international journal*, 23 (5/6), 293-314.
- Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante.
- Sutherland, J., & Schwaber, K. (2020). The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game. Scrum.org. Recuperado em 13 de maio de 2021 de: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- Tariq, M. U., & Abonamah, A. A. (2021). Role Of Game-Based Teaching In Leadership Skills Development. *Academy of Entrepreneurship Journal*, 1-15. Tranfield, D., Denyer, D., & Smart, P. (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, 14(3), 207-222. <https://doi.org/10.1111/1467-8551.00375>
- Tronvoll, B., Sklyar, A., Sörhammar, D., & Kowalkowski, C. (2020). Transformational shifts through digital servitization. *Industrial Marketing Management*, 89, 293-305.
- Vanharanta, H., Kantola, J., Markopoulos, E., & Salo, M. (2018). The degree of agility in a technology company's strategy, management, and leadership. *Management and Production Engineering Review*, 9(4), 129-137.
- Walter-Güpner, T. (2018). Effects of Agile Leadership and Organizational Competencies on Firm Performance. Evidence-Based Recommendations for Agile Transformation in the Manufacturing Industry by Comparing Software and Manufacturing SME. *Zeitschrift für interdisziplinäre ökonomische Forschung*, (1), 87-92.
- Wanasida, A. S., Bernarto, I., Sudibjo, N., & Pramono, R. (2021). Millennial transformational leadership on organizational performance in Indonesia fishery startup. *The Journal of Asian Finance, Economics, and Business*, 8(2), 555-562.
- Xu, P., & Shen, Y. (2018). The role of leadership in agile software development. *Project Management*, 12.



RISUS - Journal on Innovation and Sustainability
volume 14, número 4 - 2023
ISSN: 2179-3565
Editor Científico: Arnoldo José de Hoyos Guevara
Editor Assistente: Vitória Catarina Dib
Avaliação: Melhores práticas editoriais da ANPAD

10.2 MAPEAMENTO DE MÉTRICAS PARA DESEMPENHO DE PROJETOS ÁGEIS: REVISÃO SISTEMÁTICA DA LITERATURA

Mapping metrics for agile project performance: systematic literature review

José da Silva Azanha Neto, Renato Penha, Marcelo Luiz do Amaral Gonçalves

Universidade Nove de Julho, Brazil

Email: jose.azanha@gmail.com, renato.penha@uni9.pro.br, marluago@gmail.com

RESUMO

Em um contexto competitivo onde as organizações precisam entregar produtos ou serviços por meio da gestão ágil de projetos, um processo em que as métricas se tornam insumos importantes para o controle e monitoramento de projetos. O que se busca nesse artigo é compreender se existem organizações que, apesar de conduzirem projetos ágeis, utilizam controle os projetos por meio do uso de métricas tradicionais. Além disso, se busca entender se a utilização de métricas de projetos preditivos dificultam a visão e o trabalho da liderança corporativa para tomar decisões em projetos ágeis. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis. Como estratégia metodológica foi adotado uma Revisão Sistemática de Literatura para auxiliar no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento. Para a coleta de dados foram utilizadas as bases *Web of Science* e *Scopus*. Foram encontrados 83 artigos e os resultados apontaram que os projetos tradicionais seguem controlando os projetos com métricas tradicionais, tal como *Earned Value Management (EVM)*. Em projetos ágeis, as métricas mais comuns são métricas associadas *backlog* do produto, *delivery* e qualidade do produto. Este estudo contribui com outras pesquisas que desejarem identificar as métricas utilizadas entre as diferentes abordagens de gestão de projetos e orientar novas pesquisas para trabalhos futuros.

Palavras-chave: Desenvolvimento de Software; Métricas Ágeis e Tradicionais; Scrum; Kanban.

ACEITO EM: 10/12/2023

PUBLICADO EM: 30/12/2023

MAPPING METRICS FOR AGILE PROJECT PERFORMANCE: SYSTEMATIC LITERATURE REVIEW

Mapeamento de métricas para desempenho de projetos ágeis: revisão
sistemática da literatura

José da Silva Azanha Neto, Renato Penha, Marcelo Luiz do Amaral Gonçalves
Universidade Nove de Julho, Brazil
Email: jose.azanha@gmail.com, renato.penha@uni9.pro.br, marluago@gmail.com

ABSTRACT

In a competitive context where organizations need to deliver products or services through agile project management, a process in which metrics become important inputs for project control and monitoring. What is sought in this article is to understand whether there are organizations that, despite conducting agile projects, use project control through the use of traditional metrics. In addition, it seeks to understand whether the use of predictive project metrics hinder the vision and work of corporate leadership to make decisions in agile projects. The objective of this article was to map which are the metrics used to measure the performance of traditional and agile projects. As a methodological strategy, a Systematic Literature Review was adopted to assist in mapping and evaluating a specific intellectual structure to develop a body of knowledge. For data collection, the Web of Science and Scopus databases were used. 83 articles were found and the results showed that traditional projects continue to control projects with traditional metrics, such as Earned Value Management (EVM). In agile projects, the most common metrics are metrics associated with product backlog, delivery and product quality. This study contributes to other studies that wish to identify the metrics used between the different project management approaches and guide new research for future work.

Keywords: Software Development; Agile and Traditional Metrics; Scrum; Kanban.

1 INTRODUÇÃO

O Modelo em Cascata (MC) foi criado com o objetivo de representar a execução de fases de um projeto de desenvolvimento de software (Royce, 1970). Nesse modelo, cada fase possui dependência com a fase anterior, onde a próxima fase somente poderá ser iniciada quando a fase antecedente for concluída (Sommerville, 2007). O MC é composto pelas fases de Levantamento de Requisitos, Análise de Requisitos, Projeto, Desenvolvimento, Testes e Implantação (Pressman, 2021). Dentre elas, as fases de Levantamento de Requisitos, Análise de Requisitos e Projeto podem exigir grandes esforços durante o planejamento e concepção sobre o escopo que será desenvolvido no projeto e esperado pelo cliente (Van Casteren, 2017).

No contexto das práticas gerenciais, os projetos orientados ao MC sob as práticas tradicionais de gestão de projetos possuem incertezas no início do projeto que podem mudar o escopo e o direcionamento do software durante o projeto (Sutherland, 2019).

Em se tratando das práticas gerenciais e o desempenho de projetos, o uso de métricas tornou-se um fator estratégico para que o controle e o monitoramento do projeto se tornem eficazes (Glenwright, 2007). No tocante às práticas tradicionais, as métricas mais utilizadas pelas organizações é o gerenciamento do valor agregado (Junior e Junior, 2019). O gerenciamento do valor agregado ou Earned Value Management (EVM), foi proposto pelo Project Management Institute (PMI, 2021). O EVM é uma técnica de gerenciamento de projetos para medir o desempenho e o progresso de projetos combinando as medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido (Glenwright, 2007; PMI, 2021). Porém, a utilização das métricas pode causar uma percepção difusa entre as práticas tradicionais e as práticas ágeis em gerenciamento de projetos de software.

Em relação ao prazo do projeto, as práticas ágeis preconizam que os prazos das atividades devem ser baseados em dados históricos de entrega da equipe de projeto, em vez de estimativas baseadas no planejamento (Sutherland, 2019). No caso do MC, os prazos são baseados nas estimativas definidas na fase de planejamento e dando origem ao cronograma de atividades (Sutherland, 2019), distorcendo assim à utilização das métricas em ambas as práticas. Conforme Choudhury (2019) e Dias e Larieira (2021), essa distorção na utilização das métricas de projetos tradicionais nos projetos ágeis se justifica devido a utilização das abordagens do triângulo de ferro (escopo, prazo e custo) em projetos tradicionais e o triângulo invertido em projetos ágeis. Choudhury (2019) e Dias e Larieira (2021) adicionaram que o triângulo de ferro aborda o escopo fixo com o prazo e o custo variáveis, enquanto no ágil o triângulo invertido trata do prazo e custo fixo com o escopo variável.

No contexto das práticas ágeis, as métricas são orientadas ao valor entregue ao cliente (Gren, Goldman & Jacobsson, 2020). De acordo com Jyothi e Rao (2017) e Budacu e Pocatilu (2018), as métricas ágeis mais utilizadas são (i) velocity; (ii) Work Item Age (WIA); (iii) Throughput, (iv) Cycle Time (CT) e (v) Lead Time (LT). O velocity é uma métrica utilizada no desempenho dos desenvolvedores por meio dos pontos de história do usuário. O WIA é uma métrica para medir o tempo em dias que um requisito está em uma determinada fase do desenvolvimento. Já o Throughput é uma métrica para avaliar da capacidade de entrega da equipe de projeto ao final de cada iteração. O CT conta o tempo em que um requisito está em

um determinado ciclo ou etapa de desenvolvimento, enquanto o LT é considerado uma medida que avalia o tempo em que um requisito é iniciado em desenvolvimento até ser entregue ao cliente (Budacu & Pocatilu, 2018).

Já as práticas tradicionais utilizam o EVM para medir o desempenho e o progresso de projetos, utilizando métricas baseadas em medidas comparativas entre o esforço planejado em relação ao esforço realizado (PMI, 2021). Glenwright (2007) apresenta as métricas mais utilizadas pelas organizações sob as práticas tradicionais de gerenciamento de projetos são (i) Schedule Performance Index (SPI) e (ii) Cost Performance Index (CPI). O SPI analisa o progresso real ou o valor agregado do trabalho realizado com o progresso do trabalho planejado até o momento da medição. Já o CPI é uma medida da eficiência de custos realizados com o progresso do custo planejado até o momento da medição. Assim, as métricas possuem relações distintas de acompanhamento durante o ciclo de vida dos projetos.

Em relação às entregas do projeto, nas práticas tradicionais, o produto gerado é entregue somente no final do projeto ou entregue em fases dentro de uma estratégia de cronograma em fases (PMI, 2021). Já nas práticas ágeis o produto é entregue em até quatro semanas (Sprint) na forma de incrementos funcionais. A soma dos incrementos funcionais pode viabilizar os lançamentos do Minimum Viable Product (MVP) (Ries, 2011). O MVP é caracterizado por ser uma versão funcional e sem erros críticos de um software para possível liberação aos clientes pra receber o feedback mais rápido possível (Sutherland, 2019).

Ao se observar a utilização das métricas para análise do desempenho de projetos de desenvolvimento de software, é possível notar divergências entre as práticas ágeis e tradicionais de gestão de projetos. Essa divergência consiste em um cenário em que organizações que conduzem projetos sob as práticas tradicionais decidem implementar as práticas ágeis de gestão de projetos. Porém, a liderança permanece utilizando métricas tradicionais baseadas nas abordagens de custo, prazo e escopo diferentes entre as práticas, podendo causar impactos no desempenho dos projetos ágeis.

O problema que tange e justifica nessa pesquisa é que muitas organizações que conduzem projetos, implementaram o modelo ágil de desenvolvimento de software em equipes isoladas e com as métricas ágeis comumente na operação (Jyothi & Rao 2017; dos Santos *et al.* 2018). Além disso, a liderança realiza a medição do desempenho do projeto por meio de métricas tradicionais, o que podem causar impactos no gerenciamento de projetos

ágeis, dado que a gestão tradicional e a ágil de projetos operam com abordagens de custo, prazo e escopo de maneiras diferentes (Choudhury, 2019; Dias & Larieira, 2021).

Com base no apresentado sobre a divergência da utilização de métricas para o controle de projetos tradicionais e ágeis, abre-se uma lacuna para a realização desta pesquisa, procurando responder a seguinte questão: “Quais são as métricas utilizadas para medir o desempenho de projetos nas práticas ágeis?”. O objetivo deste artigo foi de mapear quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis. Para responder tal indagação, foi adotado como estratégia metodológica uma Revisão Sistemática de Literatura (RSL) verificando as métricas tradicionais e ágeis utilizadas no desempenho de projetos. A justificativa para tal escolha se dá pelo papel da RSL que auxilia no mapeamento e avaliação de uma estrutura intelectual específica para desenvolver um corpo de conhecimento (Tranfield, Denyer & Smart, 2003).

Na seção seguinte serão apresentados os procedimentos metodológicos. Posteriormente são apresentados os resultados e, por fim, serão apresentadas as conclusões, discutidas as limitações desta pesquisa e indicações para estudos futuros.

2 MATERIAIS E MÉTODOS

Esta pesquisa adotou uma RSL como método a fim de compreender a convergência de duas temáticas relevantes no contexto de gestão de projetos, as métricas, tanto nas práticas tradicionais como nas práticas ágeis. A RSL se difere das tradicionais revisões narrativas por adotar um processo científico sistemático que é replicável e transparente (Tranfield, Denyer & Smart, 2003). Nesse sentido, justifica-se também a sua utilização por uma RSL minimizar o viés na construção de um *corpus* teórico, bem como a possibilidade de construir uma trilha de auditoria das decisões e procedimentos aplicados (Cook, Mulrow & Haynes, 1997).

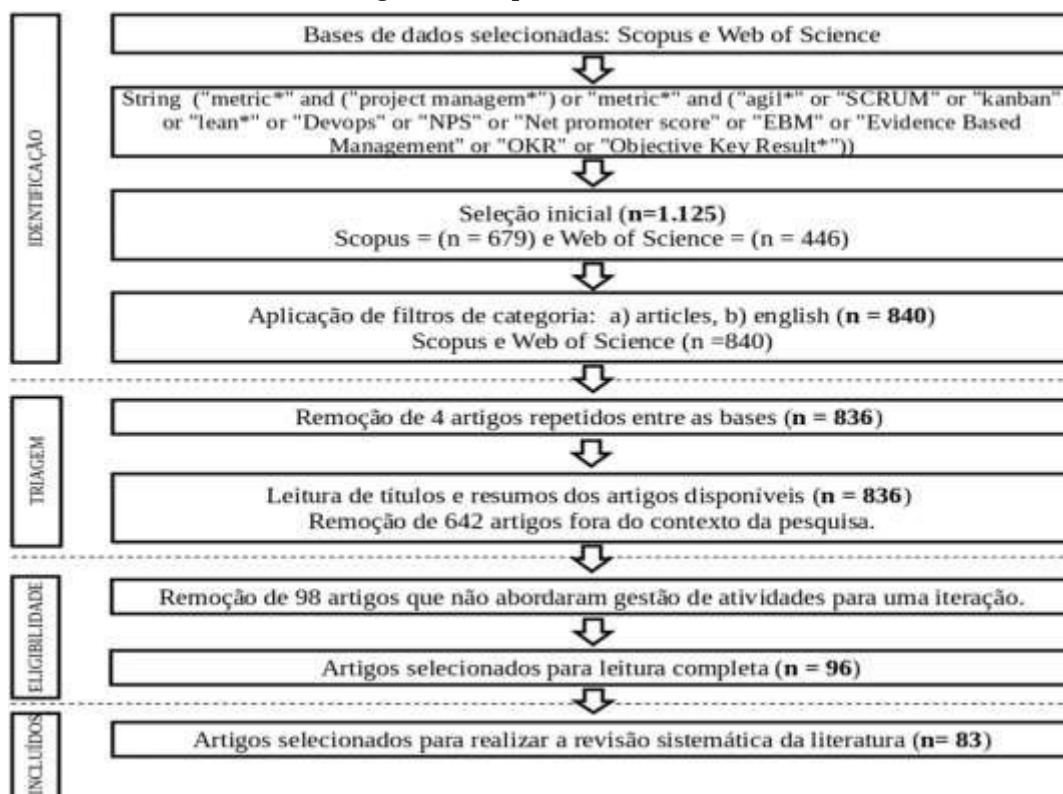
Os procedimentos para realização desta RSL seguiram seis fases prescritas por Pollock e Berge (2018): (i) esclarecer metas e objetivos de pesquisa; (ii) buscar pesquisas relevantes; (iii) coletar dados; (iv) avaliar a qualidade dos estudos; (v) sintetizar as evidências; (vi) interpretar os achados.

A primeira etapa foi motivada pela questão que norteia esta pesquisa, que foi “quais são as métricas utilizadas para medir o desempenho de projetos tradicionais e ágeis?” Assim, foram utilizadas como fontes de pesquisa as bases de dados acadêmicas *Web of Science* e

Scopus, por serem umas das principais bases de dados para acesso as pesquisas publicadas na área de ciências sociais.

A *string* utilizada para realização das buscas foi ("metric*" and ("project managem*") or "metric*" and ("agil*" or "SCRUM" or "kanban" or "lean*" or "Devops" or "NPS" or "Net promoter score" or "EBM" or "Evidence Based Management" or "OKR" or "Objective Key Result*")). A pesquisa foi realizada em 01 de maio de 2022. A utilização dos operadores booleanos “and” e “or”, além do uso do símbolo “*” permitem uma maior abrangência e controle na construção da base de pesquisa. Os operadores são aplicados levando em consideração a intersecção das duas áreas de pesquisa estudadas. O uso do asterisco incorpora todas as variações da palavra na posição posterior a que ela se encontra. Destaca-se que não foi aplicado filtro temporal para que fosse possível mapear toda a produção sobre as temáticas estudadas. Portanto, após a primeira fase que compreende o estabelecimento dos objetivos, os pesquisadores passaram para a aplicação da *string* de busca. Os resultados encontrados na primeira rodada passaram pela análise e triagem, apresentado na Figura 1, respeitando o proposto por Pollock e Berge (2018), que apresentam quatro etapas para a elaboração do *corpus* de análise. A Figura 1 apresenta a primeira etapa destaca a *string* utilizada para a pesquisa nas bases de dados. A segunda etapa é a triagem dos resultados com a finalidade de balizar os artigos correspondentes à proposta de pesquisa. Já na terceira etapa foram aplicados os critérios de elegibilidade, onde os artigos foram avaliados segundo os critérios de inclusão e exclusão. A quarta etapa constituiu o *corpus* de análise, resultando na base de dados amostral de artigos para a pesquisa.

Figura 1 - Etapas da RSL



Fonte: Adaptado de Pollock e Berge (2018).

Para a composição da base de dados, filtros foram aplicados durante a pesquisa inicial, como o critério de somente artigos em periódicos, excluindo da base os artigos em congresso, livros, entre outros. Essa exclusão foi definida, devido esses estudos não passarem por uma revisão por pares em alguns casos, além de mitigar a redundância de obras apresentadas em congressos e publicadas em periódicos. As áreas delimitadas para esta pesquisa foram: “*Management*” e “*Business*” para *Web of Science*, “*Business*” para a *Scopus*. Com a consolidação das bases de dados, foram removidos os artigos repetidos com o objetivo de eliminar a redundância. Na sequência, a base de dados foi tratada com auxílio de planilhas eletrônicas do *LibreOffice*. O *LibreOffice* permitiu realizar as análises dos dados e apresentar os resultados a partir da combinação de informações quantitativas pela análise de frequência, bem como informações qualitativas pela categorização dos conteúdos dos artigos. Esta fase da pesquisa permitiu também apresentar uma análise descritiva relevante. A partir da leitura dos títulos e resumos na fase de triagem para verificar a elegibilidade dos artigos foram estabelecidos alguns critérios de inclusão e exclusão, destacados na Tabela 1. Após verificação e leitura dos resumos e introduções dos 840 artigos selecionados previamente, a base final foi composta por 83 artigos aceitos que formaram o *corpus* de pesquisa.

Tabela 1 - Critérios de Inclusão e Exclusão

Critérios de Inclusão	Razão para Inclusão
Artigos que conceituam os construtos estudados	Permitir abordar finalidades da pesquisa: compreender os construtos estudados conforme as obras.
Artigos que abordam as relações entre os construtos estudados	Permitir contemplar alinhamento das obras: compreender as interdependências e relações entre os construtos.
Artigos publicados	Oferecer maior rigor nos argumentos e contribuições teóricas estudadas.
Critérios de Exclusão	Razão para Exclusão
Artigos com foco em Qualidade, <i>Marketing</i> , Saúde, Finanças ou outros fins fora dos construtos determinados.	Excluir artigos que não estejam focados nas questões que oferecerão <i>insights</i> para atender os objetivos da pesquisa.
Artigos sem fundamentação teórica relevante ou de baixa relação com os construtos.	Uma das finalidades do estudo é obter perspectivas futuras de pesquisa, por meio de conhecimento teórico existente em uma estrutura, para a qual os pressupostos teóricos são pré-requisitos.

Fonte: Elaborado pelos autores, 2021.

O próximo passo foi de realizar a leitura dos 83 artigos contidos no *corpus* de análise, categorizando os conteúdos em planilhas do *LibreOffice* com o objetivo de apresentar um agrupamento dos achados e comparação das categorias. As atividades aplicadas nessa fase estão alinhadas com as prescrições de Pollock e Berge (2018) nas fases (v) sintetizar as evidências e (vi) interpretar os achados. Embora tenham sido aplicados alguns tratamentos quantitativos. Nessa pesquisa, priorizou-se a análise qualitativa dos artigos a fim de constituir uma matriz que pudesse representar os achados deste estudo.

3 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Nessa seção são apresentados os resultados após a realização dos processos de análise dos artigos e as categorias evidenciadas após a análise aprofundada dos conteúdos dos artigos.

3.1 Evolução da publicação dos artigos

Os artigos selecionados nas bases de dados *Web of Science e Scopus* foram triados e o *corpus* de análise foi constituído por 83 artigos. Os artigos passaram por uma análise mais aprofundada, o que permitiu compreender os estudos sobre a relação entre as métricas ágeis e

as métricas tradicionais. Os 83 artigos analisados estão situados temporalmente entre os anos de 2011 até maio de 2021. De acordo com esta base de dados, a publicação de artigos que tratam da relação entre as métricas ágeis e métricas tradicionais ficou entre 4 e 5 artigos de 2011 até 2013. Entre os anos de 2014 até 2017 que as publicações subiram para 8 e 9 artigos publicados. Somente em 2018 que 17 artigos foram publicados e retornando para 8 artigos em 2019, 13 em 2020 e 10 em 2021. Um aspecto relevante é que no ano de 2018 teve o número maior de publicações e correspondeu com 18% de publicações nos últimos dez anos ou de 2011 a 2021. Após a análise da distribuição temporal dos artigos, também se buscou compreender a frequência de artigos por periódico. A Tabela 2 demonstra os primeiros dez periódicos e a respectiva quantidade de artigos publicados, ordenados pelo maior número de publicações e pela ordem alfabética por periódicos.

Tabela 2 - Periódicos e número de publicações no tema

#	Periódicos	Publicações
1	IEEE Access	15
2	Journal Of Systems And Software	14
3	International Journal Of Project Management	10
4	International Journal Of Lean Six Sigma	10
5	Computer Networks	7
6	Computers & Industrial Engineering	7
7	Journal Of Modelling In Management	6
8	Journal Of Software-Evolution And Process	6
9	Journal Of Manufacturing Technology Management	6
10	International Journal Of Software Engineering And Knowledge Engineering	6

Fonte: dados da pesquisa, 2022.

