



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS REITOR JOÃO DAVID FERREIRA LIMA
PROGRAMA DE GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Monique Bertan

UM ESTUDO DE CASO DE SELEÇÃO E APLICAÇÃO DE MÉTRICAS ÁGEIS

Florianópolis, Santa Catarina – Brasil
2022

Monique Bertan

UM ESTUDO DE CASO DE SELEÇÃO E APLICAÇÃO DE MÉTRICAS ÁGEIS

Trabalho de Conclusão de Curso submetido ao Programa de Graduação em Ciências da Computação da Universidade Federal de Santa Catarina para a obtenção do Grau de Bacharel em Ciências da Computação.

Orientador: Prof. Jean Carlo Rossa Hauck, Dr.

Florianópolis, Santa Catarina – Brasil

2022

Notas legais:

Não há garantia para qualquer parte do software documentado. Os autores tomaram cuidado na preparação desta tese, mas não fazem nenhuma garantia expressa ou implícita de qualquer tipo e não assumem qualquer responsabilidade por erros ou omissões. Não se assume qualquer responsabilidade por danos incidentais ou consequentes em conexão ou decorrentes do uso das informações ou programas aqui contidos.

Catálogo na fonte pela Biblioteca Universitária da Universidade Federal de Santa Catarina.
Arquivo compilado às 18:30h do dia 23 de dezembro de 2022.

Monique Bertan

Um estudo de caso de seleção e aplicação de métricas ágeis / Monique Bertan; Orientador, Prof. Jean Carlo Rossa Hauck, Dr. - Florianópolis, Santa Catarina - Brasil, 28 de novembro de 2020.

84 p.

Trabalho de Conclusão de Curso - Universidade Federal de Santa Catarina, INE - Departamento de Informática e Estatística, CTC - Centro Tecnológico, Programa de Graduação em Ciências da Computação.

Inclui referências

1. Gerenciamento de projetos, 2. Metodologias ágeis, 3. Métricas ágeis, I. Prof. Jean Carlo Rossa Hauck, Dr. II. Programa de Graduação em Ciências da Computação III. Um estudo de caso de seleção e aplicação de métricas ágeis

CDU 02:141:005.7

Monique Bertan

UM ESTUDO DE CASO DE SELEÇÃO E APLICAÇÃO DE MÉTRICAS ÁGEIS

Este(a) Trabalho de Conclusão de Curso foi julgado adequado(a) para obtenção do Título de Bacharel em Ciências da Computação, e foi aprovado em sua forma final pelo Programa de Graduação em Ciências da Computação do INE – Departamento de Informática e Estatística, CTC – Centro Tecnológico da Universidade Federal de Santa Catarina.

Florianópolis, Santa Catarina – Brasil, 28 de novembro de 2020.

Prof. Jean Everson Martina, Dr.
Coordenador(a) do Programa de
Graduação em Ciências da Computação

Banca Examinadora:

Prof. Jean Carlo Rossa Hauck, Dr.
Orientador
Universidade Federal de Santa
Catarina – UFSC

Prof. Raul Sidnei Wazlawick, Dr.
Universidade Federal de Santa Catarina –
UFSC

Gabriel Holdener Geraldeli, Bac.
Universidade Federal de Santa Catarina –
UFSC

*Este trabalho é dedicado ao meus pais,
que ao longo de toda a jornada foram meu apoio e incentivo diário.*

AGRADECIMENTOS

Primeiramente gostaria de agradecer meu orientador, prof, Jean, que por longos meses teve a paciência e sabedoria para me guiar para o melhor trabalho possível, agradeço também às equipes envolvidas no estudo de caso, em especial aos meu amigos e colegas de trabalho Nathan Werlich, Rodrigo Costa e Everton Coelho que me ajudaram a engajar as equipes durante as dinâmicas e reuniões.

Também agradeço ao meu namorado, Gustavo, que esteve presente ao meu lado desde o primeiro dia de aula até hoje, me confortando em seu abraço durante momentos difíceis e comemorando comigo todas as conquistas, seu apoio e incentivo foram fundamentais durante todos estes anos.

Aos meus queridos amigos que conheci durante a graduação, em especial Caio, Salomão, Letícia, Fernanda e Rodrigo, durante todos estes anos morando longe da família vocês se tornaram como uma família para mim, responsáveis pela companhia do almoço, boas risadas e inúmeros conselhos.

Por fim, gostaria de agradecer aos meus pais, Natal e Marildes, e ao meu irmão Mateus, meus maiores apoiadores. A dedicação de vocês me fizeram chegar até aqui e esta conquista é nossa, amo vocês, obrigada por tudo.

“Learn from yesterday, live for today, hope for tomorrow. The important thing is not to stop questioning.”
Albert Einstein

RESUMO

Com o crescimento do mercado de software e o aumento da complexidade dos softwares produzidos pela indústria nos últimos anos, mais organizações têm adotado o uso de metodologias ágeis como alternativa à gestão de projetos tradicionais. Neste contexto, métricas tradicionais são adaptadas para o controle e monitoramento de projetos, produto e organizações e então passam a ser chamadas de métricas ágeis. As métricas ágeis são utilizadas no gerenciamento dos projetos, permitindo monitorar diversos aspectos do projeto, como a produtividade da equipe e qualidade do produto em diferentes estágios do desenvolvimento. Assim, este trabalho busca avaliar a seleção e aplicação de métricas ágeis em um ambiente organizacional por meio de um estudo de caso aplicado na unidade organizacional eSUS-APS do laboratório Bridge da Universidade Federal de Santa Catarina. O estudo de caso é sistematicamente planejado e aplicado. Durante a aplicação foi aplicada uma dinâmica de seleção de métricas e o acompanhamento das métricas selecionadas por duas sprints, de duas semanas cada, em duas equipes, totalizando 13 pessoas participantes. Os dados coletados no estudo de caso levantam indícios de que métricas ágeis podem ser aplicadas para o acompanhamento de planos de ação da ferramenta roda ágil.

Palavras-chaves: Gerenciamento de projetos. Metodologias ágeis. Métricas ágeis.

ABSTRACT

The growing software market and the increasing complexity of software, produced by the industry in recent years, more organizations have adopted the use of agile methodologies as an alternative to traditional project management. In this context, traditional metrics are adapted for the control and monitoring of projects, products and organizations and then come to be called agile metrics. Agile metrics are used in project management, allowing you to monitor various aspects of the project, such as team productivity and product quality at different stages of development. Thus, this work seeks to evaluate the selection and application of agile metrics in an organizational environment through a case study applied in the organizational unit eSUS-APS of the Bridge laboratory at the Federal University of Santa Catarina. The case study is systematically planned and applied. During the application, a dynamic of selecting metrics and monitoring the selected metrics was applied for two sprints, of two weeks each, in two teams, totaling 13 people participating. The data collected in the case study raise evidence that agile metrics can be applied to monitor action plans of the wheel agile tool.

Keywords: Project management. Agile methodologies. Agile metrics.

LISTA DE FIGURAS

Figura 1	–	Relacionamentos do modelo de informação	19
Figura 2	–	Detalhamento dos relacionamentos do modelo de informação . .	22
Figura 3	–	Detalhamento do processo de medição	23
Figura 4	–	Quadro Kambam	29
Figura 5	–	Ciclo de release no XP	30
Figura 6	–	LeSS	33
Figura 7	–	LeSS Huger	33
Figura 8	–	Essential SAFe	34
Figura 9	–	Estado Da Arte	39
Figura 10	–	Estado Da Arte	40
Figura 11	–	e-SUS APS	47
Figura 12	–	Paradigma GQM	52
Figura 13	–	Etapas do estudo de caso	54
Figura 14	–	Painel Padlet	56
Figura 15	–	CFD - Sprint 1	63
Figura 16	–	CFD - Sprint 2	63
Figura 17	–	Burnup - Sprint 1	64
Figura 18	–	Burnup - Sprint 2	64
Figura 19	–	Pergunta 1	66
Figura 20	–	Pergunta 2	67

LISTA DE TABELAS

Tabela 1	–	String de Busca	37
Tabela 2	–	Membros da equipe F	48
Tabela 3	–	Membros da equipe S	48
Tabela 4	–	Objetivo de Medição 1	52
Tabela 5	–	Objetivo de Medição 2	53
Tabela 6	–	Objetivo de Medição 3	53
Tabela 7	–	Procedimento de coleta de medidas	53
Tabela 8	–	Planos de ação equipe F	56
Tabela 9	–	Categorias selecionas equipe F	57
Tabela 10	–	Descrição da métrica número de defeitos	58
Tabela 11	–	Descrição da métrica taxa de conclusão de histórias planejadas .	58
Tabela 12	–	Descrição da métrica comparação de fluxo de tarefas concluídas	58
Tabela 13	–	Planos de ação equipe S	59
Tabela 14	–	Categorias selecionas equipe S	59
Tabela 15	–	Descrição da métrica diagrama de fluxo acumulado	60
Tabela 16	–	descrição da métrica burnup	61
Tabela 17	–	Número de defeitos - Equipe F	61
Tabela 18	–	Taxa de conclusão de histórias planejadas- Equipe F	62
Tabela 19	–	Comparação de fluxo de tarefas concluídas - Equipe F	62