Com relação ao número de artigos publicados por periódico, os periódicos considerados mais profícuos, de acordo com dados da pesquisa, são: 'IEEE Access' com 15 artigos; 'Journal Of Systems And Software' com 14 artigos e 'International Journal Of Project Management' com 10 artigos. Nota-se que a grande maioria dos periódicos configura com apenas 1 e 2 artigos publicados, sumarizando um total de 100 periódicos. Apesar do tema de gestão de projetos estar associado a Administração, as três revistas com mais publicações são associadas a engenharia e tecnologia (*IEEE Access* e *Journal Of Systems And Software*) e

uma de gestão de projetos (*International Journal Of Project Management*). Na próxima seção serão discutidos artigos mapeados no *corpus* dessa pesquisa, a categorização que os artigos foram submetidos e a discussão dos achados dos artigos que compõem as categorias.

3.2 Análise e discussão das categorias encontradas

Após o mapeamento dos artigos que constituíram o *corpus* de análise, foi realizada uma análise aprofundada dos conteúdos publicados. A leitura e categorização dos conteúdos levou a classificação dos artigos em quatro categorias, apresentados na Tabela 3.

Tabela 3 - Classificação dos artigos em 4 categorias

Categoria	Qtde	Autores
Métricas de Projetos Tradicionais	40	Vanhoucke (2011); Chen (2021); Aljedaibi e Khamis (2019); Colin <i>et al.</i> (2015); Rajagopalan e Srivastava (2018); Chen, Chen e Lin (2016); Fleche <i>et al.</i> (2017); Sharma, Caldas e Mulva (2021); Batselier e Vanhoucke (2015); Chen (2018); Pajares e Lopez-Paredes (2011); Vanhoucke (2019); Colin e Vanhoucke (2014); Song, Marten e Vanhoucke (2021); Kerkhove e Vanhoucke (2017); Geng <i>et al.</i> (2018); Sato e Hirao (2013); Staron e Meding (2016); Spijkman <i>et al.</i> (2021); Ibrahim, Hanna e Kievet (2020); Chen <i>et al.</i> (2020); Ko e Cho (2020); Yousefi <i>et al.</i> (2019); Orgut <i>et al.</i> (2018); Kristiansen e Ritala (2018); Rajablu <i>et al.</i> (2017); Han, Choi e O'Connor (2017); Hanna (2016); Kim, Kim e Cho (2016); Hazır, (2015); Wauters e Vanhoucke (2015); Grau e Back (2015); Chen, H. L. (2014); Narbaev e De Marco (2014); Jethani (2013); El Asmar, Hanna e Loh (2013); Hanna, Tadt e Whited (2012); Iqbal, Naeem e Khan (2012); Pozzana <i>et al.</i> (2021); Colin e Vanhoucke (2015).
Métricas Ágeis de Fluxo	9	Grimaldi <i>et al.</i> (2016); Dennehy e Conboy (2018); Mas, Mesquida e Pacheco (2020); Ahmed <i>et al.</i> (2017); Sajedi-Badashian e Stroulia (2020); Petersen e Wohlin (2011); Choraś <i>et al.</i> (2020); Meidan <i>et al.</i> (2018); Sadler (2020).
Métricas Ágeis de Backlog do Produto	9	Azzeh e Nassif (2016); Wallace e Sheetz (2014); Parrend <i>et al.</i> (2014); Kayes, Sarker e Chakareski (2016); Zheng <i>et al.</i> (2021); Pavlova <i>et al.</i> (2021); Antinyan (2014); Prakash e Viswanathan (2018); Asha e Mani (2018).
Métricas Ágeis de Produto	25	Nidagundi e Novickis (2016); Mäkiahho, Vartiainen, e Poranen (2022); Chang (2015); Mascarenhas Hornos <i>et al.</i> (2014); Singh, Singh e Singh (2019); Simpson <i>et al.</i> (2021); Day <i>et al.</i> (2019); Üsfekes, <i>et al.</i> (2019); Abdelrahman Aljemabi, Wang e Saleh (2020); Costa, Vasconcelos, e Fragoso (2020); Ibba <i>et al.</i> (2018); Concas <i>et al.</i> (2012); Shawky e Abd-El- Hafiz (2016); Tahir, Rasool e Noman (2018); Isong e Obeten (2013); Tanveer <i>et al.</i> (2019); Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018); Yamashita e Counsell (2013); Bakhtiary, Gandomani e Salajegheh (2020); Hernández <i>et al.</i> (2020); Athanasiou <i>et al.</i> (2014); Gao <i>et al.</i> (2011); Perkusich <i>et al.</i> (2017); Savola, Frühwirth e Pietikäinen (2012).
Total	83	

Fonte: dados da pesquisa, 2022.

As quatro categorias encontradas compreendem uma abstração observada nas leituras dos artigos. Cabe explicitar que o processo de análise inicial dos artigos foi individual e,

depois, para seleção das categorias passou-se pela análise dos pesquisadores desta pesquisa. O processo de categorização foi realizado pela discussão e estabelecimento de um consenso sobre qual era a maior aderência de todos os artigos em cada uma das categorias constituídas.

Na próxima seção, serão discutidas as categorias para explicitar os achados desta pesquisa e as categorias aqui apresentadas representam uma forma de explicação e organização dos conteúdos estudados.

4 MÉTRICAS DE PROJETOS TRADICIONAIS

Como visto, as práticas de gerenciamento de projetos tradicionais se baseiam, para medir o desempenho dos projetos, a partir de um conjunto de métricas orientados frente a um planejamento prévio. Desse modo, as métricas são caracterizadas, em sua essência, utilizando as medidas de escopo, tempo e custos, mediante a um planejamento previamente estabelecido. Tais medidas são comparadas com a posição atual do projeto (trabalho realizado) em relação ao planejado até o momento da medição (trabalho realizado).

Frente ao uso das práticas tradicionais de gestão de projetos, Vanhoucke (2011) apresentou como métrica, no Monte-Carlo para simular o progresso do projeto sob a estrutura analítica do projeto (EAP), utilizando dados do gerenciamento de valor agregado (EVM) como possíveis sinais de alertas e acionadores para a necessidade de ações corretivas. Ainda sob o olhar da EAP, Colin e Vanhoucke (2015) apresentaram um modelo multivariado adotando como métrica o EVM/ES, resultando no *Earned Value Management/Earned Schedule*. A métrica é utilizada pelo gerente caso o projeto apresente algum risco de maneira preventiva. Kerkhove e Vanhoucke (2017) adotaram a técnica *Earned Incentive Metric*, empregando o EVM/ES com incentivos de custo e/ou tempo. Como resultado, os autores destacaram que *Earned Incentive Metric* apresentou em papel mais significativo que o EVM/ES em uma amostra de 4.200 projetos com tamanhos e estruturas de contrato diferentes ou parametrizando o tamanho de um projeto grande (Chen *et al.*, 2020).

Sobre a adoção do EVM como métrica, Chen, Chen e Lin (2016) concluíram que a melhoria no poder preditivo do *Planned Value* (PV), calculado pelo erro percentual médio absoluto (*Mean Absolute Percentage Error*), antes da execução do projeto fornece aos gestores informações preditivas confiáveis no desempenho do *Earned Value* (EV) e *Actual Cost Value* (ACV), permitindo uma ação proativa efetiva para garantir resultados de desempenho favoráveis. Chen (2021) e Hazır (2015) adotaram como métrica variação de

desempenho na fase de planejamento que explica uma parte substancial da dificuldade do projeto na conclusão, mediante a um modelo de classificação de problemas de projetos, com o objetivo de identificar problemas nos projetos que provavelmente falharão antes de eles começarem. No mesmo sentido, Sharma, Caldas e Mulva (2021) utilizaram o EVM nas áreas de custo, cronograma, segurança, retrabalho e mudanças no projeto com o propósito de apoiar à tomada de decisão, facilitar o orçamento de capital, o impacto das condições de mercado e na melhoria contínua e melhores práticas de gestão. Nessa mesma linha de métrica, Yousefi *et al.* (2019) apresentam o controle estatístico em índices do *Earned Duration Management* (EDM) para melhor investigar as variações do desempenho do cronograma do projeto ou para controlar projetos de inovação radical. Ainda em relação ao cronograma, Han *et al.* (2017) destacam o uso de métricas de qualidade de cronogramas baseados no limite de duração máxima (30 dias); Proporção de tarefas detalhadas para marcos; Porcentagem de tarefas no caminho crítico; Número de defasagens; Tipo de relacionamento (terminar para começar); e Marcos sem predecessor ou sucessor. Wauters e Vanhoucke (2015) criaram um experimento computacional para avaliar a estabilidade usando a nova métrica de defasagem média proposta usando o *Coefficient of Variation* (CV) do EVM. Grau e Back (2015), no mesmo contexto, destacam que as métricas de controle do cronograma devem apoiar o gestor a ser proativo, em vez de reativo, no reconhecimento de tendências e eventos melhora significativamente a capacidade da equipe de minimizar a diferença entre os resultados previstos e reais. Chen (2014) combinou séries temporais e análise de regressão para desenvolver PV em um modelo de resposta EV para prever *Work Perform* (WP) de EV. Usando a relação entre *Budgeted Cost* (BC) e o WP estimado representado no modelo de resposta EV, Chen (2014) desenvolveu ainda EV em um modelo ACV para prever *Actual Cost* (AC) de ACV.

Em relação a duração do projeto, Batselier e Vanhoucke (2015) adotaram duas técnicas baseadas no EVM que integram, respectivamente, retrabalho e sensibilidade à atividade no tempo de EVM prevendo as extensões do projeto. Lopez-Paredes (2011) introduziram duas novas métricas denominadas por Índice de Controle de Custos (ICC1) e Índice de Controle de Cronograma (ICC2). Ambos os índices compararam as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o risco, custo e cronograma, o que pode ser aplicado nas fases do ciclo de vida do projeto (Chen, 2018).

Sato e Hirao (2013) contribuíram para estabelecer o conceito de orçamento ótimo e esclarecer essa condição e a teoria permite a análise do equilíbrio custo/risco não apenas no nível do projeto, mas também no nível da rede de atividades. Kim, Kim e Cho (2016) estudaram um processo com relação linear entre atividades em que as saídas de um negócio precedente se tornam diretamente os insumos de um sucessor. Narbaev e De Marco (2014) calcularam o *Cost Estimate At Completion* (CEAC) que é um número considerado como novidade de pesquisa e que contribui para a extensão do corpo de conhecimento EVM. Narbaev e De Marco (2014) apresentaram uma nova abordagem metodológica que foi desenvolvida para fornecer previsões CEAC mais precisas, confiáveis e eficazes para os estágios iniciais e intermediários da execução do projeto de construção.

Em relação ao resultado dos projetos, Aljedaibi e Khamis (2019) desenvolveram o *Critical Success Factors* (CSF) é um método para medir, monitorar e controlar fatores críticos de sucesso de sistemas de *software* grande, por meio de escala de metas, custos, infraestrutura de TI, gestão de projetos, entre outros. Rajagopalan e Srivastava (2018) adotaram um método único usando uma variável chamada de *Project Health Index*, capaz de medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base no *Corrective And Preventive Actions*. Ibrahim, Hanna e Kievet (2020) examinaram as diferenças de desempenho tempo de processamento de pedido de mudança, aumento no prazo do cronograma, aumento de custo, mudança percentual do projeto, qualidade geral dos sistemas do projeto, mudança percentual relacionada à qualidade/valor, mudança percentual relacionada ao *design*.

Durante as fases de iniciação e planejamento do projeto, métricas da abordagem tradicional são incorporadas para prever o uso de recursos humanos e outras necessidades de execução de atividades para apoio ao controle e o monitoramento de projetos (PMI, 2017). Assim, Song, Martens e Vanhoucke (2021) apresentaram a métrica de *Using Schedule Risk Analysis* (SRA) com a restrição de recursos (*Resource Constrained* - RC) para melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos. Pozzana *et al.* (2021) apresentaram um conjunto de dados composto por projetos de engenharia descritos por suas redes de atividades, registros do desempenho e se a atividade foi entregue no prazo.

Em se tratando do envolvimento das métricas com os envolvidos nos projetos, Rajablu *et al.* (2017) desenvolveram o *Stakeholder Metrics-integrated Management Model* que integra as partes interessadas que permite gerenciá-las identificando, comunicando, engajando, capacitando e atendendo as partes interessadas com métricas e ferramentas. Além

disso, Hanna, Tadt e Whited (2012) utilizaram análise de *bootstrap* para avaliar o desempenho de um projeto de infraestrutura. Hanna (2016) avaliaram com métrica o desempenho geral mais alto no que se refere à implementação de técnicas de construção enxuta, equipe de liderança de projeto engajada e capacitada e um alto envolvimento de *stakeholders* em todo o projeto.

Algumas métricas possuem foco no desenvolvimento do produto. Fleche *et al.* (2017) implantaram métricas quantitativas e não intrusivas para os *designers*, onde cada métrica utiliza variáveis do *design* do produto planejado. Spijkman *et al.* (2021) propuseram o modelo *Requirements Engineering for Software Architecture* (RE4SA) que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. Além disso, Spijkman *et al.* (2021) forneceram métricas para quantificar o alinhamento entre requisitos e arquitetura para detectar sinais de granularidade, concentrando-se em valores discrepantes em um conjunto de dados. Jethani (2013) apresentou o esforço no nível da tarefa e monitoramento do progresso; Monitoramento de marcos e cronogramas; e Monitoramento de defeitos, com base nas atividades regulares de planejamento e monitoramento de projetos de desenvolvimento de *software*. A visão é estendida por Iqbal, Naeem e Khan (2012), que completaram que a rastreabilidade avalia os vínculos entre os requisitos dentro de um documento e métricas de integridade de requisitos, que verificam se os requisitos especificados estão completos ou não para o desenvolvimento do produto. Em ambiente de escritório de projetos, Geng *et al.* (2018) integraram o aprendizado organizacional e o desenvolvimento do conhecimento com um processo de seleção de projetos, propondo uma estrutura para avaliar essa integração baseada no *Analytic Hierarchy Process*, ou por meio de uma avaliação de eficiência a ser realizada por meio da atualização da lista de análise de projetos (Ko & Cho, 2020). Por fim, a Tabela 4 trás uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Tradicionais para auxiliar no entendimento de cada métrica e a sua contribuição à gestão tradicional.

Tabela 4 – Métricas de Projetos Tradicionais, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição de uso	Métrica Relacionada
<i>Earned Value Management</i> (EVM)	Medir o progresso e projetos combinando escopo, tempo e custos.	O escopo é fixo e mantém o tempo e custos estimados.	-
<i>Earned Value Management/ Earned Schedule</i> (EVM/ES)	Medir possíveis riscos de maneira preventiva sobre o cronograma do projeto.	Restringe-se aos riscos relativos ao prazo.	EVM

<i>Earned Duration Management (EDM)</i>	Medir as variações do desempenho do cronograma do projeto ou para controlar projetos de inovação radical.	Restringe-se a medir as variações (não as causas) no desempenho do projeto.	EVM, EVM/ES
<i>Planned Value (PV)</i>	Medir o custo orçado para o trabalho programado para ser feito.	Mantém o foco no custo orçado para apenas ser utilizado como <i>baseline</i> para avaliar o desempenho do custo realizado.	EVM
<i>Actual Cost (AC)</i>	Medir o custo realizado pelo trabalho realizado.	Mantém o foco no custo realizado e não avalia os motivos pelo desempenho.	EVM
<i>Earned Value (EV)</i>	Medir o valor agregado com base no escopo, prazo e custo.	Se restringe a avaliar o desempenho do projeto com base no escopo, prazo e custo.	EVM
<i>Coefficient of Variation (CV)</i>	Experimento computacional para avaliar a estabilidade usando a nova métrica de defasagem.	Restringe-se a medir as variações (não as causas) no desempenho do projeto.	EVM
<i>Work Perform (WP)</i>	Medir a combinação de séries temporais e análise de regressão para desenvolver PV em um modelo de resposta EV para prever WP.	Restringe-se em avaliar o desempenho do cronograma.	PV/EV
<i>Budgeted Cost (BC)</i>	Medir o custo orçado estimado.	Restringe-se em desempenho de custos.	EV/WP
<i>Actual Cost Value (ACV)</i>	Medir a relação entre BC e o WP estimado representado no modelo de resposta EV, para prever o AC de ACV	Restringe-se em desempenho de custos.	AC/EV
<i>Índice de Controle de Custos (ICC1)</i>	Compara as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o cronograma do projeto.	Com base em medições do EVM, restringe-se e avaliar o desempenho de custos.	EVM
<i>Índice de Controle de Cronograma (ICC2)</i>	Compara as medidas de EVM com os valores máximos que o projeto deveria apresentar se o projeto estivesse rodando sob uma hipótese de análise de risco que afetam o custo do projeto.	Com base em medições do EVM, restringe-se e avaliar o desempenho do cronograma.	EVM
<i>Cost Estimate at Completion (CEAC)</i>	Medir a estimativa de custo na conclusão que é um número que contribui para a extensão do corpo de conhecimento EVM.	Com base em medições do EVM, restringe-se e avaliar o desempenho de custos.	EVM
<i>Critical Success Factors (CSF)</i>	Medir, monitorar e controlar fatores críticos de sucesso por meio de escala de metas, custos, infraestrutura de TI e gestão de projetos.	Restringe-se em avaliar uma série de variáveis para determinar o sucesso do projeto.	-
<i>Project Health Index (PHI)</i>	Medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base em ações de prevenções e correções.	Está restrito em medir a saúde como progresso do projeto com base em ações de prevenções e correções.	-

<i>Using Schedule Risk Analysis (SRA)</i>	Utiliza a restrição de recursos para medir e melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos.	Restrito em medir o desempenho do projeto sob a ótica de recursos restritos.	-
<i>Stakeholder Metrics- integrated Management Model (SMMM)</i>	Integra as partes interessadas com métricas de identificação, comunicação, engajamento, capacitação e atendimento.	Restringe-se em medir o nível de integração dos <i>stakeholders</i> com o projeto.	-
<i>Requirements Engineering for Software Architecture (RE4SA)</i>	Medir a conexão entre os artefatos do projeto e que facilita a comunicação dentro da equipe de desenvolvimento de <i>software</i> .	Restringe-se em medir o relacionamento entre artefatos do projeto.	-

Fonte: Elaborado pelo autor

Os artigos dessa categoria apresentaram discussões importantes sobre as principais métricas utilizadas pelas organizações para o controle e monitoramento de projetos tradicionais. A partir dessa perspectiva, as métricas como EVM, EV, AC, PV e ES são as mais citadas na literatura e nota-se a introdução de novas métricas como EDM e métricas com *benchmarks* associados para o processo de requisição de informações em projetos.

5 MÉTRICAS ÁGEIS DE FLUXO

As métricas ágeis de fluxo visam medir o desempenho de projetos ágeis durante o fluxo de desenvolvimento de software ou de produto (Jyothi e Rao, 2017; dos Santos *et al.*, 2018). Assim, as métricas ágeis de fluxo procuram medir desde o ponto de comprometimento (Sprint Backlog) até a entrega da funcionalidade para os clientes/stakeholder. As métricas comumente utilizadas nesse contexto são o Lead Time, WIP, Cycle Time, Work Item Age entre outras que procuram medir as etapas de desenvolvimento de software.

Inserido no Scrum, Parrend *et al.* (2014) avaliaram o impacto das práticas de gerenciamento de projetos nos dois primeiros Sprints, medindo as práticas de elegibilidade do incremento com impacto mensurável no desempenho das equipes e nível de exigência que levam ao fracasso se não forem implementadas. Parrend *et al.* (2014) destacaram a importância dos princípios do Scrum e as propriedades *swarming* são práticas relevantes de sucesso e um sintoma de gerenciamento de desempenho. O *swarming* é uma técnica e uma dinâmica de gestão que consiste em empregar a inteligência coletiva da equipe na execução de uma atividade crítica para resultados mais eficazes do projeto (Parrend *et al.*, 2014).

Dennehy e Conboy (2018) realizaram a avaliação por meio de um estudo empírico do uso de fluxo de trabalho com Kanban, obtendo dois resultados. Primeiro, um conjunto de

desafios empíricos, tais como a mudança organizacional e a configuração de quadros Kanban. Segundo, identificaram que as métricas de Lead Time e Velocity são as mais utilizadas para o planejamento e nos processos de melhoria contínua no projeto (Dennehy & Conboy, 2018).

Os autores Mas, Mesquida e Pacheco (2020) realizaram um estudo para saber se empresa é ágil e madura, e se um conjunto de métricas de desempenho estiver em conformidade com as melhores práticas de gerenciamento de projetos. Um conjunto de métricas foram identificadas nas empresas ágeis estudadas, tais como efficiency flow, lead time, cycle time, reopens e tasks finished flow (Mesquida & Pacheco, 2020). A métrica efficiency flow mede quantos itens do backlog são entregues por ciclo (Mesquida & Pacheco, 2020). O lead time mede o tempo em que um item do backlog é iniciado até o momento que é entregue ao cliente (Mesquida & Pacheco, 2020). A métrica reopens mede o número de tarefas ou incidentes reabertos e a métrica tasks finished flow mede as tarefas finalizadas no fluxo de desenvolvimento, diferentemente do Throughput, que mede funcionalidades entregues ou histórias de usuário. (Mesquida & Pacheco, 2020).

Grimaldi *et al.* (2016) apresentaram como uma estrutura ágil escalável, baseada em SAFe 3.0 e mostrou como implementar e medir o desempenho considerando que as funções e responsabilidades são divididas entre a organização e o fornecedor. Grimaldi *et al.* (2016) utilizaram métricas como eficácia e eficiência, custo-hora real e planejada, impedimentos e custo do atraso durante as rodadas de Sprint. No mesmo sentido que Ahmed *et al.* (2017) discutiram sobre as métricas Sprint Burn Down, Epic Burn Down Velocity Chart, Control Chart e a maneira como muitos dos cálculos de métricas são fortemente acoplados aos pontos de história para itens como o Épico ou Sprint.

Sajedi-Badashian e Stroulia (2020) mostraram que é possível melhorar e medir a eficácia do processo de atribuição de bugs por meio na análise de dados do projeto. Desse modo, Sajedi-Badashian e Stroulia (2020) encontraram uma maneira de atribuir a pessoa mais indicada para a resolução do problema e reduzir o time to market do produto.

Do ponto de vista de Petersen e Wohlin (2011), o Lean fornece ferramentas de análise e melhoria com foco no ciclo de vida do desenvolvimento, enquanto o ágil se concentra em soluções e prescreve conjuntos de práticas para obter agilidade. Petersen e Wohlin (2011) sugeriram análises sobre as métricas como reduzir o lead time para obter alta capacidade de resposta às necessidades dos clientes e fornecer um sistema de rastreamento que mostre o status do desenvolvimento de software. No sentido do monitoramento, as métricas de processo propostas por Choraś *et al.* (2020) foram utilizadas continuamente para ajudar a

garantir a qualidade e estabilidade do software como a estimativa de tarefas e rastreamento de tarefas em tempo real. Meidan *et al.* (2018) identificaram que o desenvolvimento ágil e empresa de pequeno/médio porte foram os contextos de pesquisa mais frequentemente identificados. Meidan *et al.* (2018) demonstraram que a capacidade, desempenho e maturidade foram os atributos de processo mais medidos, enquanto esforço, desempenho e risco foram os atributos de projeto mais medidos.