LISTA DE ABREVIATURAS E SIGLAS

XP	Extreme Programming
SM	Scrum Master
PO	Product Owner
LeSS	Large Scale Scrum
SAFe	Scaled Agile Framework
RSL	Revisão Sistemática da Literatura
GQM	Goal-Question-Metric
ART	Agile Release Train

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVOS	16
1.1.1	Objetivo geral	16
1.1.2	Objetivos específicos	16
1.2	MÉTODO DE PESQUISA	16
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	MEDIÇÃO	18
2.1.1	Modelo de informação	19
2.1.1.1	Entidade	20
2.1.1.2	Atributo	20
2.1.1.3	Método de medição	20
2.1.1.4	Medida base	20
2.1.1.5	Medida derivada	20
2.1.1.6	Indicador	21
2.1.1.7	Conceito mensurável	21
2.1.2	Processo de medição	21
2.1.2.1	Planejar a medição	22
2.1.2.2	Realizar a medição	23
2.1.2.3	Avaliar a medição	23
2.1.2.4	Estabelecer e sustentar o compromisso	23
2.2	MÉTODOS ÁGEIS	24
2.2.1	Scrum	25
2.2.1.1	Papéis	26
2.2.1.2	Eventos	26
2.2.1.3	Artefatos	27
2.2.2	Kamban	29
2.2.3	XP	29
2.2.4	Lean	30
2.2.5	Scaled Agile	31
2.2.5.1	LeSS	32
2.2.5.2	SAFe	33
3	ESTADO DA ARTE	36
3.1	QUESTÃO DE PESQUISA	36
3.2	TERMOS DE BUSCA	37
3.3	FONTE DE DADOS	37
3.4	STRINGS DE BUSCA E CRITÉRIOS	37

3.5	CRITÉRIOS DE INCLUSÃO E EXCLUSÃO	37
3.6	SELEÇÃO DOS ESTUDOS	38
3.7	ANÁLISE	38
3.7.1	Q1: Qual o contexto de uso das métricas?	41
3.7.2	Q2: Qual o perfil das organizações de software que utilizam? .	41
3.7.3	Q3: Quais são as métricas utilizadas?	41
3.7.4	Q4: Como as métricas foram selecionadas?	42
3.7.5	Q5: Como as métricas são classificadas?	43
3.7.6	Q6 - Qual o impacto do uso de métricas	43
3.7.7	Ameaças à validade	46
4	PROCESSO ATUAL	47
4.1	CONTEXTO	47
4.1.1	Produto	47
4.1.2	Equipes	48
4.1.3	Processo de desenvolvimento	48
4.1.4	Roda ágil	49
5	PLANEJAMENTO DO ESTUDO DE CASO	51
5.1	OBJETIVO DO ESTUDO DE CASO	51
5.2	PERGUNTAS E MEDIDAS	52
5.3	PLANEJAMENTO DA COLETA	53
6	EXECUÇÃO DO ESTUDO DE CASO	54
6.1	SELEÇÃO DE MÉTRICAS ÁGEIS	55
6.1.1	Equipe F	55
6.1.2	Equipe S	59
6.2	APLICAÇÃO DAS MÉTRICAS SELECIONADAS	61
6.2.1	Equipe F	61
6.2.2	Equipe S	62
6.3	COLETA E ANÁLISE DOS RESULTADOS	65
6.4	DISCUSSÃO	67
6.5	AMEAÇAS A VALIDADE	67
7	CONCLUSÃO	69
	REFERÊNCIAS	70
	APÊNDICE A – ARTIGO	74

1 INTRODUÇÃO

Visando a melhoria em seus processos de desenvolvimento, um grande número de organizações e equipes têm adotado metodologias ágeis na última década. Segundo o 14º estudo anual sobre o estado ágil ([VERSIONONE, 2020](#)), dentre os motivos que atraíram empresas a adotar métodos ágeis estão o aumento de produtividade e previsibilidade de entrega ([VERSIONONE, 2020](#)).

Métodos ágeis de desenvolvimento de software são um conjunto de práticas de gestão baseadas nos quatro pilares e doze princípios do manifesto ágil ([BECK *et al.*, 2001](#)), pensados para responder às mudanças de forma ágil e eficiente ([COHEN; LINDVALL; COSTA, 2004](#)). Os métodos ágeis mais utilizados atualmente são o Scrum e métodos híbridos que o incluem, como o Scrumban, sendo adotados em cerca de 75% das organizações ([VERSIONONE, 2020](#)).

O manifesto ágil, documento que formaliza princípios e valores dos métodos ágeis, define no 12º princípio que em um intervalo definido de tempo os times devem avaliar como se tornar mais eficientes e então adequar seu comportamento de modo a atingir seus objetivos. Nesse cenário de melhoria contínua, as métricas tornam-se fundamentais para o acompanhamento e aprimoramento da maturidade da equipe ágil, sendo fundamentais para diagnosticar pontos de melhoria no processo e monitorar seu desenvolvimento ao longo do tempo ([BECK *et al.*, 2001](#)).

Dada a importância da medição para o monitoramento e controle dos projetos de software, algumas iniciativas têm buscado selecionar e adaptar métricas tradicionais para o contexto dos métodos ágeis, como por exemplo [Padmini, Bandara e Perera \(2015\)](#), [Alperowitz, Dzvonyar e Bruegge \(2016\)](#), [Boerman *et al.* \(2015\)](#) e [Cardozo *et al.* \(2020\)](#).

Assim, a partir da problemática apresentada, este trabalho se propõe a realizar um estudo de caso no Laboratório Bridge, integrado ao Centro Tecnológico (CTC) da Universidade Federal de Santa Catarina (UFSC) que atua na pesquisa e desenvolvimento de soluções tecnológicas para o governo federal ([BRIDGE, 2016](#)), com o objetivo de avaliar o impacto do uso de métricas ágeis no dia-a-dia das equipes ágeis do laboratório.

O estudo de caso utilizará de métricas propostas encontradas na literatura, durante a revisão sistemática da literatura, que deu origem a uma lista de métricas descritas no artigo "How Agile Organizations Use Metrics: A Systematic Literature Mapping" ([LEAL; VIEIRA; BERTAN, 2022](#)) em equipes com diferentes perfis do Laboratório Bridge, durante o período letivo do ano de 2022.

Para o desenvolvimento deste estudo de caso será realizado um estudo inicial sobre o processo de desenvolvimento ágil do laboratório, duas equipes diferentes serão selecionadas e com base nas métricas encontradas na literatura um conjunto de

métricas serão selecionadas para o acompanhamento dos planos de ação definidos por cada equipe durante a aplicação da roda ágil. A cada período de três meses, as equipes devem realizar os planos de ação anteriormente definidos e durante esse período as métricas selecionadas serão acompanhadas, afim de medir a evolução de cada plano de ação. Ao final do trimestre será avaliado se houve a percepção de melhora na visibilidade do andamento dos planos de ação pela equipe, e o impacto do uso de métricas no dia-a-dia do time. Desta forma serão definidas as métricas mais adequadas para o acompanhamento da evolução de cada equipe.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Este trabalho tem como objetivo geral avaliar a seleção e aplicação de métricas ágeis em um ambiente organizacional de desenvolvimento de software por meio de um estudo de caso aplicado na unidade organizacional eSUS-APS do laboratório Bridge da Universidade Federal de Santa Catarina.

1.1.2 Objetivos específicos

- Realizar um diagnóstico inicial de alinhamento da unidade organizacional, buscando determinar quais abordagens ágeis são utilizadas pela organização.
- Analisar as métricas ágeis utilizadas pelas equipes da organização.
- Selecionar métricas ágeis adequadas à realidade das equipes da organização.
- Aplicar as métricas selecionadas e acompanhar as equipes durante um período de tempo.
- Avaliar a aplicação de métricas ágeis na organização através da aplicação de um estudo de caso.

1.2 MÉTODO DE PESQUISA

Este trabalho se propõe a aplicar os conhecimentos adquiridos, através de um embasamento teórico, em uma unidade organizacional e analisar os impactos da aplicação. A metodologia será dividida em quatro etapas, sendo elas:

Etapla 1. Análise do Estado da Arte: Por meio de um mapeamento sistemático de literatura conceitos fundamentais ao trabalho serão analisados.

Atividade 1.1: Análise da literatura sobre conceitos de engenharia de software relevantes ao trabalho.

Atividade 1.2: Análise da literatura sobre conceitos de métodos ágeis.

Atividade 1.3: Análise da literatura sobre conceitos de métricas ágeis.

Etapla 2. Diagnóstico do estado atual da organização: Com base nos conceitos estudados na etapa anterior, nesta etapa o estado atual da organização será descrito e oportunidades de melhoria serão levantadas.

Atividade 2.1: Definição do processo atual do laboratório.

Atividade 2.2: Comparação do processo atual do laboratório com a literatura.

Etapa 3. Planejamento do estudo de caso: A partir das oportunidades de melhoria levantadas no tópico anterior serão definidas metas e objetivos que serão medidos através do estudo de caso.

Atividade 3.1: Definição do objetivo do estudo de caso.

Atividade 3.2: Planejamento do estudo de caso.

Etapa 4. Aplicação e avaliação do estudo de caso: O estudo de caso será executado e os dados resultantes serão analisados.

Atividade 4.1: Aplicação do estudo de caso.

Atividade 4.2: Análise dos resultados do estudo de caso.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados conceitos fundamentais para a compreensão do trabalho. Serão apresentados os conceitos de medição e métodos ágeis.

2.1 MEDIÇÃO

Medição é o processo de atribuir números ou símbolos a entidades do mundo real, de forma a tornar possível a caracterização de cada entidade por meio de regras claramente definidas (SOFTEX, 2016). De forma simplificada, Wazlawick (2019) define *medição* como sendo o processo de obtenção de medidas.

O propósito do processo de medição é coletar, armazenar, analisar e relatar os dados relativos a produtos desenvolvidos e processos implementados na organização e em seus projetos, de forma a apoiar os objetivos organizacionais, com foco sanar necessidades informacionais das partes interessadas e apoiar a tomada de decisão em relação ao produto, processo ou gestão de uma organização (SOFTEX, 2016).

A forma como a medição é implementada em uma organização pode definir seu sucesso e desempenho em termos de negócio e performance organizacional. Em geral, organizações que implementam planos de medição baseados em objetivos técnicos e organizacionais, visando fornecer informações para a tomada de decisões, mitigações de riscos e ajustes de processo tendem a obter mais sucesso. A medição é considerada significativa e cumpre seu papel quando definida a partir de necessidades já existente na organização e torna-se parte integral do processo (MCGARRY *et al.*, 2001).

Como resultado da implementação de um processo de medição bem-sucedida algumas mudanças poderão ser observadas (ISO/IEC/IEEE..., 2017):

- O estabelecimento do compromisso de medição;
- Necessidades informacionais de processos de gestão são identificadas;
- Um conjunto apropriado de medidas é identificado ou desenvolvido;
- Atividades de medição são identificadas;
- Atividades de medição são planejadas;
- Os dados considerados necessários são coletados, armazenados, analisados e os resultados interpretados;
- As informações resultantes são utilizadas para apoiar decisões;
- O processo de medição e as medidas são avaliados;

- Sugestões de melhoria no processo de medição são comunicadas ao responsável pelo processo.

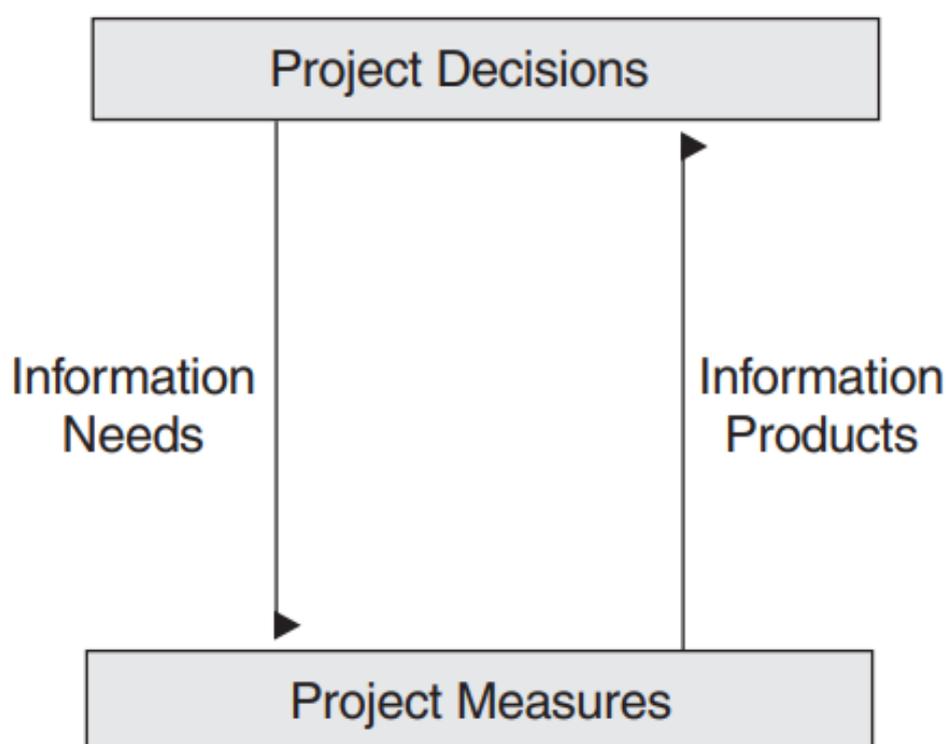
Um programa de medição é formado por dois componentes, um modelo informacional, responsável por identificar e definir, de forma clara, todos os componentes que são envolvidos no processo de medição (SOFTEX, 2016), e também pelo processo de medição, que por sua vez, é responsável por definir a estrutura de coleta, análise e reporte dos dados definidos no modelo de informação. (MCGARRY *et al.*, 2001)

2.1.1 Modelo de informação

McGarry *et al.* (2001) descreve o modelo de medição como sendo um mecanismo para vincular as necessidades de informação dos produtos e projetos de software as entidades que podem ser medidas, estabelecendo um modelo definido de conceito de medição e fornecendo uma comunicação eficaz dos resultados para a organização.

Considerado um recurso primário do processo de medição, o modelo de informação fornece uma estrutura que define as medições essenciais para fornecer informação que apoiarão decisões de projeto.

Figura 1 – Relacionamentos do modelo de informação



Fonte: (MCGARRY *et al.*, 2001)

O modelo descreve como os atributos relevantes são quantificados e convertidos

em indicadores que forneçam uma base para a tomada de decisão (ISO/IEC/IEEE..., 2017).

2.1.1.1 Entidade

Entidade é um objeto que é caracterizado por possuir atributos que podem ser medidos, uma entidade pode ter uma ou mais propriedades que são interessantes para atender necessidades de informação. Processos, Produtos, projetos e recursos são exemplos de entidades (ISO/IEC/IEEE..., 2017).

2.1.1.2 Atributo

É a propriedade ou característica da entidade que pode ser coletada qualitativamente ou quantitativamente, de forma automática ou manual. Cada entidade pode ter diversos atributos, porém somente alguns deles são relevantes para serem medidos. O primeiro passo para a definição de um modelo de informação é a seleção de atributos relevantes para suprir a necessidade de informação existente. Um atributo pode ser utilizado em diversos contextos de apoio a tomada de decisões organizacionais (ISO/IEC/IEEE..., 2017).

2.1.1.3 Método de medição

Podemos descrever um método de medição como uma sequência lógica de operações, atributos são quantificados em relação a uma escala especificada. A sequência de operações pode envolver a contagem de ocorrências ou observação de passagem de tempo. O mesmo método de medição pode ser aplicado em diferentes atributos, porém cada combinação de método de medição e atributos produz uma medida base diferente. Um procedimento de medição descreve o método de medição implementado em um determinado contexto organizacional (ISO/IEC/IEEE..., 2017).

2.1.1.4 Medida base

Medida é uma variável a qual um valor é atribuído. Medida base é uma medida independente de outras medidas, ela captura informações sobre um único atributo. A definição de um intervalo de valores esperados ajuda a verificar a qualidade do dado coletado (ISO/IEC/IEEE..., 2017).

2.1.1.5 Medida derivada

Medida derivada é o resultado da manipulação de duas ou mais medidas base. A medida derivada usualmente captura informações de diferentes atributos ou o mesmo atributo em diferentes entidades. A normalização de dados utiliza da conversão de medidas base para medidas derivadas, assim diferentes entidades podem ser

comparadas entre si (ISO/IEC/IEEE..., 2017).

2.1.1.6 Indicador

Indicador é uma medida que fornece uma estimativa ou avaliação de um atributo específico, derivado de um modelo e com o objetivo de suprir as necessidades de informação previamente estabelecidas. Indicadores são a base para a análise e tomada de decisões, sendo eles o principal resultado apresentado aos interessados (ISO/IEC/IEEE..., 2017).