O modelo de ciclo de vida de desenvolvimento e manutenção de sistemas proposto por Sadler (2020) aplicou efetivamente diferentes níveis de rigor e métricas de qualidade mais rígidas, dependendo do risco associado aos lançamentos. O modelo de Sadler (2020) foi baseado nas normas relacionadas ao gerenciamento de riscos denominado ER2C SDMLC (Enterprise Release Risk-Centric Systems Development and Maintenance Life Cycle). Dentre as métricas avaliadas por Sadler (2020), tivemos a Clareza do Artefato e Envolvimento, Propriedade do Cliente e Aplicabilidade e Rastreabilidade.

Asha e Mani (2018) apresentaram uma metodologia de teste denominada por Teste Simples de Teste de Aceitação aplicando as técnicas de Desenvolvimento Orientado a Testes de Aceitação que podem ser executados como casos de teste de software no contexto do DevOps. Além disso, Asha e Mani (2018) utilizaram os algoritmos Tarjan e Breadth-First Search que renderizam 100% de cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste aplicando as métricas de componentes funcionais, aumentando a frequência de entrega contínua e reduzindo o time to market.

A Tabela 5 a seguir realiza uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de Fluxo para auxiliar no entendimento de cada métrica e a contribuição de cada uma delas à gestão ágil de projetos.

Tabela 5 – Métricas Ágeis de Fluxo, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
Elegibilidade do Incremento	Avalia o impacto das práticas de gerenciamento de projetos nos dois primeiros <i>Sprints</i> .	Com foco na avaliação entre escopo e incremento, não avalia a agregação de valor do incremento.	-
<i>Lead Time</i> (LT)	Mede a entrega a funcionalidade desde o comprometimento até a entrega ao cliente	Com foco em agregação de funcionalidade.	-
<i>Cycle Time</i> (CT)	Mede etapas de desenvolvimento do software dentro da iteração.	Com foco em desempenho de prazo, não avalia o valor da entrega.	-

<i>Velocity</i>	Mede quantos pontos de história foram feitos por <i>Sprint</i> .	Com foco em desempenho de esforço, não avalia o valor da entrega da HU.	-
<i>Efficiency Flow (EF)</i>	Mede quantos itens do <i>backlog</i> são entregues por ciclo.	Não avalia a agregação de valor da entrega da funcionalidade.	LT
<i>Reopens</i>	Mede o número de atividades reabertas.	Somente avalia o retrabalho e não mede as causas para aprendizado.	TFF
<i>Tasks Finished Flow (TFF)</i>	Mede as tarefas finalizadas dentro do fluxo de desenvolvimento.	Com foco em tarefas realizadas, não avalia a agregação de valor da entrega do item do <i>backlog</i> .	<i>Reopens</i>
Eficácia e eficiência	Eficácia e eficiência de entregas do projeto durante a rodada da <i>Sprint</i>	Necessário medir as entregas na perspectiva do cliente e/ou do mercado.	LT, TFF, <i>Velocity</i> , EF
Custo hora real e planejada	Custo hora real e planejada de entregas do projeto durante a <i>Sprint</i> .	Medida que contradiz a expectativa de custos fixos em projetos ágeis.	LT, EF
Impedimentos	Impedimentos mapeados durante a rodada da <i>Sprint</i> .	Somente avalia o retrabalho e não mede as causas para aprendizado.	-
Custo do atraso	Custo de atraso de entregas do projeto durante a rodada da <i>Sprint</i> .	Está concentrado em medir o custo do atraso e não mede as causas para reduzir riscos.	LT, TFF, <i>Velocity</i> , EF
<i>Sprint Burn Down (SBD)</i>	Métrica sobre pontos de história no <i>Sprint</i>	Com foco em desempenho de esforço, não avalia o valor da entrega da HU.	<i>Velocity</i> , LT
<i>Epic Burn Down (EBD)</i>	Métrica sobre pontos de história no Épico	Com foco em desempenho de esforço, não avalia a agregação de valor da entrega do épico.	<i>Velocity</i> , LT
<i>Control Chart (CC)</i>	Mede o <i>Cycle Time</i> (ou <i>Lead Time</i>) na <i>Sprint</i>	Com foco em agregação de funcionalidade.	LT, CT
Atribuição de <i>bugs</i>	Atribui a pessoa mais indicada para a resolução do problema e reduz o <i>time to market</i> do produto.	Apesar de auxiliar na redução do <i>time to market</i> , não mede o valor entregue.	TFF, <i>Reopens</i>
Reduzir <i>Lead Time</i>	Reduz o <i>lead time</i> para obter alta capacidade de resposta às necessidades dos clientes	Com foco em reduzir o prazo da entrega, não avalia a agregação de valor da entrega do item do <i>backlog</i> .	LT
Status Desenvolvimento	Fornece um rastreamento do status do desenvolvimento de <i>software</i> .	O status do desenvolvimento não garante que o escopo e/ou o valor será entregue.	CT, <i>Velocity</i> , SBD, CC

Estimativa de Tarefas (ET)	Estimativa de tarefas para auxiliar na qualidade e estabilidade do software	A estimativa de tarefas tem foco apenas em prazo.	-
Rastreamento de Tarefas (RT)	Rastreamento de tarefas para ajudar na qualidade e estabilidade do software	O rastreamento de tarefas tem foco apenas em status de tarefas.	AR
Capacidade do processo	Medir a capacidade do processo ágil.	Com restrição em medir a capacidade do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Desempenho do processo	Medir o desempenho do processo ágil.	Com restrição em medir o desempenho do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Maturidade do Processo	Medir a maturidade do processo ágil.	Com restrição em medir a maturidade do processo de desenvolvimento, não avalia o produto.	LT, CT, EF, CC, SBD, EBD
Esforço no projeto	Medir o esforço do projeto ágil.	O esforço no projeto tem foco apenas em prazo.	LT, CT, EF, CC, SBD, EBD, ET
Desempenho no projeto	Medir o desempenho do projeto ágil.	O desempenho no projeto tem foco apenas em custo e prazo.	LT, CT, EF, CC, SBD, EBD, ET
Risco no projeto	Medir os riscos do projeto ágil.	Medir riscos não garante sucesso ou entrega de valor pelo projeto.	-
Clareza do Artefato (CA)	Medir a clareza do artefato produzido pelo projeto.	Medir a clareza do artefato pode auxiliar na gestão do <i>backlog</i> , mas não garante entrega de valor.	EPC
Envolvimento e Propriedade do Cliente (EPC)	Medir o envolvimento e propriedade do cliente sobre o artefato produzido pelo projeto.	Medir o envolvimento e propriedade do cliente sobre o artefato auxilia na gestão do <i>backlog</i> . Não garante entrega de valor.	CA
Aplicabilidade e Rastreabilidade (AR)	Medir a aplicabilidade e rastreabilidade do artefato produzido pelo projeto.	Medir a aplicabilidade e rastreabilidade do artefato não garante entrega de valor.	RT
Componentes funcionais	Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste.	Medir a cobertura de cenários e histórias de usuário auxilia na qualidade do <i>software</i> , mas não garante entrega de valor.	-

Fonte: Elaborado pelo autor

Os artigos dos autores apresentaram as métricas ágeis relacionadas ao fluxo operacional de projetos utilizando o *framework Scrum* e o *Kanban*. As métricas mais citadas nos estudos foram as de *Efficiency Flow*, *Lead Time*, *Cycle Time*, *Velocity*, *Sprint Burn Down*, *Epic Burn Down* e *Control Chart*. Verificou-se também que muitos dos cálculos de métricas são fortemente acoplados aos pontos de história como estimativa e monitoramento. Além disso, métricas ligadas os itens de *backlog*, como *reopens*, *tasks finished flow* e *time spent extremes*, e outras métricas baseadas no SAFe 3.0 como eficácia, eficiência, custo-hora real e planejada, resíduos, impedimentos e custo do atraso foram encontradas.

6 MÉTRICAS ÁGEIS DE BACKLOG DO PRODUTO

As métricas ágeis de *Backlog* do Produto visam medir o desempenho de projetos ágeis durante a etapa de definição do escopo do projeto que, em projetos ágeis, são organizados no *backlog* do produto (Mosser, Pulgar & Reinharz, 2022). O *backlog* do produto é composto pelo escopo do projeto dividido em diversos itens ou HUs que são descobertos e detalhados ao longo do desenvolvimento do produto do projeto (Mosser, Pulgar & Reinharz, 2022). Assim, as métricas ágeis de *Backlog do Produto* procuram medir o esforço de desenvolvimento da funcionalidade, nível de priorização e complexidade por *Use Case Points*, e estimativas por pontos de história.

Kayes, Sarker e Chakareski (2016) propuseram uma métrica no *Product Backlog Rating* (PBR) para avaliar o processo de teste no *Scrum* que forneceu uma estrutura adicional para o gerenciamento da qualidade e uma visão abrangente do processo de teste ao longo do ciclo de desenvolvimento do produto. O PBR de Kayes, Sarker e Chakareski (2016) considera a complexidade dos recursos a serem desenvolvidos em uma iteração do *Scrum*, avalia as classificações dos testes e oferece uma pontuação numérica do processo de teste. No mesmo contexto do processo *Scrum*, Zheng *et al.* (2021) mapearam as métricas de processo e do *Service-Oriented Architecture* (SOA) para auxiliar nas decisões de planejamento e *design* durante a *Sprint*. Zheng *et al.* (2021) sugeriram os *Key Performance Indicators* (KPIs) para ajudar os *stakeholders* a entender os imperativos da técnica proposta no planejamento de recursos, avaliação do desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos baseados em *Scrum* e SOA.

Pavlova *et al.* (2021) determinaram os parâmetros do projeto como as características da entrega futura como o custo, prazo e possíveis desvios de prazo e custo. Tais parâmetros

do estudo de Pavlova *et al.* (2021) resultaram em métricas para avaliar o grau de singularidade do projeto e o grau de complexidade do projeto. Para a métrica de complexidade, Pavlova *et al.* (2021) consideraram o número de variantes de soluções tecnológicas, o número de operações para cada variante e o número de alternativas para execução da operação para cada variante da tecnologia do produto do projeto. Wallace e Sheetz (2014) destacaram que a utilidade percebida de uma medida de software pode ser medida por uma avaliação da aplicabilidade da medida ao longo do ciclo de vida, dependência da linguagem de programação, capacidade de prescrever soluções e a validade da medida para definir a complexidade dos requisitos.

Azzeh e Nassif (2016) propuseram um modelo híbrido voltado para medir o esforço e a produtividade para apoiar o processo de estimativa de esforço inicial baseada no tamanho do *Use Case Point* (UCP). As análises de Azzeh e Nassif (2016) contaram com um número significativo de projetos coletados que utilizaram como base o UCP em suas estimativas.

A ontologia desenvolvida por Prakash e Viswanathan (2018) acumulou conhecimento sobre os critérios de avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos. Prakash e Viswanathan (2018) incluíram axiomas que determinam a qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com os requisitos funcionais e não-funcionais e versões de produtos. A Tabela 6 realiza uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de *Backlog* para auxiliar no entendimento de cada métrica e a contribuição de cada uma delas à gestão ágil de projetos.

Tabela 6 – Métricas Ágeis de *Backlog*, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
<i>Product Backlog Rating</i> (PBR)	Mede o nível de prioridade do item do <i>backlog</i>	Medir o nível de priorização dos itens do <i>backlog</i> não garante a priorização no sentido de valor entregue.	-
KPI de Imperativos da Proposta Técnica (IPT)	Auxilia no planejamento de recursos, testar o desempenho da agilidade, valor dos negócios, eficiência da equipe e complexidade de projeto.	Falta de foco na avaliação do KPI com abrangência em diversas fases do projeto num único KPI.	-
Grau de singularidade do projeto	Características da entrega futura como o custo, prazo, possíveis desvios de prazo e custo.	Não faz sentido medir a singularidade de projetos ágeis por custo e prazo (fixos).	IPT, GCP

Grau de Complexidade do Projeto (GCP)	Número de variantes de soluções tecnológicas, número de operações para cada variante e o número de alternativas para execução da operação para cada variante.	Avalia somente questões tecnológicas para verificar a complexidade sem considerar O mercado e escopo para medir a complexidade.	IPT, GCP
Esforço de desenvolvimento de <i>software</i> (EDS)	Avaliação da aplicabilidade ao longo do ciclo de vida, dependência da linguagem de programação específica, capacidade de preservar soluções.	Falta de foco na avaliação da métrica com abrangência em diversas fases do projeto numa única métrica.	IPT, GCP
Esforço e produtividade	Apoiar a estimativa de esforço inicial baseada no tamanho do <i>Use Case Point</i> (UCP)	Usar UCP como métrica de planejado versus realizado não garante a entrega o valor esperado.	IPT, GCP, EDS
Avaliação da qualidade das HUs	Avalia a qualidade das HUs, prioridade e o risco das HUs, relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.	Não realiza a medição das HUs que entregaram o valor esperado.	PBR, CP

Fonte: Elaborado pelo autor

As métricas de *Backlog* de Produto que foram apresentadas nos artigos, trazem métricas importantes para a gestão de requisitos em projetos ágeis. Assim, destacam-se as métricas como o *Product Backlog Rating*, qualidade e valor dos negócios, complexidade de projetos baseados em Scrum e SOA, complexidade do requisito, complexidade da arquitetura de solução e complexidade da mudança. Outros achados tivemos com a estimativa de esforço inicial baseada no tamanho do UCP, avaliação da qualidade das histórias de usuários (HU), sobre artefatos de requisitos e tipos de requisitos e a medição do tamanho do requisito a partir do comportamento criado pelo *Behavior Driven Development* (BDD) usando a notação de Gherkin (Moult & Krijnen, 2020).

7 MÉTRICAS ÁGEIS DE PRODUTO

As métricas ágeis de produto visam medir a qualidade de construção e de ações para produzir um produto de melhor qualidade e para reduzir o seu time to market (Kristensen & Shafiee, 2019; Ries, 2011). Dessa forma, as métricas ágeis de produto nessa sessão medem tanto atividades para melhorar a qualidade do código-fonte, melhoria do processo de testes de *software*, a gestão e rastreabilidade de *bugs* e o processo de desenvolvimento orientado a

testes. Além disso, a categoria buscou apresentar as métricas para medir o nível de satisfação do cliente/*stakeholder* com o produto recebido.

Mäkiahho, Vartiainen e Poranen (2022) apresentaram uma ferramenta de monitoramento de métricas que foi desenvolvida por alunos de graduação e pós-graduação para apoiar os membros do projeto, gerentes de projeto e gestores. Segundo Mäkiahho, Vartiainen e Poranen (2022) a ferramenta permitiu acompanhar a quantidade de *commits*, horas trabalhadas e se os requisitos permaneciam muito tempo em estágios no processo de desenvolvimento.

Os experimentos de Chang (2015) foram usados para detectar itens de risco de software a partir do *cluster* de dados oriundos de projetos. Tais dados foram apresentados por Chang (2015) no formato de métricas com os riscos foram mitigados e fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de *software*. Dentre as métricas apresentadas por Mascarenhas *et al.* (2014), temos o crescimento nas vendas de novos produtos, o orçamento dedicado à análise ou verificação do cliente, projetos orientados para o cliente, projetos e programas respeitando custos e orçamento, marcos respeitados e componentes reutilizáveis.

A pesquisa de Singh *et al.* (2019) propuseram análises de complexidade de programas orientados a objetos em Java com base na métrica *Chidamber-Kemerer* (CK) e o desempenho são medidos com uso da métrica *Mean Time to Execution*. Como resultado indicado por Singh *et al.* (2019), realizar reengenharia usando práticas ágeis é benéfico em termos de implementação de requisitos, estimativa de custo e melhoria de desempenho. Nesse sentido, Isong e Obeten (2013) realizaram uma validação empírica das métricas CK e *Source Lines of Code* (SLOC). A validação empírica de Isong e Obeten (2013) das métricas CK e SLOC e os modelos que as utilizaram em três aspectos: (1) nível de significância e insignificância com propensão a falhas (PF), (2) estado de validação e (3) utilidade na qualidade de software. Segundo Isong e Obeten (2013), as métricas SLOC, *Coupling between Objects* (CBO), *Response for a Class* (RFC), *Weighted Methods per Class* (WMC) possuem forte relação com PF e são consideradas os melhores preditores de PF com impacto na qualidade de *software*.

Nesse sentido que Concas *et al.* (2012) também estudou diversas iniciativas em projetos ágeis que utilizaram as métricas CK, SLOC, CBO, RFC e WMC para medir a qualidade de *software* em diversos projetos. Ainda sobre a qualidade de código-fonte, Yamashita e Counsell (2013) investigaram o potencial de *code smells* (potenciais problemas no código) para refletir indicadores de manutenibilidade em nível de *software*. Ao comparar

os *code smells* com outras abordagens, Yamashita e Counsell (2013) descobriram que as medidas estruturais forneceram mais informações sobre qual sistema tinha o *design* equilibrado, mas essa medida ignorou quando as tarefas de manutenção eram de tamanho pequeno ou médio. O estudo de Athanasiou *et al.* (2014) apresentou um modelo que avalia a qualidade do código de teste combinando métricas de código-fonte que refletem três aspectos principais da qualidade do código de teste: integridade, eficácia e manutenibilidade. Athanasiou *et al.* (2014) encontraram uma correlação positiva e estatisticamente significativa entre a qualidade do código de teste, rendimento e produtividade. As contribuições do estudo de Athanasiou *et al.* (2014) iniciaram a partir do modelo que combina métricas para fornecer uma medida da qualidade do código de teste e a investigação empírica que demonstrou uma correlação positiva significativa entre a qualidade do código de teste e na resolução de *bugs*.

Sobre a experiência do usuário do produto, de Day *et al.* (2019) utilizaram o *Net Promoter Score* (NPS) para avaliar a viabilidade de um sistema de apoio à decisão no Registro Eletrônico de Saúde (RES). O NPS foi avaliado como resultado de esforços em testes de usabilidade e a implementação em larga escala com efeito positivo no paciente e na tomada de decisão compartilhada (Day *et al.*, 2019). Simpson *et al.* (2021) procurou reduzir a carga do RES com um *Sprint* de otimização de duas semanas que trabalhou com 19 provedores de prática avançada em uma unidade de especialidade. Simpson *et al.* (2021) mediram o NPS o tempo gasto no RES com base nos dados de log do usuário, onde o NPS aumentou de 6 para 60 e para o resultado da *Sprint* de duas semanas o NPS foi de 93.

Üsfekes *et al.* (2019) investigaram por meio de uma revisão da literatura a utilidade de mecanismos de incentivo para rastreamento eficiente de *bugs* no *Application Lifecycle Management*. Para Üsfekes *et al.* (2019), isso colabora positivamente para os resultados de métricas que olham para a quantidade de *bugs* resolvidos por unidade de tempo. Abdelrahman Aljemabi *et al.* (2020) mostraram que a *Bug Tracking System-based*, exibem três diferentes níveis de coordenação (Plano, Consciente e Reflexivo) com atividades de colaboração e na métrica de tempo médio para reparo para avaliar o tempo de resolução de *bugs*.

Costa, Vasconcelos e Fragoso (2020) destacaram as etapas de identificar as métricas aplicáveis à arquitetura dos projetos e para quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos. As métricas foram propostas por Costa, Vasconcelos e Fragoso, B. (2020) foram organizadas em métricas de Funcionalidade, Confiabilidade, Eficiência, Manutenibilidade, Portabilidade e Alinhamento.

Ibban *et al.* (2018) entenderam as atividades de engenharia de software relacionadas a Ofertas Iniciais de Moedas (*Initial Coin Offering – ICO*) que é uma forma inovadora de financiar projetos baseados em *blockchain*. Além disso, Ibban *et al.* (2018) buscaram fazer uma comparação entre ICOs e ICOs ágeis. Cerca de 5% das ICOs examinadas por Ibban *et al.* (2018) aplicam práticas ágeis e analisaram as ICOs ágeis em termos de métricas de código-fonte, versões de idioma e uso de ferramentas de teste. Nidagundi e Novickis (2016) descobriram possibilidades de adoção dos modelos e métricas de transformação *Lean* no plano de teste de software para simplificar o processo de teste para uso posterior dessas métricas de teste em Canvas. Com o Canvas, Nidagundi e Novickis (2016) afirmaram que as métricas de teste podem nos dar *feedback* para o processo de teste de *software*. Ainda no tema de qualidade, Shawky e Abd-El-Hafiz (2016) apresentaram uma abordagem que estuda o efeito do processo de desenvolvimento na qualidade *software* e verificaram se existe relação entre o tipo de método de desenvolvimento utilizado e os valores de algumas métricas indicadoras de qualidade. Shawky e Abd-El-Hafiz (2016) afirmaram que as métricas usadas incluem métricas de complexidade, acoplamento e coesão, sendo as duas últimas sendo utilizadas para medir a complexidade. Ainda sobre processo de software, Tahir, Rasool e Noman (2018) identificaram 29 modelos de medição de software e 4 ferramentas projetadas para gerentes de projeto em pequenas e médias empresas. Assim, o estudo de Tahir, Rasool e Noman (2018) apresentou que 51% dos modelos de medição são referentes a melhoria de processo de software, 60% sobre uso de especialistas em medição e experiência, 40% sobre o uso de padrões de medição e 22% sobre o uso de ferramentas automatizadas.

Tanveer *et al.* (2019) desenvolveram um método híbrido que utiliza informações de análise de impacto de mudança na estimativa de esforço e do modelo de estimação baseado em árvores impulsionadas por gradiente (AIG). A avaliação subjetiva de Tanveer *et al.* (2019) sobre o *Hybrid Effort Estimation in Agile Software Development* (HyEEASe) e do *Planning Poker* (PK) foi sobre medir a facilidade de uso percebida, utilidade e aprendizagem. Tanveer *et al.* (2019) avaliaram que o PK obteve classificações superiores ao HyEEASe em facilidade de uso percebida e capacidade de aprendizado, entretanto, os avaliados consideraram que HyEEASe requer menos esforço mental. Em relação à métrica de precisão da estimativa, Tanveer *et al.* (2019) concluíram uma estimativa puramente baseada no modelo AIG não teve um desempenho melhor do que uma estimativa baseada no HyEEASe.

Pradhan e Nanniyur (2021) apresentaram um estudo de caso sobre a transformação em larga escala de um sistema de gestão de qualidade legado para um sistema moderno com

aprendizado de máquina, dados conectados, operações integradas e *Big Data*. Pradhan e Nanniyur (2021) identificaram que as métricas de qualidade precisam abranger a *Sprint* ou Projeto, Lançamentos, Portfólio e Visão Empresarial. Pradhan e Nanniyur (2021) validaram a eficácia da transformação nos perfis de defeitos recebidos, taxas de falhas e cronograma de maturidade de lançamento. Kamulegeya, Mugwanya e Hebig (2018) buscaram entender o uso e os benefícios percebidos da medição em *startups* de *software* na África Oriental e identificaram que as *start-ups* estão usando uma série de métricas orientadas a negócios e produtos e as *startups* mais antigas utilizam as métricas orientadas a projetos e as métricas organizacionais são as métricas menos utilizadas.