2.1.1.7 Conceito mensurável

Um conceito mensurável é a relação abstrata entre atributos de entidades e a necessidade informacional. Um exemplo de necessidade informacional é a necessidade de comparar a produtividade de desenvolvimento de um grupo de projetos versus a taxa alvo, neste caso o conceito mensurável é a taxa de produtividade de desenvolvimento. Outros exemplos de conceitos mensuráveis incluem qualidade, risco, performance, maturidade e valor para o usuário (ISO/IEC/IEEE..., 2017).

2.1.2 Processo de medição

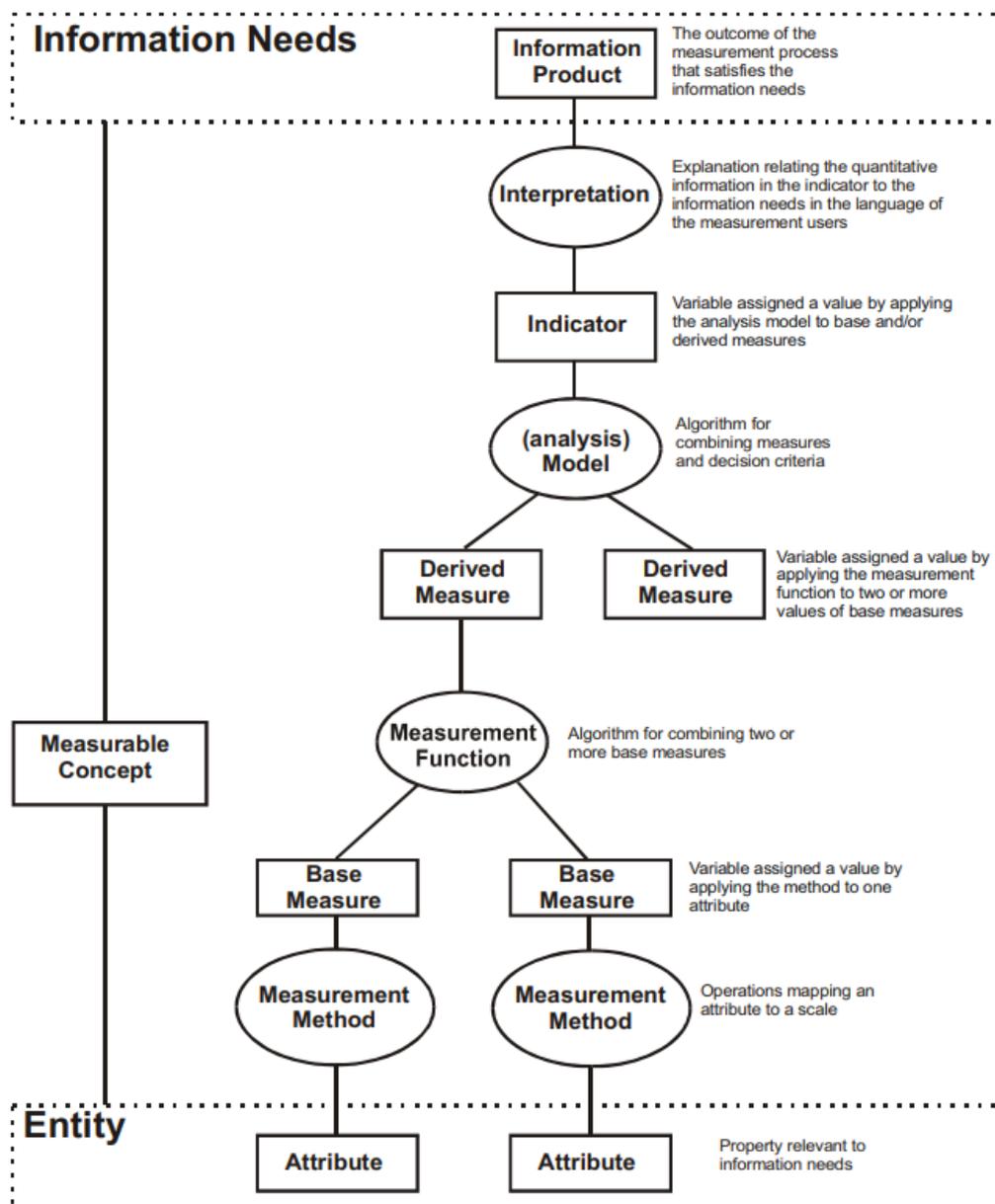
O propósito do processo de medição é coletar, analisar e reportar dados relativos a produtos desenvolvidos e processos implementados em uma unidade organizacional, para apoiar o gerenciamento de processos e demonstrar de forma objetiva a qualidade dos produtos (ISO/IEC/IEEE..., 2017).

O processo de medição complementa o modelo de informação, utilizados em conjunto os modelos fornecem um framework para a implementação de medição em um projeto. Em conjunto, os modelos definem um programa de medição único, desenvolvido apropriadamente para o contexto em que será aplicado (MCGARRY *et al.*, 2001).

O modelo é construído baseando-se na abordagem "Planejamento, Execução, Verificação e Ação" e adaptado para apoiar atividades específicas de medição, 4 atividades principais compõem o processo de medição, todas essenciais para a implementação bem-sucedida de medição (MCGARRY *et al.*, 2001).

- Planejar a medição
- Realizar a medição
- Avaliar a medição
- Estabelecer e sustentar o compromisso

Figura 2 – Detalhamento dos relacionamentos do modelo de informação



Fonte: (ISO/IEC/IEEE..., 2017)

2.1.2.1 Planejar a medição

A atividade de planejamento de medição abrange a identificação de necessidade informacional existente no projeto e a seleção de medidas apropriadas para suprir a necessidade informacional utilizando o modelo de informação. As atividades de planejamento de medição fornecem a integração de medidas a técnicas e gerenciamento de projetos já existentes, além de definir os recursos e tecnologias necessários para a implementação de um projeto de medição (MCGARRY *et al.*, 2001).

2.1.2.2 Realizar a medição

Juntamente com a atividade de planejar a medição, realizar a medição é considerada uma atividade central do processo de medição. Responsável por coletar e processar os dados da medição, relacionar o dado analisado com as necessidades informacionais da organização, gerar informações sobre produtos e apresentar os resultados de análises para proporcionar recomendações as partes interessadas (MCGARRY *et al.*, 2001).

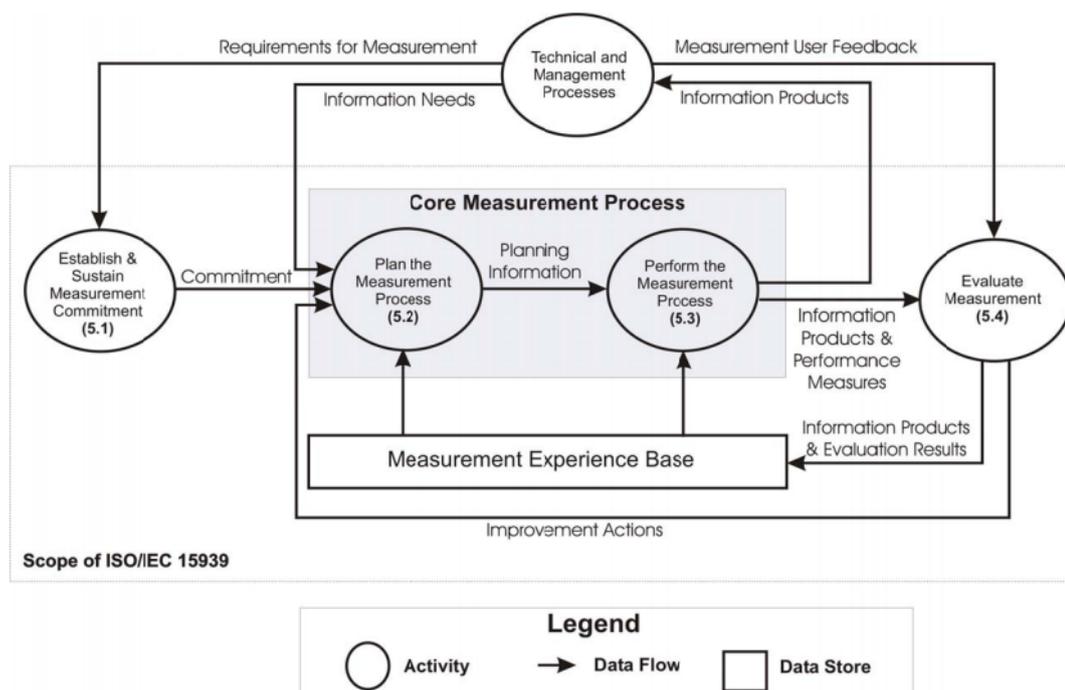
2.1.2.3 Avaliar a medição

Avaliar a medição consiste em aplicar técnicas de medição e análise ao próprio processo de medição. A atividade avalia as medidas aplicadas e o processo de medição adotado afim de identificar possíveis ações de melhorias. Esta etapa do processo é essencial para assegurar que o processo de medição está sempre alinhado com as necessidade informacionais da organização (MCGARRY *et al.*, 2001).