Bakhtiary, Gandomani e Salajegheh (2020) mediram a aplicação das abordagens de *Test-Driven Development* (TDD) e Não-TDD (NTDD) para compará-los, dada a qualidade do resultado do trabalho e a produtividade dos desenvolvedores em uma empresa de *software*. Bakhtiary, Gandomani e Salajegheh (2020) mostraram que ao aumentar a escala do projeto, o número de defeitos descobertos antes e após o lançamento aumentou mais na abordagem NTDD em relação ao TDD. Bakhtiary, Gandomani e Salajegheh (2020) observaram que houve uma redução na satisfação do cliente no NTDD em comparação com a abordagem TDD em cada ciclo, e teve desempenho melhor. Outra análise de Bakhtiary, Gandomani e Salajegheh (2020) e Antinyan (2014) trouxeram que medir a complexidade ciclomática que corresponde o nível de complexidade de um programa de computador e resultou que a abordagem TDD reduziu a complexidade no código em comparação com a abordagem NTDD. No mesmo sentido da produtividade, Hernández, *et al.* (2020) estabeleceu o nível de relevância das métricas de produtividade de equipes de desenvolvimento ágil de *software*. Para atingir o objetivo, o nível de relevância das métricas de produtividade da equipe foi identificado por Hernández *et al.* (2020) em três aspectos fundamentais: (i) entrega antecipada e frequente de *software*, (ii) agregação de valor e (iii) aspectos relacionados ao esforço. Hernández *et al.* (2020) sugeriram que é relevante para os profissionais medirem a produtividade entregando *software* com antecedência e frequência com alta relevância o esforço de implantação de *software* que resolva os problemas de clientes. Hernández *et al.* (2020) afirmaram que o cálculo sistemático do retorno do investimento é visto como uma forma relevante de medir a agregação de valor.

Com o enfoque no problema de seleção de atributos no contexto da estimativa de qualidade de *software*, o estudo de Gao *et al.* (2011) investigou comparativa é apresentada para avaliar nossa proposta de abordagem híbrida de seleção de atributos. Dado esse contexto,

Gao *et al.* (2011) destacaram que uma seleção inteligente de métricas de *software* antes de treinar um modelo de previsão de defeitos provavelmente melhorará o resultado, removendo recursos redundantes e menos importantes.

Perkusich *et al.* (2017) sugeriram que o modelo com Redes Bayesianas e que abordasse a avaliação da qualidade de processo e de produto a partir de métricas das métricas de qualidade do incremento, dos eventos do *Scrum* e do código-fonte. Assim, Perkusich *et al.* (2017) identificaram nos nós uma oportunidade para analisar a qualidade do trabalho da equipe do projeto e avaliar as chances de sucesso do projeto.

Savola, Frühwirth e Pietikäinen (2012) demonstraram o potencial prático das métricas de segurança orientadas a riscos, particularmente ao oferecer alguma visibilidade inicial da eficácia e eficiência da segurança. A principal contribuição do estudo de Savola, Frühwirth e Pietikäinen (2012) está na avaliação do potencial e significado de uma metodologia de desenvolvimento de métricas de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de *software*. Assim, a Tabela 7 apresenta uma breve análise de objetivos, restrições e relacionamentos entre as métricas de Projetos Ágeis de Produto para auxiliar no entendimento de cada métrica e a sua contribuição na gestão ágil de projetos.

Tabela 7 – Métricas Ágeis de Produto, objetivos, restrições e relacionamentos

Métrica	Objetivo	Restrição	Métrica Relacionada
Quantidade de <i>Commits</i> (QC)	Quantidade de novos códigos-fonte publicados.	Avaliar código-fonte publicados não é uma avaliação de produtividade.	-
Horas Trabalhadas (HT)	Quantidade de horas trabalhadas durante um período.	Não avalia horas produzidas que é diferente de horas trabalhadas.	TED
Estágios em Dev (TED)	Tempo do requisito nos estágios de desenvolvimento.	Como o <i>Work Item Age</i> do <i>Kanban</i> , restringe-se na medição dos estágios para apontar problemas pontuais.	HT
Características dos riscos de <i>software</i>	Fornecer informações sobre modelos de risco que podem ser usados para entender as características dos riscos de <i>software</i> .	Modelos de risco certamente não avaliam hipóteses de negócio para avaliar riscos de <i>software</i> .	-
Crescimento nas vendas de novos produtos (CVNP)	Mede o aumento nas vendas de novos produtos.	Apesar da boa prática da métrica, restringe-se a venda e não avalia as partes do produto que os clientes mais gostam.	POC, ODVC, NPS
Orçamento Dedicado à Verificação do Cliente (ODVC)	Mede os orçamentos com dedicação de análise e verificação do cliente.	Restringe-se em validar o orçamento junto ao cliente. Caso o escopo mude (que é bem provável que ocorra), o orçamento teria que ser revalidado.	POC, CVNP, NPS

Projetos Orientados para o Cliente (POC)	Mede o número de projetos criados e orientados para o cliente.	Dado que todos os projetos deveriam ser orientados ao cliente, não existe muita aplicabilidade desse métrica.	ODVC, NPS
Projetos respeitando custos e orçamento (PRCO)	Mede o número de projetos e programas respeitando custos e orçamento.	Em projetos ágeis o custo é fixo. Essa medida restringe a avaliação da relação de valor entregue versus custos realizado.	TED, POC, ODVC, HT, MR
Marcos Respeitados (MR)	Mede o número de marcos respeitados e entregues.	Marcos ou épicos, restringe-se em avaliar a entrega sem avaliar também os resultados do <i>feedback</i> dos clientes.	POC, ODVC
Componentes Reutilizáveis (CR)	Mede o número de componentes de software reutilizáveis dentro ou entre projetos.	Restringe-se na reutilização de componentes de software, mas poderia verificar o quanto desse reuso agregou na redução de falhas e desempenho.	NI, NM, NP, BRUT
<i>Chidamber-Kemerer</i> (CK)	Análises de complexidade de código orientado a objetos em Java.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NI, NM, NP, BRUT
<i>Mean Time to Execution</i> (MTE)	Mede o desempenho do código com o tempo médio de execução	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	SLC
<i>Source Lines of Code</i> (SLC)	Mede o número de linhas do código-fonte do programa.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	MTE
<i>Coupling between Objects</i>	Representa o número de classes acopladas a uma determinada classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	MS, QCF, CC
<i>Response for a Class</i>	Número total de métodos que potencialmente podem ser executados em resposta a uma mensagem recebida por um objeto de uma classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
<i>Weighted Methods per Class</i> (WMC)	Mede a soma da complexidade dos métodos em uma classe.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NI, NM, NP, BRUT
Bugs Resolvidos por Unidade de Tempo (BRUT)	Métrica que olha para a quantidade de <i>bugs</i> resolvidos por unidade de tempo.	Com restrição de falhas resolvidas, não faz avaliação se a redução contribuiu para o <i>time-to-market</i> .	NM
Nível de Alinhamento da Funcionalidade (NAF)	Mede o nível de alinhamento das funcionalidades com as estratégias organizacionais.	Com restrição de alinhamento funcional com estratégia organizacional, não faz avaliação se tal alinhamento resulta em valor de negócio.	NC, NPS
Nível de Confiabilidade (NC)	Mede o nível de confiabilidade em termos de qualidade de software.	O nível de confiabilidade pode ser uma parte da avaliação de valor para segurança da informação.	NAF, NM

Nível de Eficiência e Eficácia (NEE)	Mede o nível de eficiência em termos de qualidade de software.	O nível de eficiência pode ser uma parte da avaliação de valor.	NE, NC, NM, NI
Nível de Eficácia (NE)	Mede o nível de eficácia em termos de qualidade de <i>software</i> .	O nível de eficácia pode ser uma parte da avaliação de valor.	NRR, NM, NI
Nível de Integridade (NI)	Mede o nível de integridade em termos de qualidade de software.	O nível de integridade pode ser uma parte da avaliação de valor para segurança da informação.	BRUT, NC, NP
Nível de Manutenção (NM)	Mede o nível de manutenibilidade em termos de qualidade de <i>software</i> .	O nível de manutenibilidade pode ser uma parte da avaliação de <i>time-to-market</i> .	BRUT, NC, NP
Nível de Portabilidade (NP)	Mede o nível de portabilidade do software com outras arquiteturas de <i>software</i> .	Restringe-se na reutilização de componentes de software, mas poderia verificar o quanto desse reuso agregou na redução de falhas e desempenho.	NEE, NM
Teste em Canvas	Métrica de transformação <i>Lean</i> no plano de teste de software para simplificar o processo de teste.	Enxugar o processo de teste pode contribuir apenas ao <i>time-to-market</i> .	NEE, NM, BRUT
<i>Net Promoter Score</i>	Mede o nível de satisfação dos clientes em relação do produto recebido e sua experiência com ele.	Métrica comumente utilizada para avaliar e experiência geral dos clientes e não por determinadas partes do produto.	CVNP, ODVC, POC, PRCO, MR
Complexidade por acoplamento	Mede a complexidade por acoplamento que define o número de recursos de <i>software</i> por solução.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
Complexidade por coesão	Mede a complexidade por coesão que define o número de coesão intra componentes de <i>software</i> .	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF, CC
<i>Previsão por Planning Poker</i>	Mede o nível de compreensão e assertividade da estimativa.	Restringe-se em contribuir com o planejamento das <i>Sprints</i> .	-
Qualidade por <i>Sprint</i>	Mede o nível de qualidade do software por <i>Sprint</i> .	O nível de qualidade pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Incremento (QI)	Mede o nível de qualidade do software por <i>release</i> ou incremento.	O nível de qualidade por <i>release</i> ou incremento pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Solução (QS)	Mede o nível de qualidade do <i>software</i> por Solução.	O nível de qualidade por solução pode ser uma parte da avaliação de valor.	BRUT, NC
Qualidade por Portfólio (QP)	Mede o nível de qualidade do <i>software</i> por Portfólio de projetos.	O nível de qualidade por portfólio pode ser uma parte da avaliação de valor organizacional.	BRUT, NC
Qualidade por Visão Empresarial	Mede o nível de qualidade do software alinhado com a Visão Empresarial.	O nível de qualidade por visão empresarial pode ser uma parte da avaliação de valor organizacional.	BRUT, NC, CVNP, ODVC, POC, PRCO, MR
<i>Code smells</i> (CS)	Mede potenciais	Restringe-se em ser uma métrica técnica	QCF, QC

	problemas em código-fonte.	de engenharia que não sabemos se agregaria ao produto.	
Usou TDD (UT)	Mede se o projeto utilizou o <i>Test- Driven Development</i>	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	NUT
Não usou TDD (NUT)	Mede se o projeto não utilizou o <i>Test-Driven Development</i>	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	UT
Complexidade ciclomática (CC)	Mede a complexidade ciclomática (complexidade do código-fonte).	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QCF
Qualidade do código-fonte (QCF)	Mede a qualidade do código-fonte.	Restringe-se em ser uma métrica técnica de engenharia que não sabemos se agregaria ao produto.	CS, QC
Qualidade dos Eventos do Scrum (QES)	Mede a qualidade dos eventos do <i>Scrum</i> , como <i>Sprint Plannig, Daily, Sprint Review e Sprint Retrospective</i> .	Restringe-se em avaliar os eventos do processo do <i>Scrum</i> .	TED
Entrega Antecipada e Frequente de Software (EAFS)	Mede a relevância da produtividade entregando <i>software</i> com antecedência e frequência.	Avaliar a produtividade e a eficiência versus eficácia, sendo que a entrega antecipada e frequente reduz o <i>time-to-market</i> e aumenta o <i>feedback</i> .	QI, QS, QP
Agregação de valor	Mede o retorno do investimento como uma forma relevante de medir a agregação de valor	Medir o retorno do investimento pode ser uma parte da avaliação de valor.	NC, VCNP, ODVC, POC, PRCO, MR
Aspectos relacionados ao esforço	Mede a relevância do esforço de implantação de <i>software</i> que resolva os problemas de clientes.	Usar relevância do esforço de implantação de <i>software</i> não garante que a produção algo que não entrega o valor esperado.	ODVC, POC, PRCO
Modelo de previsão de defeitos	Modelo que mede a previsão de defeitos de <i>software</i> com base em outras métricas.	Restrito apenas para reduzir o risco de falhas durante as fases de teste de <i>software</i> .	EAFS, QI, QS, BRUT.
Segurança orientada a riscos	Métrica de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de <i>software</i> .	Medir a segurança orientada a risco pode ser uma parte da avaliação de valor para segurança da informação.	-
Melhoria de processo de <i>software</i>	Métricas para avaliar melhorias no processo de desenvolvimento de <i>software</i> .	Melhorar o processo de desenvolvimento pode contribuir apenas a entrega contínua e frequente de <i>software</i> .	EAFS, QES

Fonte: Elaborado pelo autor

Os artigos apresentaram métricas que avaliaram diferentes aspectos na avaliação no desenvolvimento de produto, seu feedback e o valor entregue. Assim, estudos apresentaram métricas baseadas em modelos de risco que podem ser usados para entender as características dos riscos de *software*, crescimento nas vendas de novos produtos, medição de projetos

orientados para o cliente, projetos e programas respeitando custos e orçamento, marcos respeitados e componentes de *software* reutilizáveis. No tocante a complexidade de programas orientados a objetos, autores dessa categoria propuseram as métricas de CK e CBO para avaliar o nível de significância e insignificância com propensão a falhas, estado de validação e utilidade na qualidade de software como *bugs* resolvidos por unidade de tempo. Outras métricas aplicáveis à arquitetura dos projetos e para quantificar o nível de alinhamento entre os projetos e os objetivos estratégicos, como métricas de funcionalidade, confiabilidade, eficiência, manutenibilidade, portabilidade e alinhamento de negócio. Para rastrear as abordagens ágeis, outros estudos apresentaram as métricas de complexidade, acoplamento e coesão. Outros autores mediram o nível de satisfação de clientes em projetos que usam o TDD e avaliaram o nível de satisfação de clientes em relação ao produto, onde identificou-se uma grande adoção da medida do NPS. Por fim, a seção seguinte apresenta, para cada categoria, a classificação das métricas utilizadas para cada prática, tanto ágil como preditiva.

8 CLASSIFICAÇÃO DAS MÉTRICAS POR CATEGORIA

A Tabela 8 destaca a classificação das métricas encontradas e suas respectivas categorias mapeadas durante o processo de RSL, além das subcategorias e processos de medição mapeados após a análise dos resultados. A classificação por Fase do Ágil apresentada na Tabela 8 foi inspirada nas etapas *Initiate, Discovery, Delivery e Release* dos projetos ágeis sugerido pelo *The Agile Landscape – Deloitte (TAL)*. Dessa forma, a categoria de Métricas de Projetos Tradicionais não foram classificadas segundo o TAL.

A partir das etapas *Initiate, Discovery, Delivery e Release* do modelo TAL e fazendo um relacionamento com as métricas identificadas na Tabela 8, são identificadas algumas lacunas no âmbito da medição em projetos ágeis. Dessa maneira, observa-se que a etapa de *Initiate* não foi incluída na classificação devido à ausência de métricas que avaliassem as ideias iniciais do produto para o projeto ágil como, por exemplo, medir o número de hipóteses de produto que se confirmaram como importantes para os clientes versus as que refutaram. Outra lacuna identificada, apesar de etapa de *Release* ser identificada na Tabela 8, muitas métricas de lançamento de produto com ou sem problemas de funcionamento, com *feedback* em nível de valor e funcional da parte do mercado e de clientes.

Assim, a etapa de *Release* somente contemplou métricas relacionadas a versão de lançamento, sem se preocupar com os resultados após o uso de tal versão no mercado e pelos

clientes. Em suma, as lacunas se apresentam nas etapas de *Initiate e Release*, onde são cruciais para se identificar quais ideias (*Initiate*) foram realmente convertidas em valor para o cliente (*Release*). Desde a gestão tradicional de projetos que a etapa de *Initiate* dá suporte à gestão de portfólio e programas. A etapa de *Discovery, Delivery e Release* suportam as iniciativas de projetos ou times. O nível de portfólio comumente gerencia um escopo e custos macro dos projetos. O nível de programa gerencia os projetos que possuem dependência técnica e/ou funcional (Lappi *et al.*, 2019). O nível de projeto ou times gerenciam as iniciativas em nível de time de desenvolvimento (Kock & Gemünden, 2019).

Tabela 8 - Classificação das métricas por categoria e subcategoria

Categoria	Subcategoria	Fase	Métrica	Resumo das Métricas
Métricas Ágeis de Backlog do Produto	Complexidade	<i>Delivery</i>	Grau de complexidade do projeto e KPI de Imperativos da técnica proposta.	Com o número de variantes de soluções tecnológicas, auxilia no planejamento de recursos, testar o desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos.
	Prioridade	<i>Delivery</i>	<i>Product Backlog Rating</i> e Avaliação da qualidade das histórias de usuários (HU).	Mede o nível de prioridade do item do <i>backlog</i> e ajuda na qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com fontes de requisitos, requisitos não- funcionais e versões de produtos.
	Monitoramento do Projeto	<i>Delivery</i>	Esforço de desenvolvimento de software, Esforço e produtividade e Grau de singularidade do projeto.	Avaliação da aplicabilidade ao longo do ciclo de vida, dependência da linguagem de programação específica e capacidade de prescrever soluções. Determina as características da entrega como o custo, prazo, possíveis desvios de prazo e custo.
Métricas Ágeis de Fluxo	Fluxo ou Processo	<i>Delivery</i>	Reduzir Lead Time, <i>Lead Time</i> , <i>Cycle Time</i> , <i>Efficiency flow</i> , <i>Tasks finished flow</i> , <i>Control Chart</i> , Capacidade do processo, Desempenho do processo e Maturidade do Processo.	Mede a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente e permite medir quantos itens do <i>backlog</i> são entregues por ciclo. Permite reduzir o <i>lead time</i> para alta capacidade de resposta às necessidades de clientes.
	Iteração ou Sprint	<i>Delivery</i>	Eficácia e eficiência, <i>Sprint Burn Down</i> , impedimentos, Custo do atraso, Custo hora real e planejada e Previsão por <i>Planning Poker</i> .	Realiza a medição por <i>Sprint</i> ou iteração considerando os pontos de história, nível de assertividade das estimativas, impedimentos, custos para medir a eficiência e eficácia do projeto.
	Incremento ou Artefato	<i>Delivery</i>	Clareza do Artefato, Envolvimento e Propriedade do Cliente, Aplicabilidade e Rastreabilidade	Medir a clareza, rastreabilidade e aplicabilidade do artefato do projeto, os níveis de envolvimento e propriedade do cliente aos entregáveis do projeto.

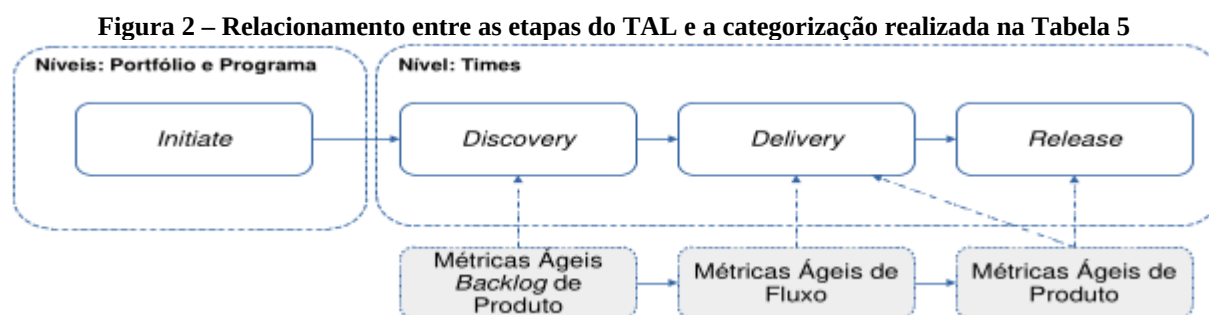
	Estimativa e Esforço	<i>Delivery</i>	<i>Epic Burn Down, Velocity</i> , Estimativa de tarefas e Esforço no projeto e Status Desenvolvimento.	Métrica sobre pontos de história no Épico (abrange uma ou muitas <i>Sprints</i>), mede o desempenho (<i>velocity</i>) através dos pontos de história feitos por <i>Sprint</i> . Realiza a estimativa de tarefas para ajudar na qualidade do software.
	Monitoramento do Projeto	<i>Delivery</i>	Componentes funcionais, Desempenho no projeto e Risco no projeto.	Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste e medir o desempenho do projeto. Medir os riscos do projeto ágil para dar visibilidade ao <i>backlog</i> do produto e sua priorização.
	Qualidade do Produto	<i>Delivery</i>	Rastreamento de tarefas, Atribuição de <i>bugs e Reopens</i> .	Rastreamento de tarefas para ajudar a garantir a qualidade e estabilidade do software. Através da atribuição da pessoa mais indicada para a resolução do problema aberto ou reaberto e reduzindo o time to <i>market</i> do produto.
Métricas Ágeis do Produto	Monitoramento do Projeto	<i>Delivery</i>	Horas trabalhadas em atividades, Tempo nos estágios de desenvolvimento, Aspectos relacionados ao esforço, Características dos riscos de software, Segurança orientada a riscos, Projetos/programas respeitando custos e orçamento e Marcos respeitados.	Analisa a quantidade de horas trabalhadas durante um período (<i>Sprint</i> ou ciclo), o tempo do requisito nos estágios de desenvolvimento, marcos e esforço para entregar ao cliente. Fornece informações sobre modelos de risco que podem ser usados para entender as características dos riscos de software.
	Qualidade do Código-Fonte	<i>Delivery /Release</i>	<i>Chidamber-Kemerer, Source Lines of Code, Coupling between Objects, Response for a Class, Weighted Methods per Class, Mean Time to Execution</i> , Componentes reutilizáveis, Complexidade por acoplamento, Complexidade por coesão, <i>Code smells</i> , Qualidade do código-fonte, Complexidade ciclomática, Usou <i>Test-Driven Development</i> , Não usou <i>Test-Driven Development</i> e Quantidade de <i>commits</i> .	Análises de complexidade de programas orientados a objetos, códigos-fonte publicados, de número de linhas de código, reuso de componentes e a avaliação da qualidade de implementação por melhores práticas orientadas a objetos com baixo acoplamento e alta coesão.

	Qualidade de Software	<i>Delivery/Release</i>	Bugs resolvidos por unidade de tempo, Nível de Confiabilidade, Nível de Eficiência/Eficácia, Nível de Eficácia, Nível de Integridade, Nível de <i>Manutenabilidade</i> , Nível de Portabilidade, Teste em <i>Canvas</i> , Qualidade por <i>Sprint</i> , Qualidade por <i>Release/Incremento</i> , Qualidade por Solução, Qualidade por Portfólio, Qualidade por Visão, Empresarial, Qualidade dos Eventos do Scrum e Modelo de previsão de defeitos.	Realiza a medição da quantidade de <i>bugs</i> resolvidos por unidade de tempo e mede o nível de confiabilidade, eficiência, eficácia, integridade, manutenibilidade e portabilidade em termos de qualidade de software. Utiliza práticas <i>Lean</i> no plano de teste de software para simplificar o processo de teste.
	Relação Clientes e Stakeholders	<i>Delivery/Release</i>	Crescimento nas vendas de novos produtos, Projetos orientados para o cliente, Orçamento dedicado à análise ou verificação do cliente, Nível de Alinhamento da Funcionalidade, Agregação de valor e <i>Net Promoter Score</i> .	Mede os orçamentos com dedicação de análise e verificação do cliente e o número de projetos criados e orientados para o cliente. Mede o nível de satisfação dos clientes em relação do produto recebido, sua experiência e agregação de valor.
	Entrega Contínua	<i>Delivery/Release</i>	Melhoria de processo de software e Entrega antecipada e frequente de software.	Utilizou métricas de produto para avaliar melhorias no processo de desenvolvimento de <i>software</i> e a relevância da produtividade entregando software com antecedência e frequência.
Métricas de Projetos Tradicionais	Análise do Valor Agregado	Não se aplica	EVM (<i>Earned Value Management</i>) e <i>Work Perform</i> .	Mede o desempenho e o progresso de projetos, com a capacidade de combinar medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido. Combina séries temporais e análise de regressão para desenvolver PV (<i>planned value</i>) e resposta EV (<i>earned value</i>) para prever <i>work perform</i> .
	Qualidade do Produto	Não se aplica	Monitoramento de defeitos, Qualidade geral dos sistemas do projeto e Mudança percentual relacionada à qualidade/valor.	Monitora os defeitos identificados durante o desenvolvimento do projeto e monitora a mudança do percentual relacionado à qualidade/valor do produto.

	Monitoramento do Projeto	Não se aplica	Variação de Desempenho, Número de defasagens, EDM (<i>Earned Duration Management</i>), Índice de Controle de Cronograma, Desempenho de cada atividade, Monitoramento de marcos e cronogramas, Análise <i>bootstrap</i> , <i>Project Health Index</i> , Esforço da tarefa e monitoramento, Índice de Controle de Custos, Análise do equilíbrio custo/risco, Análise de Risco do Cronograma com restrição de recursos e <i>Critical Success Factors</i> .	Análise de inicialização para avaliar o desempenho de um projeto e a classificação de problemas de projetos para identificar problemas em projetos que provavelmente falharão antes de começarem. Calcula o número de defasagens (atrasos) no cronograma e mede os custos do projeto. Utiliza o conceito de orçamento ótimo no nível do projeto e no nível da rede de atividades.
	Alinhamento Organizacional/Conhecimento	Não se aplica	Seleção de Projetos.	Integraram o aprendizado organizacional e o desenvolvimento do conhecimento adquirido através do processo de seleção de projetos.
	Requisitos do Produto	Não se aplica	Modelo RE4SA e Integridade de requisitos.	Modelo <i>Requirements Engineering for Software Architecture (RE4SA)</i> que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento. Avalia se os requisitos especificados são completos ou não para o desenvolvimento.
	Análise da Qualidade do Cronograma	Não se aplica	Proporção de tarefas detalhadas para marcos, Porcentagem de tarefas no caminho crítico, Tipo de relacionamento, Marcos sem predecessor ou sucessor e Impacto Predecessor no Sucessor.	Calcula o percentual de tarefas do projeto no caminho crítico do cronograma e investiga a proporção de tarefas do projeto que foram detalhadas para marcos específicos. Avalia os marcos do cronograma sem definição de predecessor e sucessor e calcula o tipo de relacionamento de tarefas predecessoras.
	Gestão de Mudanças	Não se aplica	Tempo de resolução de pedido de mudança e Mudança percentual relacionada ao design.	Calcula o tempo de processamento de pedido de mudança no projeto e avalia a mudança percentual relacionada ao design.
	Relação Cliente/Stakeholders	Não se aplica	<i>Stakeholder Metrics- integrated Management Model</i>	Integra as partes interessadas com métricas de identificação, comunicação, engajamento, capacitação e atendimento.

Fonte: dados da pesquisa, 2022.

Dessa forma, a Figura 2 apresenta o relacionamento entre as etapas do TAL e as categorias identificadas durante essa pesquisa e destaca uma lacuna na ausência de métricas na etapa de *Initiate*. Assim, identificou-se métricas de Gestão Orçamentária (GO) que não se relacionam com todas as fases do TAL.



Fonte: Dados da pesquisa, 2022.

Conforme os achados das métricas da Tabela 8, a GO é compreendida por métricas como Custo do Atraso, Custo-Hora Real e Planejada, Desempenho no Projeto (Métricas Ágeis de Fluxo) e Projetos/Programas respeitando Custos e Orçamento (Métricas Ágeis do Produto). Conforme o PMI (2021), a gestão de projetos conta com três medidas básicas de gestão de custos, sendo o Custo Real (CR) comparado ao Custo Planejado (CP), Variação de Custo (*Cost Variance - CV*) e o Índice de Desempenho de Custo (*Cost Performance Index - CPI*). Assim, PMI (2021) determina que o CR comparado ao CP, trata-se de uma medida de custo que compara o custo realizado de mão de obra ou recursos com o custo estimado; o CV é uma variação de custo simples sendo determinada comparando o custo realizado de uma entrega com o custo estimado e, quando usado com gerenciamento de valor agregado, é a diferença entre o valor agregado e o custo real; e o CPI é uma medida de gerenciamento de valor agregado que indica a eficiência com que o trabalho está sendo executado em relação ao custo orçado do trabalho. Assim, entende-se a oportunidade de medir o custo planejado *versus* o realizado dos entregáveis do projeto por ciclos menores iniciando desde a execução de Iterações ou *Sprints*, dos entregáveis realizados por trimestre ou *release* pelo programa, e as entregas esperadas em nível de portfólio de projetos.

Grimaldi *et al.* (2016) apresentaram como uma estrutura ágil escalável, baseada em SAFe 3.0 e mostrou como implementar e medir o desempenho considerando que as funções e responsabilidades são divididas entre a organização e o fornecedor. Grimaldi *et al.* (2016) utilizaram métricas como eficácia e eficiência, custo-hora real e planejada, impedimentos e custo do atraso durante as rodadas de *Sprint*. No mesmo sentido que Ahmed *et al.* (2017) discutiram sobre as métricas *Sprint Burn Down*, *Epic Burn Down Velocity Chart*, *Control*

Chart e a maneira como muitos dos cálculos de métricas são fortemente acoplados aos pontos de história para itens como o *Épico* ou *Sprint*.

Do ponto de vista de Petersen e Wohlin (2011), o *Lean* fornece ferramentas de análise e melhoria com foco no ciclo de vida do desenvolvimento, enquanto o ágil se concentra em soluções e prescreve conjuntos de práticas para obter agilidade. Petersen e Wohlin (2011) sugeriram análises sobre as métricas como reduzir o *lead time* para obter alta capacidade de resposta às necessidades dos clientes e fornecer um sistema de rastreamento que mostre o status do desenvolvimento de *software*. No sentido do monitoramento, as métricas de processo propostas por Choraś *et al.* (2020) foram utilizadas continuamente para ajudar a garantir a qualidade e estabilidade do *software* como a estimativa de tarefas e rastreamento tarefas em tempo real. Assim, pode-se identificar um *gap* sobre a ausência de métricas que aborde todas as etapas do projeto com objetivo de obter novas discussões sobre um processo de medição desde o portfólio, programa e time de projetos para trabalhos futuros é necessário. Esse contraponto sobre as métricas apresentadas pela Tabela 8 pode ser dividido em dois cenários dentro do contexto das práticas de gestão ágil de projetos: 1) escassez de propostas de métricas que venham medir as fases de *initiate*, *discovery*, *delivery* e a entrega contínua (*release*), mesmo que isoladamente, no desenvolvimento de um produto e 2) escassez de propostas de um processo de medição das etapas de *discovery*, *delivery* e entrega contínua para que assim seja possível medir e monitorar o projeto desde a iniciação até a entrega em partes (*Sprints*) e/ou no encerramento do projeto.

Com base na Tabela 5 e aos cenários 1 e 2 acima apresentados, verifica-se ainda a lacuna de processos de gestão ágil de projetos que contribuam como um *framework* de métricas ágeis capaz de ser um instrumento gerencial de maneira a orientar a liderança na gestão de projetos ágeis e para a tomada de decisão a partir de métricas estratégicas para a organização e para o desenvolvimento ágil de produtos. Os projetos ágeis que desenvolvem produtos, precisam testar hipóteses para resolver problemas organizacionais ou de clientes. Nesse sentido que o resultado oriundo das hipóteses deve ser medido para a tomada de decisão de refutar ou seguir adiante com a estratégia organizacional.

9 CONSIDERAÇÕES FINAIS

A partir da pesquisa realizada nas bases Web of Science e Scopus, a RSL encontrou-se 83 artigos que foram selecionados, dos quais apenas 42 abordavam aspectos relacionados a métricas de projetos tradicionais, 10 sobre as métricas ágeis de fluxo, 7 sobre as métricas ágeis de backlog do produto e 24 sobre as métricas ágeis de produto.

Apesar do histórico da adoção dos projetos tradicionais existir há décadas, o total dos artigos sobre as métricas em projetos tradicionais (40) encontradas nesse estudo não apresentaram um número tão maior que os resultados das métricas ágeis (43 artigos). Assim, percebeu-se que as métricas tradicionais focaram em apresentar a métrica do EVM com a adoção de melhorias na métrica, tal como melhorar a previsibilidade de riscos e alcançar o melhor valor agregado e o sucesso do projeto. Outras métricas muito citadas foram as adaptações inspiradas no EVM como o EDM e a utilização do SPI buscando medir o desempenho do cronograma.

Nas métricas ágeis, percebeu-se que as métricas se concentraram no discovery e no delivery, ou seja, em relação a medir o backlog do produto (9 artigos) no discovery e no fluxo e produto, tivemos 34 artigos no delivery. As métricas de *backlog* do produto se concentraram em medir a qualidade do requisito, enquanto as de produto, verificou-se uma grande concentração de métricas com preocupações com a qualidade do produto, ou seja, no tocante a quantidade de *bugs* resolvidos, tempo de resolução de problemas, qualidade do código-fonte e sobre a satisfação do cliente. Sobre as métricas de fluxo, identificou-se métricas como o *Lead Time*, *Velocity*, *Efficiency Flow*, *Cycle Time*, *Sprint Burn Down*, *Epic Burn Down*, *Control Chart*, entre outras. Assim, a Figura 2 apresenta uma lacuna de métricas que suporte um processo de gestão de projetos a partir do portfólio, programa e times de projeto e não apenas a gestão e métricas isoladas nas etapas de *discovery*, *delivery* e *release*. Além disso, na fase de *Initiate*, pode-se identificar um *gap* sobre a ausência de métricas para a gestão de iniciativas para o portfólio de projeto. Por fim, entende-se com esse estudo que ainda existe uma lacuna de processos de gestão ágil de projetos que contribuam como um *framework* capaz de ser um instrumento gerencial com métricas estratégicas para a organização e para projetos de desenvolvimento ágil de produtos. Assim, projetos ágeis precisam testar hipóteses para resolver problemas organizacionais ou de clientes e precisam ser medidas para a tomada de decisão do que fazer a seguir.

Espera-se que este estudo possa contribuir com outras pesquisas que desejam identificar as métricas utilizadas entre as abordagens ágil e tradicional de gestão de projetos. Além disso, orientar novas pesquisas para trabalhos futuros no sentido de analisar uma

possível lacuna entre na utilização de métricas tradicionais em projetos ágeis, que se compreende também em ser uma limitação dessa pesquisa. Além disso, é preciso estender a pesquisa para uma verificação em bases de patentes e na literatura cinzenta.

REFERÊNCIAS

Abdelrahman Aljemabi, M. *et al.* (2020). Mining social collaboration patterns in developer social networks. *IET Software*, 14(7), 839-849.

Ahmed, A. R. *et al.* (2017). Impact of story point estimation on product using metrics in scrum development process. *Intern. Journal of Advanced Computer Science and Applications*, 8(4).

Aljedaibi, W. & Khamis, S.(2019).Towards Measuring the Project Management Process During Large Scale Software System Implementation Phase.The ISC Intern. Journal of Info. Security,11(3),161-172.

Antinyan, V. (2014). Monitoring evolution of code complexity and magnitude of changes. *Acta Cybernetica*, 21(3), 367-382.

Asha, N. & Mani, P. (2018). Knowledge-based Acceptance Test driven agile Approach for Quality Software Development. *International Journal of Resent Tech. and Engineering*, 7.

Athanasiou, D. *et al.* (2014). Test code quality and its relation to issue handling performance. *IEEE Transac. on Software Eng*, 40(11),1100-1125.

Azzeh, M.& Nassif, A. B. (2016). A hybrid model for estimating software project effort from Use Case Points. *Applied Soft Computing*, 49, 981-989.

Bakhtiary, V. *et al.* (2020).The effectiveness of test-driven development approach on software projects: A multi-case study.*Bulletin of Electrical Eng and Info.*,9(5),2030-2037.

Batselier, J.& Vanhoucke, M. (2015). Evaluation of deterministic state-of-the-art forecasting approaches for project duration based on earned value management. *International Journal of Project Management*, 33(7), 1588-1596.

Budacu, E. N.& Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, 22(4), 70-79.

Chang, C. P. (2015). Software risk modeling by clustering project metrics. *International Journal of Software Engineering and Knowledge Engineering*, 25(06), 1053-1076.

Chen, H. L. (2018). Early prediction of project duration: A longitudinal study. *Engineering Management Journal*, 30(3), 191-202.

Chen, H. L. (2021). Early identification of distressed capital projects: A longitudinal approach. *International Journal of Managing Projects in Business*.

Chen, H. L. (2014). Improving forecasting accuracy of project earned value metrics: Linear modeling approach. *Journal of Management in Engineering*, 30(2), 135-145.

Chen, H. L., Chen, W. T.& Lin, Y. L. (2016). Earned value project management: Improving the predictive power of planned value. *International Journal of Project Management*, 34(1), 22-29.

- Chen, Z. *et al.* (2020). A Bayesian approach to set the tolerance limits for a statistical project control method. *International Journal of Production Research*, 58(10), 3150-3163.
- Choraś, M. *et al.* (2020). Measuring and improving agile processes in a small-size software development company. *IEEE access*, 8, 78452-78466.
- Choudhury, I. (2019). *Agile Methods for Engineering*. In *Management for Scientists*. Emerald Publishing Limited.
- Costa, J., Vasconcelos, A. & Fragoso, B. (2020). An Enterprise Architecture Approach for Assessing the Alignment Between Projects and Goals. *International Journal of Information Technology Project Management (IJITPM)*, 11(3), 55-76.
- Colin, J. *et al.* (2015). A multivariate approach for top-down project control using earned value management. *Decision Support Systems*, 79, 65-76.
- Colin, J. & Vanhoucke, M. (2014). Setting tolerance limits for statistical project control using earned value management. *Omega*, 49, 107-122.
- Colin, J. *et al.* (2015). A comparison of the performance of various project control methods using earned value management systems. *Expert Systems with Applications*, 42(6), 3159-3175.
- Concas, G. *et al.* (2012). An empirical study of software metrics for assessing the phases of an agile project. *International Journal of Software Engineering and Knowledge Engineering*, 22(04), 525-548.
- Day, F. C. *et al.* (2019). Feasibility study of an EHR-integrated mobile shared decision making application. *International journal of medical informatics*, 124, 24-30.
- Dennehy, D. & Conboy, K. (2018). Identifying challenges and a research agenda for flow in software project management. *Project Management Journal*, 49(6), 103-118.
- Dias, K. R. S. *et al.* (2021). Hybrid Project Management Method for managing ICT project's scope: a case study in a Brazilian company. *Brazilian Journal of Develop.*, 7(8), 75984-76014.
- El Asmar, M. *et al.* (2013). Quantifying performance for the integrated project delivery system as compared to established delivery systems. *Journal of construction engineering and management*, 139(11), 04013012.
- Fleche, D. *et al.* (2017). Collaborative project: Evolution of computer-aided design data completeness as management information. *Concurrent Engineering*, 25(3), 212-228.
- Gao, K. *et al.* (2011). Choosing software metrics for defect prediction: an investigation on feature selection techniques. *Software: Practice and Exp.*, 41(5), 579-606.
- Geng, S. *et al.* (2018). Knowledge contribution as a factor in project selection. *Project Management Journal*, 49(1), 25-41.
- Glenwright, E. (2007). A survey of the 30 most serious flaws in scheduling. In *PMICoS 2007 Annual Conference*. Grau, D. & Back, W. E. (2015). Predictability index: Novel metric to assess cost and schedule performance. *Journal of Construction Eng. and Management*, 141(12), 04015043.
- Gren, L., Goldman, A. & Jacobsson, C. (2020). Agile ways of working: a team maturity perspective. *Journal of Software: Evolution and Process*, 32(6), e2244.
- Grimaldi, P. *et al.* (2016). An agile, measurable and scalable approach to deliver software applications in a large enterprise. *Intern. Journal of Agile Systems and Manag.* 9(4), 326-339.

- Han, S., Choi, J. O. & O'Connor, J. T. (2017). Quality of baseline schedules: Lessons from higher education capital facility projects. *Journal of Professional Issues in Engineering Education and Practice*, 143(1), 04016017.
- Hanna, A. S. (2016). Benchmark performance metrics for integrated project delivery. *Journal of Construction Engineering and Management*, 142(9), 04016040.
- Hanna, A. S. *et al.* (2012). Request for information: benchmarks and metrics for major highway projects. *Journal of construction engineering and management*, 138(12), 1347-1352.
- Hazır, Ö. (2015). A review of analytical models, approaches and decision support tools in project monitoring and control. *International Journal of Project Manag.* 33(4), 808-815.
- Hernández, G. *et al.* (2019). Productivity Metrics for an Agile Software Develop. Team: A Systematic Review. *Tecnológicas*, 22(SPE), 63-81.
- Ibba, S. *et al.* (2018). Initial coin offerings and agile practices. *Future Internet*, 10(11), 103.
- Ibrahim, M. W. *et al.* (2020). Quantitative comparison of project performance between project delivery systems. *Journal of management in engineering*, 36(6), 04020082.
- Iqbal, S. *et al.* (2012). Yet another set of requirement metrics for software projects.
- Isong, B. & Obeten, E. (2013). A systematic review of the empirical validation of object-oriented metrics towards fault-proneness prediction. *International Journal of Software Engineering and Knowledge Engineering*, 23(10), 1513-1540.
- Jethani, K. (2013). Software metrics for effective project management. *International Journal of System Assurance Engineering and Management*, 4(4), 335-340.
- Junior, V. B. *et al.* (2019). Elaboração de um Plano de Gerenciamento de Projeto para Implantação de Laboratório de Metrologia Utilizando o Guia PMBOK 5ª Edição. *Cippus*, 7(2), 127-145.
- Jyothi, V. E. & Rao, K. N. (2017). Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Eng. Research and Science*, 4(3), 237085.
- Kamulegeya, G. *et al.* (2018). Measurements in the early stage software start-ups: a multiple case study in a nascent ecosystem. *Found. of Computing and Decision Sciences*, 43(4), 251-280.
- Kayes, I. *et al.* (2016). Product backlog rating: a case study on measuring test quality in scrum. *Innovations in Systems and Software Eng.* 12(4), 303-317.
- Kerkhove, L.P. & Vanhoucke, M. (2017). Extensions of earned value management: Using the earned incentive metric to improve signal quality. *Intern. Journal of Project Management*, 35(2), 148-168.
- Kristiansen, J. N. & Ritala, P. (2018). Measuring radical innovation project success: typical metrics don't work. *Journal of Business Strategy*.
- Kim, T., Kim, Y. W. & Cho, H. (2016). Customer earned value: performance indicator from flow and value generation view. *Journal of management in Eng.* 32(1), 04015017.
- Ko, B. S. & Cho, M. S. (2020). Evaluating the relative efficiency of defense R&D projects in a multi-project environment using EVM and CAIV measures. *ICIC Express Letters, Part B*:

- Applications,11(7), 675-682. Kock, A.& Gemünden, H. G. (2019). Project lineage management and project portfolio success. *Project Management Journal*, 50(5), 587-601.
- Yamashita, A.& Counsell, S. (2013). Code smells as system-level indicators of maintainability: An empirical study. *Journal of Systems and Software*, 86(10), 2639-2653.
- Yousefi, N. *et al.* (2019). Using statistical control charts to monitor duration-based performance of project. arXiv preprint arXiv:1902.02270.
- Lappi, T. M. *et al.* (2019). Project governance and portfolio management in government digitalization. *Transforming Government: People, Process and Policy*, 13(2), 159-196.
- Mas, A. *et al.* (2020). Supporting the deployment of ISO-based project manag. processes with agile metrics. *Computer Standards & Interfaces*, 70, 103405.
- Mascarenhas Hornos da Costa, J., Oehmen, J., Rebentisch, E.& Nightingale, D. (2014). Toward a better comprehension of Lean metrics for research and product development management. *R&D Management*, 44(4), 370- 383.
- Mäkiäho, P. *et al.* (2022). MMT: a tool for observing metrics in software projects. In *Research Anthology on Agile Software, Software Develop and Testing* (pp. 1077-1089). IGI Global.
- Meidan, A. *et al.* (2018). Measuring software process: a systematic mapping study. *ACM Computing Surveys (CSUR)*, 51(3), 1-32.
- Narbaev, T. *et al.* (2014). Combination of growth model and earned schedule to forecast project cost at completion. *Journal of Construction engineering and management*, 140(1), 04013038.
- Nidagundi, P.& Novickis, L. (2016). Introduction to Lean Canvas Transformation Models and Metrics in Software Testing. *Appl. Comput. Syst.*, 19(1), 30.
- Orgut, R. E. *et al.* (2018). Metrics that matter: Core predictive and diagnostic metrics for improved project controls and analytics. *Journal of Construction Engineering and Management*, 144(11), 04018100.
- Pajares, J.& Lopez-Paredes, A. (2011). An extension of the EVM analysis for project monitoring: The Cost Control Index and the Schedule Control Index. *International Journal of Project Management*, 29(5), 615-621.
- Parrend, P. *et al.* (2014, May). Swarm Projects: Beyond the Metaphor. In *International Conference on Swarm Intelligence Based Optimization* (pp. 131-138). Springer, Cham.
- Pavlova, N. *et al.* (2021). Creating the Agile Model to Manage the Activities of Project Oriented Transport Companies. *Eastern-European Journal of Enterprise Tech*, 1(3), 109.
- Perkusich, M. *et al.* (2017). Assisting the continuous improvement of Scrum projects using metrics and Bayesian networks. *Journal of Software: Evolution and Process*, 29(6), e1835.
- Petersen, K.& Wohlin, C. (2011). Measuring the flow in lean software development. *Software: Practice and experience*, 41(9), 975-996.
- Pradhan, S.& Nanniyur, V. (2021). Large scale quality transformation in hybrid development organizations—A case study. *Journal of Systems and Software*, 171, 110836.

Prakash B. *et al.* (2018). Ontology based risks management model for agile software development. *Journal of Advanced Research in Dynamical and Control Systems*, 10(11 Special Issue):319–330.

Pressman, R. S.& Maxim, B. R. (2021). *Engenharia de software-9*. McGraw Hill Brasil.

Project Management Institute (PMI) (2021): *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, PMI, Newtown Square, PA, 2021.