2.1.2.4 Estabelecer e sustentar o compromisso

A atividade de estabelecer e sustentar o compromisso assegura que a organização e o projeto possuem os recursos necessários para implementar um programa de medição viável e com comprometimento de todos os níveis organizacionais envolvidos (MCGARRY *et al.*, 2001).

Figura 3 – Detalhamento do processo de medição



Fonte: (ISO/IEC/IEEE... , 2017)

2.2 MÉTODOS ÁGEIS

Os métodos ágeis surgiram em meados dos anos 90, como alternativa aos processos tradicionais de desenvolvimento de software, conhecidos também como processos "pesados", eles eram caracterizados por pesadas regulamentações, regimentações e micro-gerenciamento. Inicialmente os métodos ágeis eram conhecidos como processos "leves" e priorizavam as pessoas e os processos, tentando eliminar tudo o que era burocrático.

No ano 2001, 17 representantes dos frameworks que até então eram chamados de métodos leves se reuniram e criaram o manifesto ágil, comprometendo-se a seguir e disseminar o que haviam definido naquele documento, a partir de então o termo métodos ágeis passou a ser adotado (BECK *et al.*, 2001).

O manifesto ágil (BECK *et al.*, 2001) definiu quatro valores, sendo eles:

- Indivíduos e interações mais que processos e ferramentas
- Software em funcionamento mais que documentação abrangente
- Colaboração com o cliente mais que negociação de contratos
- Responder a mudanças mais que seguir um plano

Doze princípios também foram definidos:

- Nossa maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de software com valor agregado.
- Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente.
- Entregar frequentemente software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
- Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
- Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessário e confie neles para fazer o trabalho.
- O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.
- Software funcionando é a medida primária de progresso.

- Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
- Contínua atenção à excelência técnica e bom design aumenta a agilidade.
- Simplicidade—a arte de maximizar a quantidade de trabalho não realizado—é essencial.
- As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizáveis.
- Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

Atualmente, segundo o [VersionOne \(2020\)](#) a metodologia ágil mais utilizada no mundo é o Scrum, com cerca de 66% das organizações utilizando sua forma mais tradicional e 15% utilizando derivações do Scrum como ScrumBan e Scrum/XP híbridos. Kambam aparece na pesquisa com cerca de 4% de usuários e XP e Lean com 1% cada. A pesquisa também aponta que dentre as técnicas ágeis que mais são praticadas dentro das organizações estão as daily meetings, retrospectivas e sprint planning.

2.2.1 Scrum

Scrum é um framework para gerenciamento de projetos que tem como objetivo ajudar pessoas, organizações e times a gerar valor através de soluções adaptativas para problemas complexos ([SCRUM.OCR, 2020](#)), considerado um framework leve e fácil de entender. Três pilares sustentam o Scrum: Transparência, inspeção e adaptação.

Transparência - Indica que informações importantes relativas ao processo e ao produto devem estar visíveis para todos os interessados. O cliente deve poder acompanhar o estágio do desenvolvimento e desenvolvedores devem ser capazes de perceber a necessidade do cliente e a satisfação com relação ao produto que está em desenvolvimento ([WAZLAWICK, 2019](#)). A falta de transparência pode levar ao aumento de risco e decisões que diminuem o valor do produto, pois a inspeção sem transparência é enganosa e gera desperdício ([SCRUM.OCR, 2020](#)).

Inspeção - Tem por objetivo detectar variações ou problemas potencialmente indesejáveis, ([SCRUM.OCR, 2020](#)) sendo assim, sempre que possível o trabalho executado por uma pessoa deve ser inspecionado por outra. O objetivo não é fiscalizar ou punir, mas certificar que o trabalho foi executado de maneira apropriada ([WAZLAWICK, 2019](#)). A inspeção habilita adaptação, sem que a adaptação seja realizada a inspeção é considerada inútil, visto que os pontos de melhoria levantados não serão melhorados ([SCRUM.OCR, 2020](#)).

Adaptação - É necessária pois os requisitos inicialmente definidos se mantêm em constante mudança, sendo descobertos e atualizados mesmo ao longo do desenvolvimento (WAZLAWICK, 2019). Se o processo ou o produto desviar do resultado esperado, seu rumo deve ser ajustado o mais rápido possível, para assim minimizar o impacto e desvios futuros (SCRUM.OGR, 2020).

O Scrum também composto por papéis, artefatos, eventos e regras, que serão explicados abaixo (WAZLAWICK, 2019).

2.2.1.1 Papéis

O Time Scrum é composto de um Scrum Master, um Product Owner e desenvolvedores. São times multifuncionais e autogerenciáveis, normalmente composto por até 10 pessoas. O time é responsável por todas as atividades relacionadas ao produto necessárias para o seu desenvolvimento que gere um incremento valioso e útil a cada sprint (SCRUM.OGR, 2020).

Product Owner (PO) - É o responsável por definir os requisitos e prioridades do produto e representar o cliente dentro do time Scrum, defendendo seus interesses. Dentre suas atribuições podemos citar o desenvolvimento da meta do produto, criar, priorizar e manter o Backlog e garantir que ao final da sprint o produto desenvolvido está de acordo com a meta da sprint (SCRUM.OGR, 2020).

Scrum Master (SM) - É responsável por auxiliar na compreensão de todas as práticas, regras, reuniões, papéis, teorias, pilares e valores do Scrum dentro do time Scrum. Dentre suas atividades principais estão a remoção de impedimentos ao progresso do time Scrum, garantir que as reuniões sejam executadas, produtivas e se mantenham dentro do tempo estipulado. (SCRUM.OGR, 2020)

Desenvolvedores - São responsáveis por criar incrementos utilizáveis a cada sprint. Suas principais atividades são criar e seguir um plano para que a meta definida para a sprint seja alcançada, atingindo a qualidade e definição de pronto definidas antes do início da sprint (SCRUM.OGR, 2020).

2.2.1.2 Eventos

Sprint - É o nome atribuído a um período de tempo que varia entre 2 a 4 semanas e acontece de forma cíclica. Durante esse período é definido um objetivo, criado um incremento funcional no produto que cumpra o objetivo, inspecionado o incremento e inspecionado o processo para que, no próximo ciclo, haja uma adaptação que melhore o processo (SCRUM.OGR, 2020).

Uma sprint pode ser cancelada apenas se o objetivo definido no início se tornar obsoleto, o Product Owner é o único que tem autoridade para isso.

Sprint planning - O evento é realizado no início de cada sprint, o objetivo da sprint é definido, os elementos do Product Backlog são priorizados e transferidos para

o sprint Backlog, os elementos colocados no sprint Backlog correspondem a lista de funcionalidades a serem desenvolvidas durante a sprint. Nesse momento a equipe se compromete a desenvolver todos os itens até o fim da sprint e o Product Owner se compromete a não incluir mais itens até o final da sprint (WAZLAWICK, 2019). Para sprints de 4 semanas o tempo máximo definido é de 8 horas, para sprints mais curtas o evento é mais curto (SCRUM.ORG, 2020).

Daily scrum - É realizado diariamente, tem como objetivo alinhar as atividades realizadas por cada membro do time desde o último daily scrum, o que vai ser realizado até a próxima daily e os possíveis impedimentos existentes (WAZLAWICK, 2019). O evento conta com a participação de todos os desenvolvedores e se o SM e o PO estiverem trabalhando em itens do product Backlog eles também participam. O evento tem um tempo máximo de 15 minutos e geralmente é realizado em pé.

Sprint review - O propósito da sprint review é inspecionar o resultado do que foi desenvolvido durante a sprint e determinar possíveis adaptações. O time scrum apresenta o resultado da sprint para os interessados e o progresso em direção a meta final é discutido. Todo o time Scrum participa da reunião, o cliente também pode estar presente, caso seja do seu interesse. O tempo máximo definido para uma sprint de um mês é de quatro horas, podendo ser mais curto para sprints mais curtas (SCRUM.ORG, 2020).

Sprint retrospective - Assim como o Sprint review, o sprint retrospective também é uma cerimônia de inspeção, nela a sprint que está finalizando é revisada em relação a indivíduos, interações, processos, ferramentas e a definição de pronto. O time discute o que deu certo e o que pode ser melhorado. O evento marca o final da Sprint e tem um tempo máximo de três horas para a sprint de um mês, sprints mais curtas tem um tempo menor. todos os membros do time participam do evento (SCRUM.ORG, 2020).

2.2.1.3 Artefatos

Artefatos no Scrum são gerados a partir dos eventos de cada sprint e representam trabalho ou valor, são projetados para fornecer informações necessárias para todos os envolvidos no processo. Cada artefato de trabalho está relacionado com um artefato de valor (SCRUM.ORG, 2020):

- Através do Product Backlog pode-se visualizar o progresso para atingir a meta do produto;
- Através do Sprint Backlog pode-se visualizar o progresso para atingir a meta do sprint;
- Através do incremento pode-se visualizar o progresso para atingir a Definição de Pronto.