Pozzana, I. *et al.* (2021). Spreading of performance fluctuations on real-world project networks. *Applied Network Science*, 6(1), 1-15.

Rajablu, M. *et al.* (2017). Managing for stakeholders: Introducing stakeholder metrics-integrated model to lead project ethics and success. *International Journal of Project Organisation and Management*, 9(1), 31-56.

Rajagopalan, J. *et al.* (2018). Introduction of a new metric “Project Health Index”(PHI) to successfully manage IT projects. *Journal of Organizational Change Management*.

Ries, E. (2011). *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.

Royce, W. (1970). The software lifecycle model(waterfall model). In *Proc. Westcon*, V.314.

Sampaio, R. F.& Mancini, M. C. (2007). Systematic review studies: a guide for careful synthesis of the scientific evidence. *Brazilian Journal of Physical Therapy*, 11, 83-89.

Sato, T.& Hirao, M. (2013). Optimum budget allocation method for projects with critical risks. *International Journal of Project Management*, 31(1), 126-135.

Savola, R. *et al.* (2012). Risk-driven security metrics in agile software development: An industrial pilot study. *Journal of Universal Computer Science*, 18(12), 1679-1702.

Sadler, H. (2020). ER2C SDMLC: enterprise release risk-centric systems development and maintenance life cycle. *Software Quality Journal*, 28(4), 1755-1787.

Sajedi-Badashian,A. & Stroulia,E.(2020).Investigating the info. value of different sources of evidence of developers' expertise for bug assignment in open-source projects. *IET Software*,14(7),748-758.

Sharma, V. *et al.* (2021). Development of metrics and an external benchmarking program for healthcare facilities. *International Journal of Construction Management*, 21(6), 615-630.

Shawky, D. M.& Abd-El-Hafiz, S. K. (2016). Characterizing software development method using metrics. *Journal of Software: Evolution and Process*, 28(2), 82-96.

Staron, M. & Meding, W. (2016). Mesram—a method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation. *Journal of Systems and Software*, 113, 76-100.

Singh, J., Singh, K.& Singh, J. (2019). Reengineering framework to enhance the performance of existing software. *System*, 1139, 120-3.

Simpson, J. R. *et al.* (2021). Optimizing the electronic health record: An inpatient sprint addresses provider burnout and improves electronic health record satisfaction. *Journal of the American Medical Informatics Association*, 28(3), 628-631.

Spijkman, T. *et al.* (2021). Alignment and granularity of requirements and architecture in agile development: A functional perspective. *Information and Software Technology*, 133, 106535.

Sommerville, I. (2007). *Eng. de Software*, 9. Ed. Pearson, Addison Wesley, 8(9), 10, p.44-45.

Song, J., Martens, A. & Vanhoucke, M. (2021). Using schedule risk analysis with resource constraints for project control. *European Journal of Operational Research*, 288(3), 736-752.

Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante, 2019.

Tahir, T. *et al.* (2018). A systematic mapping study on software measurement programs in SMEs. *E-Informatica Software Engineering Journal*, 12(1).

Tanveer, B. *et al.* (2019). An evaluation of effort estimation supported by change impact analysis in agile software develop. *Journal of Software: Evolution and Process*, 31(5), e2165.

Tranfield, D. *et al.* (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, 14(3), 207-222.

Wallace, L. G. & Sheetz, S. D. (2014). The adoption of software measures: A technology acceptance model (TAM) perspective. *Information & Management*, 51(2), 249-259.

Wauters, M. & Vanhoucke, M. (2015). Study of the stability of earned value management forecasting. *Journal of Construction Engineering and Management*, 141(4), 04014086.

Van Casteren, W. (2017). *The Waterfall Model and the Agile Methodologies: A comparison by project characteristics*. Research Gate, 1-6.

Vanhoucke, M. (2011). On the dynamic use of project performance and schedule risk information during project tracking. *Omega*, 39(4), 416-426.

Vanhoucke, M. (2019). Tolerance limits for project control: An overview of different approaches. *Computers & Industrial Engineering*, 127, 467-479.

Üsfekes, Ç. *et al.* (2019). Auction-based serious game for bug tracking. *IET Software*, 13(5), 386-392.

Zheng, M. *et al.* (2021). Key performance indicators for the integration of the service-oriented architecture and scrum process model for IOT. *Scientific Programming*, 2021.

10.3 ESTUDO DE PATENTES SOBRE MÉTRICAS PARA A GESTÃO ÁGIL DE PROJETOS

MSc José da Silva Azanha Neto (Universidade Nove de Julho)

jose.azanha@uni9.edu.br

Prof. Dr. Renato Penha (Universidade Nove de Julho)

renato.penha@uni9.pro.br

RESUMO

Em um contexto de novos *frameworks* e práticas para a gestão de projetos ágeis em constante mudança, a importância de realizar o monitoramento e controle de projetos por meio de métricas se faz necessário. São respostas às rápidas mudanças e a agilidade organizacional que precisam ser medidas em todo o ciclo de vida dos projetos. O objetivo desse artigo foi realizar uma pesquisa em base de dados de patentes de domínio público para identificar possíveis patentes sobre o desenvolvimento de um processo orientado a métricas ágeis. Como estratégia metodológica foi adotado uma análise exploratória com base em informações disponibilizadas nos bancos de dados de patentes em domínio público. Para a coleta de dados, foram utilizadas as ferramentas PATENTSCOPE, ESPACENET e LENS para o procedimento de pesquisa qualitativa exploratória em patentes. Os resultados apontaram a existência de patentes que propuseram métricas ágeis somente nas etapas de delivery e release, o que abre uma lacuna para outras etapas em projetos ágeis que precisam de medição para atender as necessidades da gestão de projetos ágeis.

Palavras-chave: Gestão Ágil de Projetos; Métricas Ágeis; *Agile Landscape*; Desenvolvimento de *Software*; *Discovery e Delivery*.

ABSTRACT

In a context of new frameworks and practices for managing agile projects in constant change, the importance of monitoring and controlling projects through metrics is necessary. These are responses to rapid change and organizational agility that need to be measured throughout the project lifecycle. The objective of this article was to conduct a search in a public domain patent database to identify possible patents on the development of a process oriented to agile metrics. As a methodological strategy, an exploratory analysis was adopted based on information available in patent databases in the public domain. For data collection, the PATENTSCOPE, ESPACENET and LENS tools were used for the exploratory

qualitative research procedure on patents. The results pointed to the existence of patents that proposed agile metrics only in the delivery and release stages, which opens a gap for other stages in agile projects that need measurement to meet the needs of agile project management.

Keywords: Agile Project Management; Agile Metrics; Agile Landscape; Software development; Discovery and Delivery.

1 INTRODUÇÃO

A abordagem ágil em gestão de projetos para o desenvolvimento de software tem crescido ao longo dos anos, desde o advento do Manifesto Ágil em 2001 (Agile Manifesto, 2001). Como Penha, da Silva e Russo (2020) destacaram, a gestão ágil de projetos inclui processos, técnicas e ferramentas de agilidade para introduzir o desenvolvimento contínuo e incremental de software de valor aos clientes e stakeholders. (Sutherland, 2019). Essa visão vai de encontro do modelo de desenvolvimento de software e de gestão tradicional com grandes etapas de planejamento somente no início do projeto. (Sommerville, 2007; Pressman, 2021). Conforme Sutherland (2019), Rubin (2017) e Adkins (2020), novas abordagens para gestão de projetos ágeis surgiram para suportar o processo incremental para entrega de software, tais como o Extreme Programming, Scrum, Kanban, SAFe, entre outras.

Em relação ao desempenho dos projetos desenvolvimento de *software*, Jyothi e Rao (2017) e dos Santos *et al.* (2018) destacam que na gestão ágil de tais projetos, o contexto muda para o valor entregue ao cliente ao invés de, segundo o *Project Management Institute* (PMI, 2017), as métricas tradicionais orientadas ao planejamento total do projeto, com destaque ao *Earned Value Management* (EVM). Nesse sentido, as métricas tradicionais utilizadas para apoio aos líderes podem ser consideradas inadequadas, ou se aplicadas fora do contexto de gerenciamento ágil de projetos de desenvolvimento de software (Budacu & Pocatilu, 2018), podendo causar impactos no controle e gerenciamento de tais projetos, como a visão tradicional de escopo fixo quando no ágil o escopo é estimado e a utilização do EVM em projetos ágeis quando o ágil considera o custo e o tempo são fixos. (Misra & Omorodion, 2011; Oza & Korkala, 2012).

Segundo Jyothi e Rao (2017) e dos Santos *et al.* (2018), em um ambiente ágil de projetos de desenvolvimento de software, as métricas se concentram, predominantemente, na etapa de *delivery* e *release* do projeto. Para Jyothi e Rao (2017), o *delivery* é uma etapa de desenvolvimento e entrega de um incremento de *software* e a *release* como uma etapa de

entrega de uma versão estável para a produção, clientes e mercado. Budacu e Pocatilu (2018) destacaram que as métricas ágeis mais utilizadas são (a) *velocity*; (b) *Work Item Age* (WIA); (c) *Throughput*, (d) *Cycle Time* (CT) e (e) *Lead Time* (LT).

O *velocity* é uma métrica para avaliação utilizada no desempenho de entrega dos desenvolvedores de *software* por meio dos pontos de histórias do usuário. O WIA é uma métrica para medir o tempo em dias que um requisito está em uma etapa ou ciclo de desenvolvimento de *software*. Já o *Throughput* representa uma métrica para avaliação da capacidade de entrega da equipe ao final de cada iteração. O CT conta o tempo em que um requisito está em um determinado ciclo ou etapas de desenvolvimento de *software*, enquanto o LT é considerado uma medida que avalia o tempo em que um requisito funcional é comprometido para início do desenvolvimento *software* até o mesmo ser entregue ao cliente, mercado ou *stakeholder*. (Budacu & Pocatilu, 2018). Em contrapartida, as métricas tradicionais se concentram na etapa de controle e monitoramento do projeto com objetivo de controlar o planejamento *versus* o realizado, como o EVM (PMI, 2017).

Apesar disso, as métricas calculadas apenas nas etapas de *delivery* e *release* não oferecem a visão de *Key Performance Indicators* (KPIs), o qual a liderança empresarial necessita para a tomada de decisão na gestão de projetos e estratégia organizacional (Jyothi & Rao, 2017). Em outras palavras, Budacu e Pocatilu (2018) apresentaram que o uso de tais métricas ainda é muito fragmentado em questões operacionais do projeto e que não garante que clientes, executivos da organização e *stakeholders* estejam alinhados entre valor percebido pelo cliente as metas da organização. Assim, a utilização de métricas apenas nas etapas de *discovery* e *delivery* não fornecem a visibilidade nas etapas que precedem a gestão de projetos e quando se busca colocar o cliente no centro no desenvolvimento de produtos de *software*.

Assim, o objetivo desse artigo foi realizar uma pesquisa patentária em base de dados de domínio público para identificar se existem patentes sobre o desenvolvimento de um processo orientado a métricas ágeis. Vale destacar que a pesquisa procurou identificar métricas em todas as etapas do modelo *The Agile Landscape – Deloitte* (TAL) (Paragano, 2021), que compreendem o *initiate*, *discovery*, *delivery* e *release*.

A pesquisa utilizou uma análise exploratória com base em informações disponibilizadas nos bancos de dados de patentes em domínio público (Creswell, 2014). Para a coleta de dados, foram utilizadas as ferramentas PATENTSCOPE, ESPACENET e

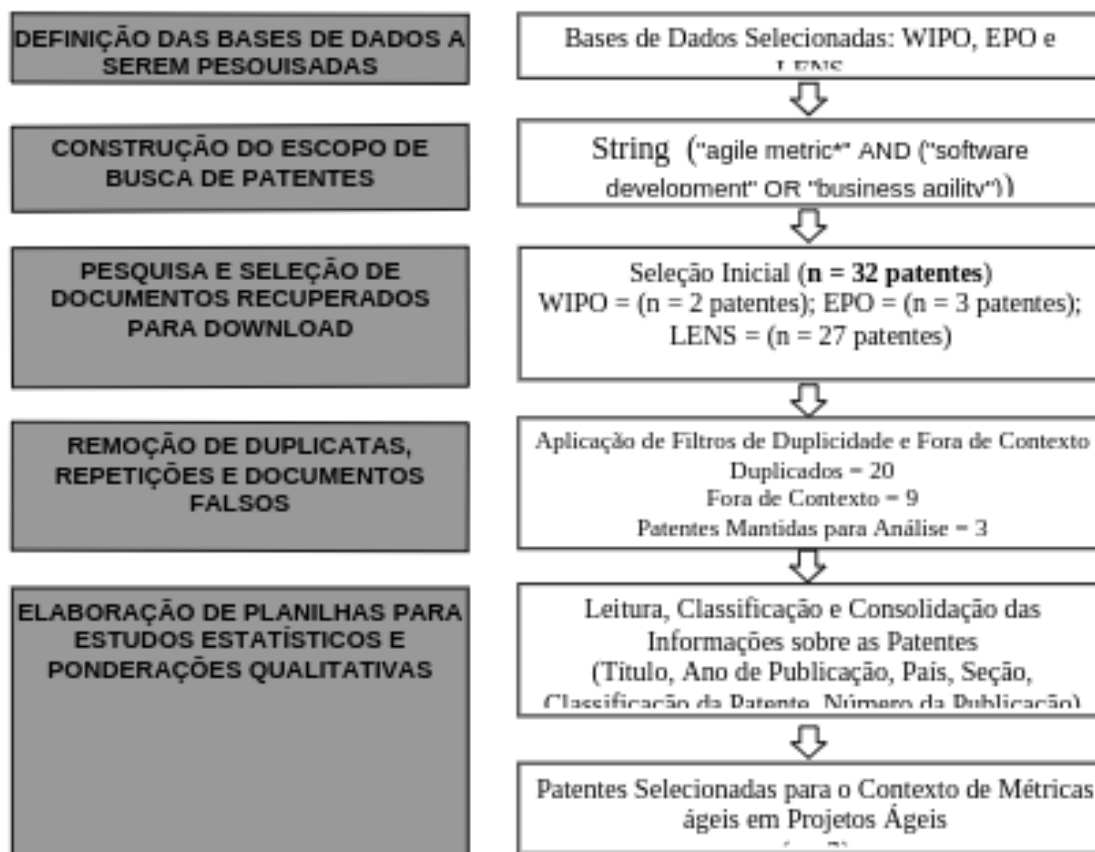
LENS - *Patent Search and Analyse* para o procedimento metodológico aplicado na pesquisa qualitativa exploratória em patentes como sugerido por Paranhos e Ribeiro (2018). Os resultados obtidos permitiram a identificação de métricas ágeis presentes em patentes para atender demandas de organizações no contexto de medição em gestão ágil de projetos. Os resultados do mapeamento das métricas ágeis nas patentes podem contribuir para a proposição de modelos centrados em auxiliar as organizações na operacionalização de métricas ágeis, bem como ajudar as organizações a criarem processos orientados às métricas ágeis mais adequadas para um contexto que envolve a entrega de valor para clientes e alinhamento organizacional e estratégico.

A próxima seção abordará a fundamentação teórica que norteia este estudo: etapas de concepção de requisitos, etapas de desenvolvimento de *software*, métricas ágeis e patentes. Em seguida, o método metodológico adotado será apresentado através de seus procedimentos, seguidos da apresentação dos resultados e, por fim, a conclusão do artigo com a apresentação das limitações e recomendações da pesquisa para estudos futuros.

2 MATERIAIS E MÉTODOS

Esta pesquisa utilizou a estratégia de abordagem qualitativa e exploratória visando compreender se existem estudos com patentes registradas em torno do uso de métricas em gestão de projetos ágeis. A abordagem metodológica desse estudo seguiu o proposto por Creswell (2014) por meio da pesquisa qualitativa explanatória. No campo da Administração, a pesquisa exploratória é comumente conduzida com propósitos profissionais e para subsidiar trabalhos acadêmicos. (Gil e Reis Neto, 2020). No tocante ao mapeamento de patentes presente no protocolo metodológico a seguir, foram seguidos cinco passos conforme sugerido por Paranhos e Ribeiro (2018): (1) Definição das bases de dados a serem pesquisadas; (2) Construção do escopo de busca de patentes; (3) Pesquisa e seleção de documentos recuperados para download; (4) Remoção de duplicatas, repetições e documentos falsos; (5) Elaboração de planilhas para estudos estatísticos e ponderações qualitativas. O protocolo metodológico é representado na Figura 1.

Figura 1. Procedimento metodológico aplicado na pesquisa qualitativa exploratória



Fonte: Elaborado pelos autores conforme Paranhos e Ribeiro (2018)

A Figura 1 apresenta todas as etapas do processo de Paranhos e Ribeiro (2018). A primeira etapa é descrita como a “Definição das Bases de Dados a Serem Pesquisadas” e, nessa etapa, foram selecionadas as bases European Patent Office (EPO), *World Intellectual Property Organization* (WIPO) e *Leans (Patent Search Service)* que são bancos de dados de patentes de conhecimento e domínio público. Como ferramental tecnológico foram utilizadas as ferramentas PATENTSCOPE, ESPACENET e LENS - *Patent Search and Analysis* para a coleta de dados sobre as patentes.

Na segunda etapa definida como “Construção do Escopo de Busca de Patentes”, foi definido o escopo de busca através da *string* de pesquisa (“*agile metric**” and (“*software development*” or “*business agility*”)) que foi executada nas bases de dados EPO, WIPO e LENS. Na próxima etapa definida por “Pesquisa e Seleção de Documentos Selecionados para *Download*” executada nas bases de dados WIPO, EPO e LENS, foram encontradas 32 patentes somadas com as três bases de dados, sendo duas patentes na base da WIPO, três patentes na base do EPO e 27 patentes na base do LENS. Em seguida, foi realizada a quarta fase de “Remoção de Duplicatas, Repetições e Documentos Falsos”, na qual foi verificada a

ocorrência de patentes repetidas entre as bases e a remoção de duplicatas num total de 20 casos. Ainda nessa etapa, identificou-se nove casos fora do contexto da pesquisa, restando assim um total final de três patentes para continuidade da análise.

Na última fase, “Elaboração de Planilhas para Estudos Estatísticos e Ponderação Qualitativa”, os dados de três patentes foram importadas para uma planilha no LibreOffice e para a elaboração das tabelas para sequência da análise. Dessa forma, as informações importadas na planilha foram tratadas para realizar uma análise mais detalhada com objetivo de compreender a representação das patentes selecionadas. Foi realizada a consolidação das informações quantitativas e o agrupamento das informações por meio da *International Patent Classification* (IPC) que representa uma forma de classificação que utiliza letras e números em sua constituição para indexar patentes (WIPO, 2023).

Além dessa classificação, realizamos uma análise para determinar se as métricas apresentadas nas patentes atendiam o modelo TAL que considera o ciclo de iterações em quatro etapas definidas por *Initiate*, *Discovery*, *Delivery* e *Release*. A etapa de *Initiate* representa a concepção das ideias iniciais do projeto e que constitui a um *Backlog* do Produto inicial (Sutherland, 2019). A etapa de *Discovery* representa o detalhamento dos requisitos, identificação de hipóteses de mercado a serem validadas, definição de jornada de usuário, protótipos e refinamento para ser objeto de entrada de uma iteração. A etapa de *Delivery* representa uma iteração onde são desenvolvidos os requisitos funcionais em forma de *software*. Por fim, a etapa de *Release* representa a liberação de uma versão estável de *software* ou um *Minimum Viable Product* (MVP) para clientes ou mercado.

A partir das abordagens de classificação das informações consolidadas e tratadas na planilha do LibreOffice, foram geradas tabelas e figuras para sintetizar os resultados e trazer à tona o entendimento das patentes selecionadas que serão apresentadas na próxima seção.

3 RESULTADOS

Esta seção apresenta os resultados do levantamento de patentes de conhecimento e domínio público. Inicialmente, é apresentado o mapeamento e a classificação das patentes que constituíram o corpus de análise da pesquisa. A seguir, são apresentadas as categorias destacadas após uma análise aprofundada do conteúdo das patentes.

3.1 MAPEAMENTO E CLASSIFICAÇÃO DE PATENTES

As três patentes identificadas na pesquisa que abordam aspectos de métricas ágeis em gestão ágil de projetos em um período que vai de 2013 a 2023. Na análise dos resultados, foram localizados os países envolvidos no depósito das patentes em bases de dados do LENS, WIPO e EPO. Apesar do baixo número de patentes selecionadas neste estudo, os resultados mostram que os Estados Unidos possuem três de patentes relacionadas com métricas ágeis e a China com apenas uma patente, conforme indicado na Tabela 1.

Tabela 1 - Classificação por IPC das patentes selecionadas nesse estudo

ID	Título	Ano de Publicação	País	Total de Patentes	Seção	Classificação da Patente	Número da Publicação	IPC
1	<i>System and method for computer development data aggregation</i>	2022	Estados Unidos	2	G	G06F	US 2021175015 78 A	G06F8/77
2	<i>Method and Apparatus for Calculating Performance Indicators</i>	2014	Estados Unidos		G	G06Q	US 2012137157 08 A	G06Q10/06
3	<i>GDASD task allocation method based on reliability</i>	2019	China	1	G	G06Q	CN1105556 13A	G06Q10/06

Fonte: Elaborado pelos autores com base nos resultados da pesquisa

Ao se observar a Tabela 1, é possível notar que as patentes estão classificadas nos grupos G06F e G06Q, sendo uma patente classificada no grupo G06F publicada nos Estados Unidos e duas patentes classificadas no G06Q, sendo uma patente publicada nos Estados Unidos e outra na China. Após identificar os países depositários de patentes que abordam o tema com métricas ágeis em projetos ágeis, incluindo como verificar a evolução do assunto ao longo dos anos e a demonstração das patentes de acordo com sua classificação. Dessa maneira, a próxima etapa visou avaliar, dentro da proposta do estudo, patentes que tratam de aspectos relacionados as métricas ágeis.

A Tabela 1 mostra que as patentes selecionadas estão concentradas em apenas uma seção de classificação G (física). Para contextualizar as patentes que compõem a seleção final, as características de cada patente foram analisadas quanto às métricas ágeis em gestão ágil de projetos. Dessa forma, a Tabela 2 destaca as patentes, apresenta o conceito e classificação entorno das etapas segundo o modelo TAL e as métricas utilizadas em cada uma dessas etapas. Além disso, a Tabela 2 apresenta que essa pesquisa somente localizou patentes nas etapas de *Delivery* e *Release* (TAL).

Tabela 2 - Patentes selecionadas conforme The Agile Landscape (Deloitte)

Contexto								
ID	Título	Objetivo	Delivery	Release	Métricas Delivery	Análise: Métricas Delivery	Métricas Release	Análise: Métricas Release
1	<i>System and method for computer development data aggregation</i>	Um sistema e método para integrar informações de desenvolvimento do projeto com base no ciclo de vida de desenvolvimento de software.	Aplica métricas de tempo de entrega, capacidade de entrega e da velocidade do time de desenvolvimento de software.	Aplica métricas de entrega automatizada de software, de qualidade de código-fonte e quantidade de publicações de código-fonte no repositório.	<i>Lead Time, Velocitye Througput</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Fluxo onde foram classificadas as métricas da etapa operacional ou de desenvolvimento de software.	<i>Deploy automatizado, Code Quality e Pull requests.</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Produto onde foram classificadas as métricas relacionadas com a qualidade de software (produto) e com a cultura DevOps.
2	<i>Method and Apparatus for Calculating Performance Indicators</i>	Um método e aparelho para gerar indicadores de desempenho e melhorar o desempenho do time de desenvolvimento de software com base em métricas.	Aplica métricas de tempo médio de entrega do time de desenvolvimento e do tempo médio de ciclo nas etapas de desenvolvimento.	Não se Aplica	<i>Throughtput Médio, Cycle Time e Cycle Time Médio.</i>	Trata-se de métricas relacionadas com a categoria de Métricas Ágeis de Fluxo onde foram classificadas as métricas da etapa operacional ou de desenvolvimento de software.	Não se Aplica	Não se Aplica
3	<i>GDASD task allocation method based on reliability</i>	Fornecer um método de alocação de tarefa com base na confiabilidade no problema de alocação de tarefas no	Não há métricas ágeis. Trata-se de aplicação de um método de alocação de tarefas com base na confiabilidade.	Não se Aplica	Não se Aplica	Não se Aplica	Não se Aplica	Não se Aplica

		desenvolvimento ágil distribuído remoto.						
--	--	--	--	--	--	--	--	--

Conforme a Tabela 2, a patente 1 apresentou as métricas ágeis nas etapas de Delivery e Release. A patente 2 apresentou somente as métricas ágeis na etapa de Delivery e, por fim, a patente 3 apresentou um método de alocação de tarefas em desenvolvimento ágil distribuído e remoto e, dessa forma, não apresentou relação alguma com o modelo TAL. Assim, a patente 1 apresentou as métricas relacionadas com a categoria de Métricas Ágeis de Fluxo (conforme o estudo 2) onde foram classificadas as métricas da etapa de desenvolvimento de software. Ainda na patente 1 temos as métricas relacionadas com a categoria de Métricas Ágeis de Produto (conforme o estudo 2), onde foram classificadas as métricas relacionadas com a qualidade de software e com as práticas de DevOps (Toh, Sahibuddin & Mahrin, 2019).

Em relação a adoção de métricas em projetos ágeis e apresentadas na Tabela 2, a etapa de Delivery contém as métricas de tempo de entrega, capacidade de entrega e da velocidade do time de desenvolvimento de software. Na etapa de Release, identificou-se as métricas de entrega automatizada de software (DevOps), de qualidade de código-fonte e quantidade de publicações de código-fonte no repositório de código.

Em relação as etapas do TAL de cada métrica ágil, a patente 1 apresentou as métricas ágeis de Lead Time, Velocity e Througput para a etapa de Delivery e as métricas ágeis de Deploy Automatizado, Code Quality e Pull Requests para a etapa de Release. A patente 2 apresentou as métricas de *Throughtput* Médio, *Cycle Time* e *Cycle Time* Médio para a etapa de *Delivery*. Por fim, a patente id 3 apresentou um método para calcular métricas de alocação de tarefas com base na confiabilidade e, assim, não se encaixa no contexto final dessa análise onde consideramos fundamental a classifica por meio do modelo TAL.

4 ANÁLISE E DISCUSSÃO

Os resultados apresentados pela Tabela 2 trouxeram apenas três patentes selecionadas para análise de métricas em projetos de desenvolvimento de software. Assim, somente duas patentes que apresentaram as métricas ágeis em projetos de desenvolvimento de *software* nas etapas de *Delivery* e *Release*, e com relacionamento com o modelo TAL. Além disso, as métricas desse estudo estão relacionadas com as categorias de Métricas Ágeis de Produto (conforme o estudo 2) e Métricas Ágeis de Fluxo (conforme o estudo 2) que são representadas pelo modelo TAL pelo *Delivey* e *Release*, respectivamente.

Assim, não foi possível encontrar um processo de medição contínuo nas bases de patentes de domínio público para todas as etapas do modelo TAL. Além disso, também não

foi possível identificar alguma proposta que mensure um processo de desenvolvimento de *software*, como o *Scrum*. Por fim, não foram encontradas métricas para coletar o *feedback* do cliente ou do mercado após o processo de *Release* do *software*. Tal questão leva a uma lacuna que sinaliza a importância de um processo de medição que contemple todas as etapas do modelo TAL e que possa suportar com métricas um processo de melhoria contínua para o *software* em desenvolvimento e/ou para as mudanças organizacionais, estruturais e na liderança através da organização que conduz o projeto.

5 CONCLUSÃO

Esta pesquisa teve como objetivo pesquisar nas bases de patentes de domínio público WIPO, EPO e LENS as possíveis patentes com propostas de métricas ágeis em gestão ágil de projetos. Na seleção inicial, foram encontradas 32 patentes para leitura e análise. Diante dos filtros aplicados, somente três patentes apresentaram métricas que abordassem, de maneira geral, o apoio ao desenvolvimento de *software* e aderência ao modelo TAL. Dentre as três patentes, somente duas apresentaram métricas ágeis que envolvessem as etapas de Fluxo de Valor e as etapas de *Delivery* e *Release*, sendo assim aderentes em partes ao modelo TAL. A terceira patente apresentou somente métricas de alocação de pessoas em atividades em um projeto ágil.

Não foram identificadas patentes que tratassem de métricas nas etapas de *Initiate* e *Discovery* conforme o TAL e que fossem específicas para obter *feedback* sobre o produto de *software* de valor e para validação de hipóteses do produto, clientes e mercado. Além disso, não foram identificadas as patentes que apresentassem métricas ágeis que refletissem o conhecimento obtido a partir do *feedback* do incremento (*release*) de *software* e de clientes (Sutherland, 2019), na validação de hipóteses (Gothelf, 2022; Ries, 2011) para confirmar ou refutar o valor de determinados requisitos para refletir diretamente no *Backlog* do Produto (Sutherland, 2019; Gothelf, 2022; Ries, 2011).

Também não foram encontradas métricas que sinalizassem mudanças necessárias nos times operacionais do projeto (Sutherland, 2019), na estrutura organizacional como os entregáveis não entregues por impedimentos não resolvidos (Kristensen & Shafiee, 2019), no portfólio de projeto como os itens do *backlog* do portfólio não entregues no trimestre corrente (PMI, 2017), e estratégia organizacional e de produto (Kristensen & Shafiee, 2019; Gothelf, 2022; Ries, 2011). Isso remete à necessidade de um processo de melhoria contínua que seja suportado por métricas de *feedback* de processo de desenvolvimento e de produto. Nesse

sentido que o processo de melhoria proverá as melhorias embasadas em métricas em todas as etapas do TAL que podem auxiliar a organização no planejamento de mudanças nas estratégias organizacionais, no portfólio de projetos, nos programas e times de projeto. Tais mudanças podem ajudar a organização no desenvolvimento de software que entregue o valor esperado por clientes e *stakeholders*. Por fim, não foram encontradas propostas de um processo de medição contínuo e incremental para a melhoria contínua do processo de desenvolvimento de *software* nas organizações nas etapas de *Initiate, Discovery, Delivery* e *Release* para entrega mais assertiva de valor aos clientes e para melhoria na estrutura organizacional, gestão do portfólio e estratégia organizacional. Este estudo possui limitações que tange à pesquisa de processos de medição em gerenciamento de projetos ágeis que podem ser uma base sólida para a identificação das métricas mais indicadas para cada processo em um ambiente de agilidade de negócios de uma organização. Assim, a partir desse prisma, o processo de medição em nível organizacional de projetos pode ajudar gestores e líderes no alinhamento das estratégias organizacionais, em como trazer uma nova visão sobre como medir o valor entregue, além de abrir uma lacuna para estudos futuros.

REFERÊNCIAS BIBLIOGRÁFICAS

ADKINS, Lyssa. Coaching agile teams: a companion for ScrumMasters, agile coaches, and project managers in transition. Pearson Education India, 2010.

Agile Manifesto (2001). Manifesto for Agile Software Development. Disponível em <<http://agilemanifesto.org/>> Acesso em: 26 jun. 2023.

BUDACU, Eduard Nicolae; POCATILU, Paul. Real Time Agile Metrics for Measuring Team Performance. *Informatica economica*, v. 22, n. 4, 2018.

CRESWELL, John W. Research: Qualitative, quantitative, and mixed methods approaches. California. EUA: Sage, 2003.

DA SILVA, Luciano Ferreira; RUSSO, Rosária de Fátima Segger Macri; DE OLIVEIRA, Paulo Sergio Gonçalves. Quantitativa ou qualitativa? um alinhamento entre pesquisa, pesquisador e achados em pesquisas sociais. *Revista Pretexto*, p. 30- 45, 2018.

DOS SANTOS, Paulo Sérgio Medeiros et al. On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *Journal of Software Engineering Research and Development*, v. 6, n. 1, p. 1-29, 2018.

GIL, Antonio Carlos; DOS REIS NETO, Aline Crespo. Survey de experiência como pesquisa qualitativa básica em administração. *Ciencias da Administração*, v. 22, n. 56, p. 125-137, 2020.

GOTHELF, Jeff. Lean UX: Applying lean principles to improve user experience. " O'Reilly Media, Inc.", 2013.

JYOTHI, V. Esther; RAO, K. Nageswara. Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean

Kanban. *International Journal of Advanced Engineering Research and Science*, v. 4, n. 3, 2017.

KRISTENSEN, Saeedeh Shafiee; SHAFIEE, Sara. Rethinking organization design to enforce organizational agility. In: *11th Symposium on Competence-Based Strategic Management*. 2019.

MISRA, Sanjay; OMORODION, Martha. Survey on agile metrics and their inter-relationship with other traditional development metrics. *ACM SIGSOFT Software Engineering Notes*, v. 36, n. 6, p. 1-3, 2011.

OZA, Nilay; KORKALA, Mikko. Lessons learned in implementing agile software development metrics. 2012.

PARAGANO, Carmine. Agile Contracting, come gestire un contratto. *PROJECT MANAGER (IL)*, n. 2021/46, 2021.

PARANHOS, Rita de Cassia Santos; RIBEIRO, Núbia Moura. Importância da prospecção tecnológica em base em patentes e seus objetivos da busca. *Cadernos de Prospecção*, v. 11, n. 5, p. 1274, 2018.

PENHA, Renato; DA SILVA, Luciano Ferreira; RUSSO, Rosária de Fátima Segger Macri. Escalando as práticas ágeis. *Revista de Gestão e Projetos*, v. 11, n. 2, p. 1- 11, 2020.

PMI. (2017). *A Guide to the project management body of knowledge (6th edição)*. Disponível em <<https://doi.org/10.5860/CHOICE.34-1636>>. Acessado em: 25 de Jun. 2023.

PRESSMAN, Roger S.; MAXIM, Bruce R. *Engenharia de software-9*. McGraw Hill Brasil, 2021.

RIES, Eric. *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. Currency, 2011.

RUBIN, Kenneth S. *Essential Scrum: A practical guide to the most popular Agile process*. Addison-Wesley, 2012.

Sommerville, Ian. *Engenharia de Software, 9. Edição*. Pearson, Addison Wesley, 8(9), 10, p. 44-45, 2007.

SUTHERLAND, Jeff. *SCRUM: A arte de fazer o dobro de trabalho na metade do tempo*. Leya, 2014.

TOH, M. Zulfahmi; SAHIBUDDIN, Shamsul; MAHRIN, Mohd Naz'ri. Adoption issues in DevOps from the perspective of continuous delivery pipeline. In: *Proceedings of the 2019 8th International Conference on Software and Computer Applications*. 2019. p. 173-177.

WIPO (2023). *International Patent Classification (IPC)*. Disponível em: <<https://www.wipo.int/classifications/ipc/en/>>. Acessado em: 25 de Jun. 2023.

“O conteúdo expresso no trabalho é de inteira responsabilidade dos autores”.

11 ANEXO I

11.1 TABELA ADICIONAL DO ESTUDO 2

Com objetivo de compreender as métricas em projetos tradicionais e em projetos ágeis, a Tabela 19 apresenta o relacionamento das métricas ágeis e tradicionais encontradas nas quatro categorias identificadas pelo Estudo 2, separadas pelas etapas de *Discovery* e *Delivery*, objetivo de cada métrica e seus respectivos autores.

Tabela 19- Relacionamento das Métricas encontradas no Estudo 2, separadas pelas etapas de *Discovery* e *Delivery*, objetivo de cada métrica e autores

Métricas	Métricas Ágeis de <i>Backlog</i> do Produto		Métricas Ágeis de Produto		Métricas Ágeis de Fluxo		Métricas de Projetos Tradicionais	
	<i>Discovery</i>		<i>Delivery</i>				<i>Discovery e Delivery</i>	
<i>EVM (Earned Value Management)</i>	Não Encontrada		Não Encontrada		Não Encontrada		Mede o desempenho e o progresso de projetos, com a capacidade de combinar medidas de escopo, tempo e custos, mediante um planejamento previamente estabelecido.	Vanhoucke (2011); Colin e Vanhoucke (2015) ; Kerkhove e Vanhoucke (2017); Chen <i>et al.</i> (2020); Chen, Chen e Lin (2016); Sharma, Caldas e Mulva (2021); Wauters e Vanhoucke (2015); Grau e Back (2015); Batselier e Vanhoucke

								(2015); Chen, 2018; Narbaev e De Marco (2014)
EDM (Earned Duration Management)	Não Encontrada		Não Encontrada		Não Encontrada		Investiga as variações do desempenho do cronograma	Yousefi <i>et al.</i> (2019); Orgut <i>et al.</i> , 2018); Ibrahim, Hanna e Kievet (2020)
Proporção de tarefas detalhadas para marcos	Não Encontrada		Não Encontrada		Não Encontrada		Investiga a proporção de tarefas do projeto que foram detalhada para marcos específicos.	Han <i>et al.</i> (2017)
Porcentagem de tarefas no caminho crítico	Não Encontrada		Não Encontrada		Não Encontrada		Calcula o percentual de tarefas do projeto no caminho crítico do cronograma.	Han <i>et al.</i> (2017)
Número de defasagens	Não Encontrada		Não Encontrada		Não Encontrada		Calcula o número de defasagens (atrasos) no cronograma.	Han <i>et al.</i> (2017)
Tipo de relacionamento (terminar para começar)	Não Encontrada		Não Encontrada		Não Encontrada		Calcula o tipo de relacionamento de tarefas predecessoras.	Han <i>et al.</i> (2017)
Marcos sem predecessor ou	Não Encontrada		Não Encontrada		Não Encontrada		Investiga os marcos no	Han <i>et al.</i> (2017)

sucessor							cronograma sem definição de predecessor e sucessor.	
Variação de Desempenho	Não Encontrada		Não Encontrada		Não Encontrada		Classificação de problemas de projetos para identificar problemas nos projetos que provavelmente falharão antes de começarem	Chen (2021) e Hazır (2015) ;
Work Perform	Não Encontrada		Não Encontrada		Não Encontrada		Combinou séries temporais e análise de regressão para desenvolver PV (<i>planned value</i>) em um modelo de resposta EV (<i>earned value</i>) para prever <i>work perform</i>	Chen (2014)
Índice de Controle de Custos	Não Encontrada		Não Encontrada		Não Encontrada		Indicador que mede os custos do projeto.	Lopez-Paredes (2011); Ibrahim, Hanna e Kievet (2020)
Índice de Controle de Cronograma	Não Encontrada		Não Encontrada		Não Encontrada		Indicador que mede o andamento do cronograma do projeto.	Lopez-Paredes (2011); Ibrahim, Hanna e Kievet (2020)
Análise do	Não Encontrada		Não Encontrada		Não Encontrada		Conceito de	Sato e Hirao

equilíbrio custo/risco							orçamento ótimo no nível do projeto e no nível da rede de atividades.	(2013)
Impacto Predecessor no Sucessor	Não Encontrada		Não Encontrada		Não Encontrada		Relação linear entre atividades em que as saídas de um precedente se tornam diretamente os insumos de um sucessor.	Kim, Kim e Cho (2016)
Critical Success Factors	Não Encontrada		Não Encontrada		Não Encontrada		Medir, monitorar e controlar fatores críticos de sucesso por meio de escala de metas, custos, infraestrutura de TI e gestão de projetos.	Aljedaibi e Khamis (2019)
Project Health Index	Não Encontrada		Não Encontrada		Não Encontrada		Medir o sucesso, em contraste com a visão tradicional de tempo e custo como único critério, onde o resultado da variável pode ser usado como monitor de progresso com base no	Rajagopalan e Srivastava (2018)

							<i>Corrective And Preventive Actions</i>	
Tempo de processamento de pedido de mudança	Não Encontrada		Não Encontrada		Não Encontrada		Tempo de processamento de pedido de mudança no projeto.	Ibrahim, Hanna e Kievet (2020)
Mudança percentual relacionada ao design	Não Encontrada		Não Encontrada		Não Encontrada		Mudança percentual relacionada ao design.	Ibrahim, Hanna e Kievet (2020)
Mudança percentual relacionada à qualidade/valor	Não Encontrada		Não Encontrada		Não Encontrada		Mudança percentual relacionada à qualidade/valor.	Ibrahim, Hanna e Kievet (2020)
Qualidade geral dos sistemas do projeto	Não Encontrada		Não Encontrada		Não Encontrada		Qualidade geral dos sistemas do projeto.	Ibrahim, Hanna e Kievet (2020)
Análise de Risco do Cronograma com restrição de recursos	Não Encontrada		Não Encontrada		Não Encontrada		Visa melhorar as ações corretivas e o desempenho do projeto sob o uso de recursos restritos	Song, Martens e Vanhoucke (2021)
Desempenho de cada atividade	Não Encontrada		Não Encontrada		Não Encontrada		Desempenho de cada atividade, medindo se a atividade foi entregue no prazo ou atrasada.	Pozzana <i>et al.</i> (2021)

Stakeholder Metrics-integrated Management Model	Não Encontrada		Não Encontrada		Não Encontrada		Integra as partes interessadas que permite gerenciar elas identificando, comunicando, engajando, capacitando e atendendo com métricas.	Rajablu <i>et al.</i> (2017)
Análise bootstrap	Não Encontrada		Não Encontrada		Não Encontrada		Análise de inicialização para avaliar o desempenho de um projeto de infraestrutura.	Hanna, Tadt e Whited (2012)
Modelo RE4SA	Não Encontrada		Não Encontrada		Não Encontrada		Modelo <i>Requirements Engineering for Software Architecture</i> que fornece uma conexão entre os artefatos e que facilita a comunicação dentro da equipe de desenvolvimento.	Spijkman <i>et al.</i> (2021)
Esforço da tarefa e monitoramento	Não Encontrada		Não Encontrada		Não Encontrada		Esforço no nível da tarefa e monitoramento do progresso.	Jethani (2013)

Monitoramento de marcos e cronogramas	Não Encontrada		Não Encontrada		Não Encontrada		Monitora o andamento de entregáveis (marcos) e o monitoramento de cronogramas.	Jethani (2013)
Monitoramento de defeitos	Não Encontrada		Não Encontrada		Não Encontrada		Monitora os defeitos identificados durante o desenvolvimento do projeto.	Jethani (2013)
Integridade de requisitos	Não Encontrada		Não Encontrada		Não Encontrada		<i>Avalia se os requisitos especificados estão completos ou não para o desenvolvimento.</i>	Iqbal, Naeem e Khan (2012)
Seleção de Projetos	Não Encontrada		Não Encontrada		Não Encontrada		Integraram o aprendizado organizacional e o desenvolvimento do conhecimento com um processo de seleção de projetos.	Geng <i>et al.</i> (2018) ; Ko & Cho (2020)
Elegibilidade do incremento	Não Encontrada		Não Encontrada		Avalia o impacto das práticas de gerenciamento de projetos nos dois primeiros <i>Sprints</i> .	Parrend, <i>et al.</i> (2014)	Não Encontrada	

Lead Time	Não Encontrada		Não Encontrada		Mede o tempo para realizar a entrega de uma funcionalidade desde o comprometimento até a entrega ao cliente.	Dennehy e Conboy (2018); Mas, Mesquida e Pacheco (2020)	Não Encontrada	
Cycle Time	Não Encontrada		Não Encontrada		Mede o tempo para realizar um item do <i>backlog</i> em etapas dentro da iteração.	Dennehy e Conboy (2018); Mas, Mesquida e Pacheco (2020)	Não Encontrada	
Velocity	Não Encontrada		Não Encontrada		Mede quantos pontos de história foram feitos por <i>Sprint</i>	Dennehy e Conboy (2018); Ahmed <i>et al.</i> (2017)	Não Encontrada	
Efficiency flow	Não Encontrada		Não Encontrada		Mede quantos itens do <i>backlog</i> são entregues por ciclo.	Mas, Mesquida e Pacheco (2020)	Não Encontrada	
Reopens	Não Encontrada		Não Encontrada		Mede o número de tarefas ou incidentes reabertos.	Mas, Mesquida e Pacheco (2020)	Não Encontrada	
Tasks finished flow	Não Encontrada		Não Encontrada		Mede as tarefas finalizadas dentro do fluxo de desenvolvimento	Mas, Mesquida e Pacheco (2020)	Não Encontrada	
Eficácia e eficiência	Não Encontrada		Não Encontrada		Eficácia e eficiência de entregas do	Grimaldi <i>et al.</i> (2016)	Não Encontrada	

					projeto durante a rodada da <i>Sprint</i>			
Custo hora real e planejada	Não Encontrada		Não Encontrada		Custo hora real e planejada de entregas do projeto durante a rodada da <i>Sprint</i>	Grimaldi <i>et al.</i> (2016)	Não Encontrada	
Impedimentos	Não Encontrada		Não Encontrada		Impedimentos mapeados durante a rodada da <i>Sprint</i>	Grimaldi <i>et al.</i> (2016)	Não Encontrada	
Custo do atraso	Não Encontrada		Não Encontrada		Custo de atraso de entregas do projeto durante a rodada da <i>Sprint</i>	Grimaldi <i>et al.</i> (2016)	Não Encontrada	
<i>Sprint Burn Down</i>	Não Encontrada		Não Encontrada		Métrica sobre pontos de história no <i>Sprint</i>	Ahmed <i>et al.</i> (2017)	Não Encontrada	
<i>Epic Burn Down</i>	Não Encontrada		Não Encontrada		Métrica sobre pontos de história no Épico	Ahmed <i>et al.</i> (2017)	Não Encontrada	
<i>Control Chart</i>	Não Encontrada		Não Encontrada		Mede o <i>Cycle Time</i> (ou <i>Lead Time</i>) na <i>Sprint</i>	Ahmed <i>et al.</i> (2017)	Não Encontrada	
Atribuição de bugs	Não Encontrada		Não Encontrada		Métrica sobre atribuição da pessoa mais indicada para a resolução do	Sajedi-Badashian e Stroulia (2020)	Não Encontrada	

					problema e reduzindo o <i>time to market</i> do produto.			
Reduzir Lead Time	Não Encontrada		Não Encontrada		Reduz o <i>lead time</i> para obter alta capacidade de resposta às necessidades dos clientes	Petersen e Wohlin (2011)	Não Encontrada	
Status Desenvolvimento	Não Encontrada		Não Encontrada		Fornecer um sistema de rastreamento que mostre o status do desenvolvimento de produtos de software.	Petersen e Wohlin (2011)	Não Encontrada	
Estimativa de tarefas	Não Encontrada		Não Encontrada		Estimativa de tarefas para ajudar a garantir a qualidade e estabilidade do software	Choraś <i>et al.</i> (2020)	Não Encontrada	
Rastreamento de tarefas	Não Encontrada		Não Encontrada		Rastreamento de tarefas para ajudar a garantir a qualidade e estabilidade do software	Choraś <i>et al.</i> (2020)	Não Encontrada	
Capacidade do processo	Não Encontrada		Não Encontrada		Medir a capacidade do processo ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	