Product Backlog - É a lista de funcionalidade a serem implementadas em um projeto. Os itens do Product Backlog são ordenados pelo Product Owner por ordem de prioridade e durante a cerimônia de sprint planning são refinados em conjunto com o time, recebendo uma descrição clara para todos e uma estimativa de esforço, se o item for muito grande ele é quebrando em itens menores, mais precisos e novamente cada item é estimado (SCRUM.OGR, 2020). Itens mais acima da lista possuem um grau de certeza maior do que deve ser realizado, enquanto itens mais ao fim da lista possuem um grau maior de incertezas. A ordenação também acontece pelo grau de urgência ou ROI, os itens mais acima são desenvolvidos primeiro (WAZLAWICK, 2019). O product Backlog está em constante mudança, sendo adicionado a ele incrementos ao produto, melhoria e tarefas de manutenção, ele é a única fonte de trabalho para o time scrum.

Meta do produto - Descreve o estado futuro do produto a longo prazo para o time Scrum, a partir da meta serão derivadas tarefas que, quando cumpridas, resultarão no cumprimento da meta do produto, as tarefas são incluídas no product Backlog e o progresso do produto pode ser acompanhado por lá. Para que uma nova meta seja criada a antiga deve ter sido cumprida ou abandonada. (SCRUM.OGR, 2020)

Sprint Backlog - Assim como o Product Backlog, o Sprint Backlog é uma lista de funcionalidade a serem implementadas durante a sprint atual, ela é definida em conjunto com o time durante a reunião de sprint planning e reflete a meta da sprint. A lista é priorizada pelo Product Owner e os itens mais acima devem ser desenvolvidos primeiramente. Novos itens só podem ser incluídos ou removidos com a permissão do Product Owner (WAZLAWICK, 2019). O product Backlog é utilizado também para inspecionar o progresso durante a daily scrum (SCRUM.OGR, 2020).

Meta da sprint - É o objetivo da sprint. Criada durante o Sprint planning e inspecionado durante o Sprint review, espera-se que ao final da sprint a meta da sprint tenha sido alcançada, gerando valor ao produto (SCRUM.OGR, 2020).

Incremento - É o resultado de todo o trabalho realizado durante a sprint, um novo estado para o produto que a meta da sprint se propôs a trabalhar (WAZLAWICK, 2019). Ao final de cada sprint o incremento resultante é revisado e incorporado ao incremento já existente das sprints passadas. O incremento deve ser utilizável, a fim de fornecer valor. Para que o trabalho seja considerado incremento ele deve cumprir a definição de pronto. Ao final da Sprint o incremento é apresentado na reunião de sprint review para os interessados (SCRUM.OGR, 2020).

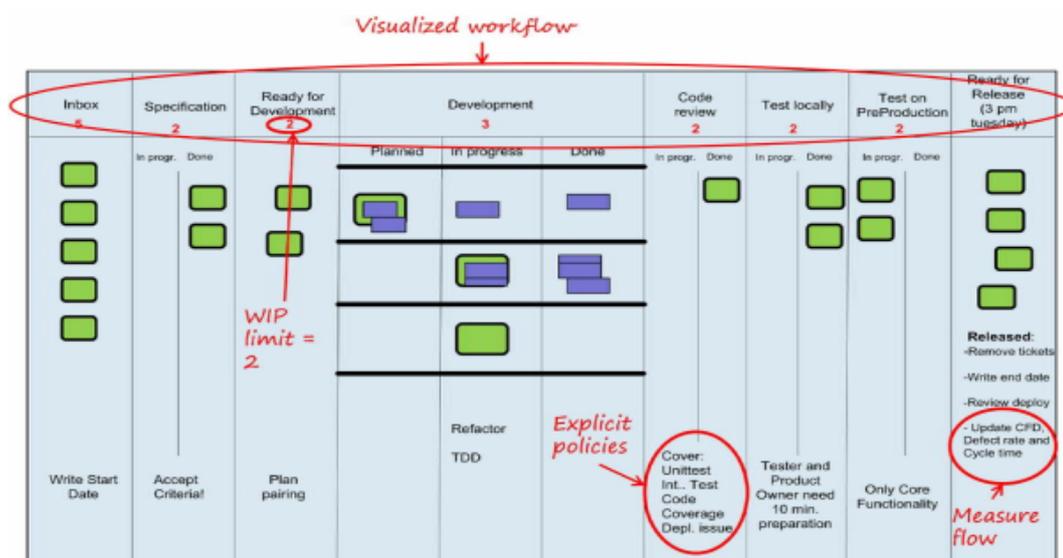
Definição de pronto - É uma descrição formal das medidas de qualidade que o incremento precisa atingir para que seja considerado pronto. A definição de pronto é importante pois gera transparência sobre os requisitos necessários ao final do desenvolvimento. Se um incremento não cumprir a definição de pronto pré-estabelecida ele não poderá ser liberado ou apresentado no Sprint review (SCRUM.OGR, 2020). Toda equipe deve estar ciente da definição de pronto adotada (WAZLAWICK, 2019).

2.2.2 Kamban

Segundo (ANDERSON; CARMICHAEL, 2016) kamban é um método para definir, gerenciar e melhorar produtos que dependem de conhecimento humano, como design, trabalhos criativos ou softwares. Baseado em uma implementação mais direta do Lean, com foco no fluxo e contexto, o kamban tem se tornado um complemento ao Scrum e XP (BOEG, 2010).

Composto de um quadro com colunas de estados e cartões com pequenas tarefas, o objetivo do quadro kamban é entender e focar nas necessidades e expectativas dos clientes; gerenciar o trabalho, dando abertura para que a equipe seja auto-gerenciável; desenvolver políticas para melhorar o resultado esperado pelo cliente e pela organização (ANDERSON; CARMICHAEL, 2016).

Figura 4 – Quadro Kamban



Fonte: (ANDERSON; CARMICHAEL, 2016)

2.2.3 XP

Os primeiros conceitos associados ao extreme programming (XP) são datados do final da década de 80, contudo o primeiro trabalho sobre o tema foi escrito por Kent Beck em 1999 (PRESSMAN; MAXIM, 2016). A abordagem foi proposta com o objetivo de impulsionar boas práticas já conhecidas, como o desenvolvimento iterativo (SOMMERVILLE, 2011).

Os princípios básicos do XP, segundo Wazlawick (2019), são:

- Feedback rápido.
- Presumir simplicidade.

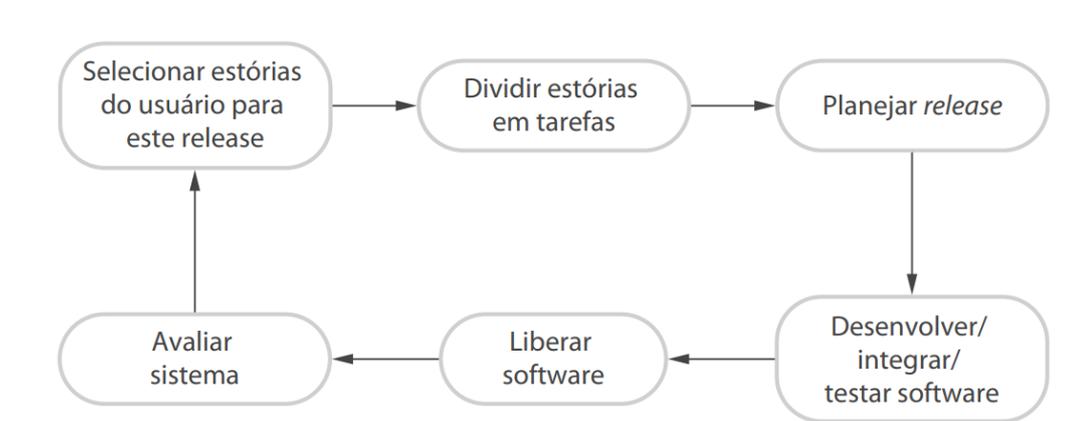
- Mudanças incrementais.
- Abraçar mudanças.
- Trabalho de alta qualidade

Mudanças são entendidas como algo positivo, que faz parte do processo, necessárias para que o desenvolvimento incremental e o rápido feedback aconteçam. A qualidade também é um ponto muito valorizado, considera-se que pequenos ganhos a curto prazo em detrimento de qualidade não compensam as perdas geradas a medio e longo prazo (WAZLAWICK, 2019).

No XP os requisitos são escritos em formato de histórias de usuário, cenários de uso que descrevem a necessidade do usuário e os requisitos necessários para que elas sejam alcançadas, que posteriormente serão decompostas em tarefas. Os programadores trabalham em pares e iniciam escrevendo testes para a tarefa a ser executada, posteriormente o código é escrito e ao integrar a tarefa todos os testes devem ser executados com sucesso (SOMMERVILLE, 2011).

Funcionalidades mais importantes são priorizadas e desenvolvidas antes das consideradas menos prioritárias, assim, se o trabalho não puder ser concluído integralmente, as partes mais importantes do projeto terão sido entregues, gerando mais valor ao cliente (WAZLAWICK, 2019).

Figura 5 – Ciclo de release no XP



Fonte: (SOMMERVILLE, 2011)

2.2.4 Lean

O Lean surgiu inicialmente durante o ano de 1940, criado por Taiichi Ohno e implementado na Toyota. O principal objetivo de Ohno era produzir carros sob demanda de forma barata, sem que fosse necessário utilizar produção em massa já que não havia mercado para tantos carros no Japão naquela época. Ohno então criou o prin-

cipio fundamental do Lean: Eliminar desperdício (POPPENDIECK; POPPENDIECK; POPPENDIECK, 2003).

Em 2003 o termo Lean Software Development foi cunhado através do livro de (POPPENDIECK; POPPENDIECK; POPPENDIECK, 2003) que trouxe uma adaptação do Lean utilizado na indústria automobilística para o desenvolvimento de software.

A filosofia Lean é baseada em sete princípios, sendo eles:

- Eliminar o desperdício.
- Amplificar a aprendizagem.
- Decidir o mais tarde possível.
- Entregar o mais cedo possível.
- Empoderar a equipe.
- Construir integridade.
- Ver o todo.

Na indústria do software o princípio de eliminação de desperdício é aplicado identificando e eliminando documentos, artefatos e trabalho que não é necessário para para atingir o resultado final esperado. O foco é eliminar requisitos desnecessários, re-trabalho, e soluções complexas que poderiam ser simplificadas. Código não funcional ou documentos não consultados são exemplos de desperdícios no Lean (WAZLAWICK, 2019).

2.2.5 Scaled Agile

Inicialmente os métodos ágeis eram utilizados em projetos pequenos e médios, as equipes eram usualmente de pequeno porte e seus membros ficavam presentes na mesma sala, podendo se comunicar de maneira ágil e informal. Com o passar do tempo houve o interesse em escalonar os métodos ágeis para que eles passassem a atender as necessidades de grandes organizações com sistemas maiores (SOMMERVILLE, 2011).

Denning, Gunderson e Hayes-Roth (2008), Leffingwell (2007) e Moore e Spens (2008) levantaram alguns pontos que diferenciam o desenvolvimento de sistemas de grande porte de sistemas pequenos, dentre eles podemos destacar:

- Sistemas de grande porte usualmente são compostos por coleções de sistemas que se comunicam, frequentemente cada equipe fica responsável por um dos sistemas. É praticamente impossível que todos tenham uma visão geral de todo o sistema, logo, cada equipe tem como prioridade completar sua parte do sistema, sem que o todo seja levado em consideração (SOMMERVILLE, 2011).

- Devido ao longo tempo de aquisição e desenvolvimento é difícil manter os mesmos colaboradores do início ao fim, desta forma a informação que não é documentada se perde (SOMMERVILLE, 2011).
- Em grandes projetos há um conjunto variado de stakeholders, tornando o envolvimento de todo o conjunto de stakeholders no processo de desenvolvimento muito complexo (SOMMERVILLE, 2011).

2.2.5.1 LeSS

Large Scale Scrum, ou simplesmente LeSS, é um framework ágil que orienta a adoção do Scrum escalável em organizações compostas por mais de um time ágil que trabalham no mesmo produto. Sendo dividido em dois frameworks: LeSS e LeSS Huge (LARMAN; VODDE, 2016).

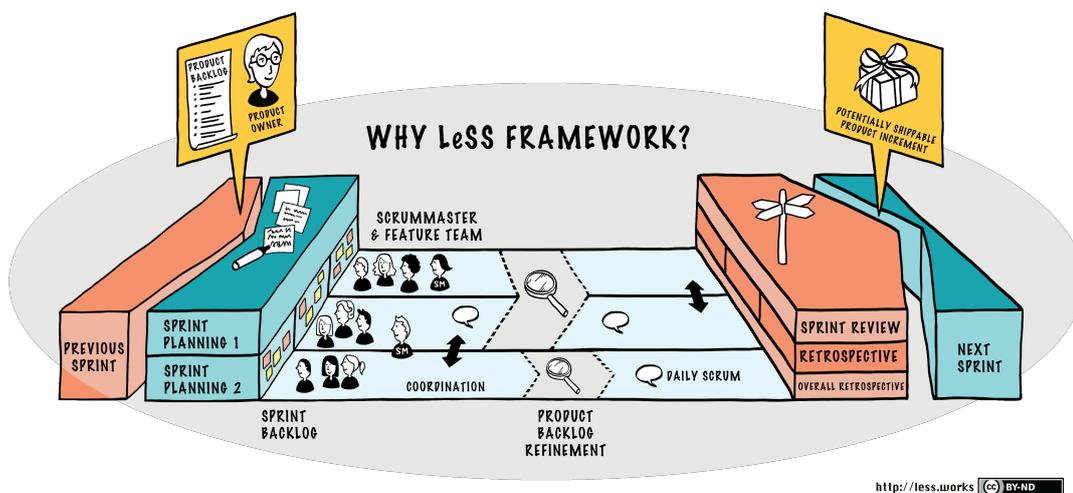
O LeSS é recomendado para organizações que possuem entre 2 e 8 times trabalhando no mesmo produto, enquanto o LeSS Huge contempla organizações com mais de 8 times, embora Larman e Vodde (2016) defendam que o LeSS também pode ser utilizado em um contexto com mais de 8 times. Os dois frameworks possuem em comum:

- Um único Product Owner e product Backlog.
- Uma sprint comum entre todos os times.
- Um incremento de produto entregável ao final da sprint.

LeSS - Um único Product Owner é responsável por todo o produto e por gerenciar um único Backlog que é utilizado por todos os times. Um Scrum Master é responsável por de um a tres times, que são multi-funcionais e trabalham no mesmo ambiente compartilhado de código, todos contribuindo para criar itens entregáveis. Há um product Backlog para cada time. As sprints são sincronizadas, tornando possível que ao final da sprint haja um incremento de produto entregável (LARMAN; VODDE, 2016).

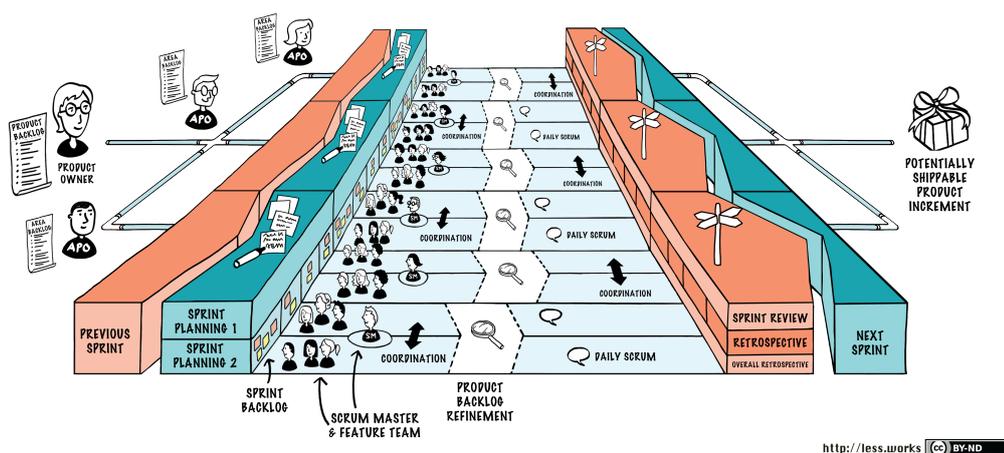
LeSS Huge - Um novo papel é introduzido, o Area Product Owner é responsável por uma área de requisitos, cada área de requisitos possui um product Backlog e múltiplos times de desenvolvimento trabalhando em conjunto. Do ponto de vista do time de desenvolvimento, cada área é uma implementação do LeSS, sendo o Area Product Owner o responsável por manter o Area product Backlog. O Product Owner, por sua vez, torna-se responsável pelo product Backlog, que alimenta as demandas de cada um dos Areal product Backlogs. As Sprint continuam sincronizadas e ao final espera-se que haja um produto entregável (LARMAN; VODDE, 2016).

Figura 6 – LeSS



Fonte: (LARMAN; VODDE, 2016)

Figura 7 – LeSS Huge



Fonte: (LARMAN; VODDE, 2016)

2.2.5.2 SAFe

Criado por Dean Leffingwell em 2011, The Scaled Agile Framework (SAFe) é um framework escalável e configurável, baseado nos princípios e valores Lean-Agile, promove orientação sobre papéis, responsabilidades, artefatos e atividades necessárias para sua aplicação em grandes organizações (LEFFINGWELL, 2018).