Desempenho do processo	Não Encontrada		Não Encontrada		Medir o desempenho do processo ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	
Maturidade do Processo	Não Encontrada		Não Encontrada		Medir a maturidade do processo ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	
Esforço no projeto	Não Encontrada		Não Encontrada		Medir o esforço do projeto ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	
Desempenho no projeto	Não Encontrada		Não Encontrada		Medir o desempenho do projeto ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	
Risco no projeto	Não Encontrada		Não Encontrada		Medir os riscos do projeto ágil.	Meidan <i>et al.</i> (2018)	Não Encontrada	
Clareza do Artefato	Não Encontrada		Não Encontrada		Medir a clareza do artefato produzido pelo projeto.	Sadler (2020)	Não Encontrada	
Envolvimento e Propriedade do Cliente	Não Encontrada		Não Encontrada		Medir o envolvimento e propriedade do artefato produzido pelo projeto.	Sadler (2020)	Não Encontrada	
Aplicabilidade e Rastreabilidade	Não Encontrada		Não Encontrada		Medir a aplicabilidade e rastreabilidade do artefato produzido pelo projeto.	Sadler (2020)	Não Encontrada	

Componentes funcionais	Não Encontrada		Não Encontrada		Cobertura dos cenários e histórias de usuário que auxiliaram no desempenho do processo de teste.	Asha e Mani (2018)	Não Encontrada	
Product Backlog Rating	Mede o nível de prioridade do item do <i>backlog</i>	Kayes, Sarker e Chakareski (2016)	Não Encontrada		Não Encontrada		Não Encontrada	
KPI de Imperativos da solução técnica proposta	KPI para auxiliar no planejamento de recursos, testar o desempenho da agilidade, qualidade e valor dos negócios, eficiência da equipe e complexidade de projetos.	Zheng <i>et al.</i> (2021)	Não Encontrada		Não Encontrada		Não Encontrada	
Grau de singularidade do projeto	Características da entrega futura como o custo, prazo, possíveis desvios de prazo e custo.	Pavlova <i>et al.</i> (2021)	Não Encontrada		Não Encontrada		Não Encontrada	

Grau de complexidade do projeto	Número de variantes de soluções tecnológicas, o número de operações para cada variante e o número de alternativas para execução da operação para cada variante da tecnologia do produto do projeto.	Pavlova <i>et al.</i> (2021)	Não Encontrada		Não Encontrada		Não Encontrada	
Esforço de desenvolvimento de software	Avaliação da aplicabilidade ao longo do ciclo de vida, dependência de uma linguagem de programação específica, capacidade de prescrever soluções e a validade da medida.	Wallace e Sheetz (2014)	Não Encontrada		Não Encontrada		Não Encontrada	
Esforço e produtividade	Apoiar a estimativa de esforço inicial baseada no tamanho do Use Case Point (UCP)	Azzeh e Nassif (2016)	Não Encontrada		Não Encontrada		Não Encontrada	

Avaliação da qualidade das histórias de usuários (HU).	Determina a qualidade da redação das HUs, a definição de prioridade e o risco das HUs, os relacionamentos de HU com fontes de requisitos, requisitos não-funcionais e versões de produtos.	Prakash e Viswanathan (2018)	Não Encontrada		Não Encontrada		Não Encontrada	
Quantidade de commits	Não Encontrada		Quantidade de novos códigos-fonte publicados.	Mäkiäho, Vartiainen e Poranen (2022)	Não Encontrada		Não Encontrada	
Horas trabalhadas	Não Encontrada		Quantidade de horas trabalhadas durante um período.	Mäkiäho, Vartiainen e Poranen (2022)	Não Encontrada		Não Encontrada	
Tempo nos estágios de desenvolvimento	Não Encontrada		Tempo do requisito nos estágios de desenvolvimento.	Mäkiäho, Vartiainen e Poranen (2022)	Não Encontrada		Não Encontrada	

Características dos riscos de software	Não Encontrada		Fornecer informações sobre modelos de risco que podem ser usados para entender as características dos riscos de <i>software</i> .	Chang (2015)	Não Encontrada		Não Encontrada	
Crescimento nas vendas de novos produtos	Não Encontrada		Medir o aumento nas vendas de novos produtos	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Orçamento dedicado à análise ou verificação do cliente	Não Encontrada		Medir os orçamentos com dedicação de análise e verificação do cliente.	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Projetos orientados para o cliente	Não Encontrada		Medir o número de projetos criados e orientados para o cliente.	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Projetos/ programas respeitando custos e orçamento	Não Encontrada		Medir o número de projetos e programas respeitando custos e orçamento.	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Marcos respeitados	Não Encontrada		Medir o número de marcos respeitados e entregues.	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	

Componentes reutilizáveis	Não Encontrada		Mede o número de componentes de software reutilizáveis dentro ou entre projetos.	Mascarenhas <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Chidamber-Kemerer	Não Encontrada		Análises de complexidade de programas orientados a objetos em Java.	Singh <i>et al.</i> (2019); Isong e Obeten (2013); Concas <i>et al.</i> (2012)	Não Encontrada		Não Encontrada	
Mean Time to Execution	Não Encontrada		Mede o desempenho do código com o tempo médio de execução	Singh <i>et al.</i> (2019)	Não Encontrada		Não Encontrada	
Source Lines of Code	Não Encontrada		mede o número de linhas do código-fonte do programa.	Isong e Obeten (2013); Ibban <i>et al.</i> (2018); Concas <i>et al.</i> (2012)	Não Encontrada		Não Encontrada	
Coupling between Objects	Não Encontrada		Representa o número de classes acopladas a uma determinada classe.	Isong e Obeten (2013); Concas <i>et al.</i> (2012)	Não Encontrada		Não Encontrada	
Response for a Class	Não Encontrada		Número total de métodos que potencialmente podem ser executados em resposta a uma mensagem	Isong e Obeten (2013); Concas <i>et al.</i> (2012)	Não Encontrada		Não Encontrada	

			recebida por um objeto de uma classe.					
Weighted Methods per Class	Não Encontrada		Mede a soma da complexidade dos métodos em uma classe.	Isong e Obeten (2013); Concas <i>et al.</i> (2012)	Não Encontrada		Não Encontrada	
Bugs resolvidos por unidade de tempo	Não Encontrada		Métrica que olha para a quantidade de bugs resolvidos por unidade de tempo.	Üsfekes <i>et al.</i> (2019); Aljemabi <i>et al.</i> (2020)	Não Encontrada		Não Encontrada	
Nível de Alinhamento da Funcionalidade	Não Encontrada		Mede o nível de alinhamento das funcionalidades com as estratégias organizacionais.	Costa, Vasconcelos e Fragoso, B. (2020)	Não Encontrada		Não Encontrada	
Nível de Confiabilidade	Não Encontrada		Mede o nível de confiabilidade em termos de qualidade de software.	Costa, Vasconcelos e Fragoso, B. (2020)	Não Encontrada		Não Encontrada	
Nível de Eficiência e Eficácia	Não Encontrada		Mede o nível de eficiência em termos de qualidade de software.	Costa, Vasconcelos e Fragoso, B. (2020)	Não Encontrada		Não Encontrada	
Nível de Eficácia	Não Encontrada		Mede o nível de eficácia em termos de qualidade de	Athanasiou <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	

			<i>software.</i>					
Nível de Integridade	Não Encontrada		Mede o nível de integridade em termos de qualidade de <i>software.</i>	Athanasiou <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Nível de Manutenibilidade	Não Encontrada		Mede o nível de manutenibilidade e em termos de qualidade de <i>software.</i>	Costa, Vasconcelos e Fragoso, B. (2020); Athanasiou <i>et al.</i> (2014)	Não Encontrada		Não Encontrada	
Nível de Portabilidade	Não Encontrada		Mede o nível de portabilidade do <i>software</i> com outras arquiteturas de <i>software.</i>	Costa, Vasconcelos e Fragoso, B. (2020)	Não Encontrada		Não Encontrada	
Teste em Canvas	Não Encontrada		Métrica de transformação <i>Lean</i> no plano de teste de <i>software</i> para simplificar o processo de teste.	Nidagundi e Novickis (2016)	Não Encontrada		Não Encontrada	
Net Promoter Score	Não Encontrada		Mede o nível de satisfação dos clientes em relação do produto recebido e sua experiência com ele.	Day <i>et al.</i> (2019); Simpson <i>et al.</i> (2021)	Não Encontrada		Não Encontrada	

Complexidade por acoplamento	Não Encontrada		Mede a complexidade por acoplamento que define o número de recursos de <i>software</i> por solução.	Shawky e Abd-El-Hafiz (2016)	Não Encontrada		Não Encontrada	
Complexidade por coesão	Não Encontrada		Mede a complexidade por coesão que define o número de coesão intra componentes de <i>software</i> .	Shawky e Abd-El-Hafiz (2016)	Não Encontrada		Não Encontrada	
Previsão por Planning Poker	Não Encontrada		Mede o nível de compreensão e assertividade da estimativa.	Tahir, Rasool e Noman (2018)	Não Encontrada		Não Encontrada	
Qualidade por Sprint	Não Encontrada		Mede o nível de qualidade do software por <i>Sprint</i> .	Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018)	Não Encontrada		Não Encontrada	
Qualidade por Release ou Incremento	Não Encontrada		Mede o nível de qualidade do software por <i>Release</i> ou incremento.	Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018); Perkusich <i>et al.</i> (2017).	Não Encontrada		Não Encontrada	

Qualidade por Solução	Não Encontrada		Mede o nível de qualidade do <i>software</i> por Solução.	Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018)	Não Encontrada		Não Encontrada	
Qualidade por Portfólio	Não Encontrada		Mede o nível de qualidade do <i>software</i> por Portfólio de projetos.	Pradhan e Nanniyur (2021)	Não Encontrada		Não Encontrada	
Qualidade por Visão Empresarial	Não Encontrada		Mede o nível de qualidade do <i>software</i> alinhado com a Visão Empresarial.	Pradhan e Nanniyur (2021); Kamulegeya, Mugwanya e Hebig (2018)	Não Encontrada		Não Encontrada	
Code smells	Não Encontrada		Mede potenciais problemas em código-fonte.	Yamashita e Counsell (2013)	Não Encontrada		Não Encontrada	
Usou Test-Driven Development	Não Encontrada		Mede se o projeto utilizou o <i>Test-Driven Development</i>	Bakhtiary, Gandomani e Salajegheh (2020)	Não Encontrada		Não Encontrada	
Não usou Test-Driven Development	Não Encontrada		Mede se o projeto não utilizou o <i>Test-Driven Development</i> .	Bakhtiary, Gandomani e Salajegheh (2020)	Não Encontrada		Não Encontrada	
Complexidade ciclomática	Não Encontrada		Mede a complexidade ciclomática (complexidade	Bakhtiary, Gandomani e Salajegheh (2020)	Não Encontrada		Não Encontrada	

			do código-fonte).					
Qualidade do código-fonte	Não Encontrada		Mede a qualidade do código-fonte.	Perkusich <i>et al.</i> (2017)	Não Encontrada		Não Encontrada	
Qualidade dos Eventos do Scrum	Não Encontrada		Mede a qualidade dos eventos do <i>Scrum</i> , como <i>Sprint Plannig</i> , <i>Daily</i> , <i>Sprint Review</i> e <i>Sprint Retrospective</i> .	Perkusich <i>et al.</i> (2017)	Não Encontrada		Não Encontrada	
Entrega antecipada e frequente de software	Não Encontrada		Mede a relevância da produtividade entregando <i>software</i> com antecedência e frequência.	Hernández, <i>et al.</i> (2020)	Não Encontrada		Não Encontrada	
Agregação de valor	Não Encontrada		Mede o retorno do investimento como uma forma relevante de medir a agregação de valor	Hernández, <i>et al.</i> (2020)	Não Encontrada		Não Encontrada	
Aspectos relacionados ao esforço	Não Encontrada		Mede a relevância o esforço de implantação de <i>software</i> que resolva os problemas de	Hernández, <i>et al.</i> (2020)	Não Encontrada		Não Encontrada	

			clientes.					
Modelo de previsão de defeitos	Não Encontrada		Modelo que mede a previsão de defeitos de <i>software</i> com base em outras métricas.	Gao <i>et al.</i> (2011)	Não Encontrada		Não Encontrada	
Segurança orientada a riscos	Não Encontrada		Métrica de segurança hierárquicas orientadas a risco no contexto do desenvolvimento ágil de <i>software</i> .	Savola, Frühwirth e Pietikäinen (2012)	Não Encontrada		Não Encontrada	
Melhoria de processo de <i>software</i>	Não Encontrada		Utilizou métricas de produto para avaliar melhorias no processo de desenvolvimento de <i>software</i> .	Tahir, Rasool e Noman (2018)	Não Encontrada		Não Encontrada	

Fonte: elaborado pelo autor, 2023 (Estudo 2).

REFERÊNCIAS

- Adkins, L. (2020). *Coaching agile teams: a companion for ScrumMasters, agile coaches, and project managers in transition*. Pearson Education India.
- Agile Manifesto (2001). Manifesto for Agile Software Development. Disponível em <<http://agilemanifesto.org/>> Acesso em: 07 fev. 2021.
- Agile Manifesto (2001). Manifesto for Agile Software Development. Disponível em <<http://agilemanifesto.org/>> Acesso em: 07 fev. 2021.
- Al-Zewairi, M., Biltawi, M., Etaawi, W., e Shaout, A. (2017). Agile software development methodologies: survey of surveys. *Journal of Computer and Communications*, 5(05),74.
- Anwer, F., Aftab, S., Shah, S. M., e Waheed, U. (2017). Comparative analysis of two popular agile process models: Extreme Programming and Scrum. *International Journal of Computer Science and Telecommunications*, 8(2), 1-7.
- Azanha Neto, J. D. S., Penha, R., da Silva, L. F., & Scafuto, I. C. (2022). The importance of leadership in agile projects: systematic literature review. *Research, Society and Development*, 11(5), e44511528117-e44511528117.
- CAPES. (2019). *Produção Técnica – Grupo de Trabalho (Relatório de Grupo de Trabalho)*, Brasília, DF, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Ministério da Educação. Recuperado em 20/11/2023 de <https://www.gov.br/capes/pt->.
- Caroli, P. (2017). *Lean Inception*. São Paulo, Brasil: Editora Caroli.org.
- Child, J., e Mansfield, R. (1972). Technology, size, and organization structure. *Sociology*, 6(3), 369-393.
- Child, J. (1973). Predicting and understanding organization structure. *Administrative science quarterly*, 168-185.
- Dorninger, B., & Ziebermayr, T. (2021). Software-Qualitätssicherung im Maschinen-und Anlagenbau: automatisierte Bewertung der technischen Qualität von SPS-Code. e & i *Elektrotechnik und Informationstechnik*, 138(6), 315-320.
- Bakhtiary, V. *et al.* (2020). The effectiveness of test-driven development approach on software projects: A multi-case study. *Bulletin of Electrical Eng and Info.*,9(5),2030-2037.
- Beck, Kent; Beedle, Mike; Bennekum, Arie Van; Cockburn, Alistair; Cunningham, Ward; Fowler, Martin; Grenning, James; Highsmith, Jim; Hunt, Andrew; Jeffries, Ron; Kern, Jon; Marick, Brian; Martin, Robert. C.; Mellor, Steve; Schwaber, Ken; Sutherland, Jeff; Thomas, Dave. Manifesto for agile software development. Available: <http://www.agilemanifesto.org>. [04 Out. 2022].
- Budacu, E. N., e Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, 22(4), 70-79.
- Daft, R. L. (2015). *Organization theory and design*. Cengage learning.
- Dos Santos, P. S. M., Beltrão, A. C., de Souza, B. P., e Travassos, G. H. (2018). On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *Journal of Software Engineering Research and Development*, 6(1), 1-29.
- Galbraith, J. R. (2012). The future of organization design. *Journal of Organization Design*, 1(1).

- Genari, J. O. S., e Ferrari, F. C. (2016). Times de alto desempenho no contexto das metodologias Scrum e Kanban. *Revista TIS*, 4(3).
- Gokarna, M., & Singh, R. (2021, February). DevOps: a historical review and future works. *In 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 366-371). IEEE.
- Huijgens, H., Lamping, R., Stevens, D., Rothengatter, H., Gousios, G., & Romano, D. (2017, August). Strong agile metrics: mining log data to determine predictive power of software metrics for continuous delivery teams. *In Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering* (pp. 866-871).
- Hron, M., e Obwegeser, N. (2018, January). Scrum in practice: an overview of Scrum adaptations. *In Proceedings of the 51st Hawaii International Conference on System Sciences*.
- Isniah, S., Purba, H. H., & Debora, F. (2020). Plan do check action (PDCA) method: literature review and research issues. *Jurnal Sistem dan Manajemen Industri*, 4(1), 72-81.
- Junior, V. B., e Junior, S. D. (2019). Elaboração de um Plano de Gerenciamento de Projeto para Implantação de Laboratório de Metrologia Utilizando o Guia PMBOK® 5º Edição. *Cippus*, 7(2), 127-145.
- Jyothi, V. E., e Rao, K. N. (2017). Effective Implementation of Agile Software Development with a Framework, Metric Tool, and in Association with Cloud and Lean Kanban. *International Journal of Advanced Engineering Research and Science*, 4(3), 237085.
- Kim, G., Humble, J., Debois, P. & Willis, J. (2018). Manual de DevOps: Como obter Agilidade, Confiabilidade e Segurança em Organizações Tecnológicas. Rio de Janeiro: Alta Books Editora, 2018 (pp. 3).
- Kristensen, S. S., & Shafiee, S. (2019, September). Rethinking organization design to enforce organizational agility. *In 11th Symposium on Competence-Based Strategic Management*, Stuttgart (pp. 1-13).
- Misra, S., & Omorodion, M. (2011). Survey on agile metrics and their inter-relationship with other traditional development metrics. *ACM SIGSOFT Software Engineering Notes*, 36(6), 1-3.
- Miterev, M., Mancini, M., e Turner, R. (2017). Towards a design for the project-based organization. *International Journal of Project Management*, 35(3), 479-491.
- Moodley, D., Sutherland, M. & Pretorius, P. (2016) Comparing the power and influence of functional managers with that of project managers in matrix organizations: The challenge in duality of command, *South African Journal of Economic & Management Sciences*, vol. 19, no. 1, pp. 103-117, 2016.
- Mosser, S., Pulgar, C., & Reinharz, V. (2022). Modelling Agile Backlogs as Composable Artifacts to support Developers and Product Owners. *Journal of Object Technology (JOT)*.
- Moult, D., & Krijnen, T. (2020). Compliance checking on building models with the Gherkin language and Continuous Integration. *In EG-ICE 2020 Workshop on Intelligent Computing in Engineering, Proceedings* (pp. 294-303).

- Nawaz, Z., Aftab, S., e Anwer, F. (2017). Simplified FDD Process Model. *International Journal of Modern Education e Computer Science*, 9(9).
- Nugroho, F. A., e Sugiarto, T. (2017). Extreme Programming Methodology Approach to Pamulang University. In *Prosiding Seminar Nasional Informatika* Vol. 2549, p. 4805.
- Oza, N., & Korkala, M. (2012). Lessons learned in implementing agile software development metrics.
- Paragano, C. (2021). Agile contracting, come gestire un contratto. Agile contracting, come gestire un contratto, 15-18.
- Penha, R., da Silva, L. F., e Russo, R. D. F. S. M. (2020). Escalando as práticas ágeis. *Revista de Gestão e Projetos*, 11(2), 1-11.
- Pinochet, L. H. C., Pachelli, I. L., & Da Rocha, F. M. M. (2018). Uso de Métricas em Mídias Sociais e Indicadores de Desempenho do Site e sua Relação com o Valor da Marca em Empresas de Cosméticos no Brasil. *Revista Brasileira de Marketing*, 17(1), 80-99.
- PMI. (2017). A Guide to the project management body of knowledge (6th edição). Disponível em <<https://doi.org/10.5860/CHOICE.34-1636>>.
- Rachman, A., & Ratnayake, R. C. (2018). Adoption and implementation potential of the lean concept in the petroleum industry: state-of-the-art. *International journal of lean six sigma*.
- Ries, E. (2011). *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. New York: Crown Business.
- Rubin, K. S. (2017). *Essential Scrum: A practical guide to the most popular Agile process*. Addison-Wesley.
- Saltz, J., e Sutherland, A. (2020, January). SKI: A new agile framework that supports DevOps, continuous delivery, and lean hypothesis testing. In *Proceedings of the 53rd Hawaii International Conference on System Sciences*.
- Schwaber, K., e Sutherland, J. (2020). *Scrum Guide*. Disponível em <<https://www.scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>> Acesso em: 07 fev. 2021.
- Sommerville, I. (2007). *Engenharia de Software*, 9. Edição. Pearson, Addison Wesley, 8(9), 10, p. 44-45.
- Stray, V., Moe, N. B., Vedal, H., & Berntzen, M. (2023). Using objectives and key results (OKRs) and slack: a case study of coordination in large-scale distributed agile.
- Sutherland, J. (2019). *Scrum: A arte de fazer o dobro do trabalho na metade do tempo*. Rio de Janeiro: Sextante, 2019.
- Süß, J. G., Swift, S., & Escott, E. (2022). Using DevOps toolchains in Agile model-driven engineering. *Software and Systems Modeling*, 21(4), 1495-1510.
- Thiry, M., & Deguire, M. (2007). Recent developments in project-based organizations. *International journal of project management*, 25(7), 649-658.
- Toh, M. Z., Sahibuddin, S., & Mahrin, M. N. R. (2019). Adoption issues in DevOps from the perspective of continuous delivery pipeline. In *Proceedings of the 2019 8th International Conference on Software and Computer Applications* (pp. 173-177).

- Van Casteren, W. (2017). The Waterfall Model and the Agile Methodologies: A comparison by project characteristics. *Research Gate*, 1-6.
- Vedal, H., Stray, V., Berntzen, M., & Moe, N. B. (2021). Managing dependencies in large-scale agile. In *Agile Processes in Software Engineering and Extreme Programming—Workshops: XP 2021 Workshops, Virtual Event, June 14–18, 2021, Revised Selected Papers 22* (pp. 52-61). *Springer International Publishing*.
- Zafar, I., Nazir, A. K., e Abbas, M. (2017). The impact of agile methodology (DSDM) on software project management. In *Circulation in Computer Science: International Conference on Engineering, Computing and Information Technology (ICECIT 2017)* (pp. 1-6).