SAFe suporta quatro configurações de ambientes de desenvolvimento, denominados como:

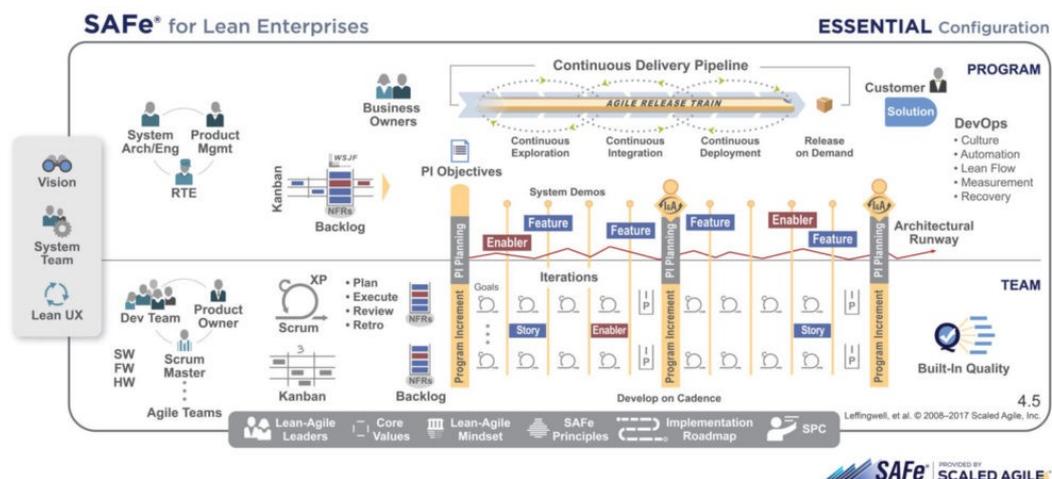
- Essencial SAFe.
- Large Solution SAFe.

- Portfolio SAFe.
- Full SAFe.

Neste trabalho apenas o Essential SAFe será apresentado, ele é considerado a base para outras configurações de SAFe.

Essential SAFe - É considerado o coração do SAFe, o ponto de partida para a implementação, a construção básica que pode evoluir para as outras configurações de SAFe, descrevendo os elementos mais críticos necessários para o funcionamento do framework (LEFFINGWELL, 2018).

Figura 8 – Essential SAFe



Fonte: (LEFFINGWELL, 2018)

Elementos chave do Essential SAFe:

- O Agile Release Train (ART) deve ser alinhado entre gerentes, times e stakeholders para uma única visão do produto, roadmap e program backlog.
- O ART entrega as funcionalidades e a estrutura técnica necessária para gerar valor e promover uma base sustentável.
- As iterações dos times são sincronizadas e possuem a mesma duração, data de início e data de fim.
- Cada ART entrega um incremento testado e que possui valor a cada duas semanas.
- Program Increments (PI) fornecem incrementos mais longos.
- Soluções podem ser lançadas sob demanda, sem estar atreladas ao PI, baseadas nas necessidades organizacionais.

- O PI planning é realizado presencialmente, para assegurar colaboração, alinhamento e rápida adaptação.
- O ART constrói e mantém processos de entrega contínua, usados para desenvolver e lançar pequenos incrementos com valor agregado.
- O ART promove práticas de User experience (UX) através de princípios Lean(LEFFINGWELL, 2018).

O essencial SAFe conta também com três atividades primárias: PI Planning, System Demo e Inspect and Adapt e cinco papéis não descritos em outros métodos ágeis: System Architect/Engineer, Product Management, Release Train Engineer (RTE), Business Owners e Customers [Leffingwell \(2018\)](#).

3 ESTADO DA ARTE

Segundo o Manifesto Ágil, o termo ágil é descrito como a capacidade de responder às mudanças e a habilidade de lidar com um ambiente incerto e turbulento (BECK *et al.*, 2001). A pesquisa State of Agile Report, realizada anualmente pela VersionOne, destaca que o ano de 2021 foi marcado pelo forte crescimento da adoção do ágil nos times desenvolvedores de software, com crescimento de 37% em 2020 para 86% em 2021. A habilidade para gerenciar mudanças de prioridade, aceleração na entrega do software e aumento de produtividade do time foram as maiores vantagens apontadas pelos times na pesquisa. Dentre as métricas mais utilizadas, (49%) possuem métricas com foco no valor do negócio entregue, (49%) possuem foco na satisfação do cliente e 45% utilizam a medida da velocidade de entrega (VERSIONONE, 2021).

Visto o expressivo crescimento da adoção ágil e uso de métricas ágeis, surge o interesse de identificar e analisar o impacto e o contexto de uso de métricas em organizações desenvolvedoras de software, por meio de uma revisão sistemática da literatura (RSL) (KEELE *et al.*, 2007).

O objetivo de uma RSL é buscar respostas para perguntas de pesquisa, com o objetivo de embasar conclusões através de uma busca na literatura realizada utilizando procedimentos sistemáticos. Este trabalho seguirá o guia proposto por Keele *et al.* (2007), o guia organiza a RSL em três principais atividades, sendo elas: planejar a revisão, conduzir a revisão e relatar a revisão. Os detalhes de cada atividade serão apresentados a seguir.

3.1 QUESTÃO DE PESQUISA

A questão geral de pesquisa "Como são utilizadas métricas de software em organizações que utilizam métodos ágeis?" foi derivada a partir da necessidade de pesquisa identificada, em seguida termos de pesquisa foram definidos, *strings* de busca foram desenvolvidas e fontes de dados selecionadas.

Da pergunta de pesquisa foram derivadas as questões de análise:

- a) Q1: Qual o contexto de uso das métricas?
- b) Q2: Qual o perfil das organizações de software que utilizam?
- c) Q3: Quais são as métricas utilizadas?
- d) Q4: Como as métricas foram selecionadas?
- e) Q5: Como as métricas são classificadas?
- f) Q6: Qual o impacto do uso de métricas?
- g) Q7: Qual o público-alvo das métricas?

3.2 TERMOS DE BUSCA

A busca incluiu os termos *software*, *metric*, *quality* e *agile*, buscando por relatos e descrições de uso de métricas em abordagens de desenvolvimento ágil de software. Com enfoque em trabalhos que especificavam o uso de métricas nas metodologias mais utilizadas segundo [VersionOne \(2021\)](#).

3.3 FONTE DE DADOS

As buscas foram realizadas nas bibliotecas digitais mais relevantes para a área de Engenharia de Software, sendo elas: IEEEExplore, ScienceDirect, Springer, Scopus, ACM e Wiley Online Library. Foi aplicada também a técnica de *Backward Snowballing* que consiste em utilizar as referências dos estudos encontrados para localizar outros trabalhos e autores relevantes ([WOHLIN, 2014](#)).

3.4 STRINGS DE BUSCA E CRITÉRIOS

Na tabela 1 é apresentada uma *string* de busca genérica, derivada das perguntas de pesquisa. Para cada motor de busca a sintaxe *string* genérica precisou ser adaptada.

Tabela 1 – String de Busca

(software) AND (metric OR measure OR indicator OR KPI) AND (agile OR scrum OR kanban OR xp OR extreme programming OR scrumban OR lean OR fdd)) OR ((software) AND(metric OR measure OR indicator OR KPI) AND (agile OR scrum OR kanban OR xp OR extreme programming OR scrumban OR lean OR fdd))
--

Fonte: autora.

3.5 CRITÉRIOS DE INCLUSÃO E EXCLUSÃO

Os critérios de inclusão e exclusão definidos foram: estudos disponíveis em bases de dados científicas; disponibilizados a partir de 2007, ano no qual o tema sobre métricas, passou a ter maior relevância; artigos publicados em conferências ou periódicos; estudos que relatam a experiência do uso de métricas, estratégia de implementação utilizada, o uso em diferentes perfis de empresas desenvolvedoras de software em contextos ágeis. Os estudos devem referir-se a relatos de experiência, estudo de caso ou estudos empíricos, abordando informações referentes ao uso de métricas para contextos ágeis. Estudos que abordem somente a descrição de métricas serão desconsiderados, bem como estudos não acessíveis por meio do portal CAPES.

3.6 SELEÇÃO DOS ESTUDOS

A RSL foi realizada conforme protocolo definido, em colaboração com citeonlinestephanie e citeonlinegustavo. A partir dos estudos primários encontrados e selecionados foram extraídos os dados necessários para responder à pergunta de pesquisa, realizadas as análises e derivadas as conclusões. Três ciclos foram realizados para a seleção dos estudos: Busca Inicial, Iteração 1 e Iteração 2. Durante a busca inicial foram encontrados 100 artigos nas bases de dados selecionadas, após a leitura de todos os títulos e *abstracts* e aplicação dos critérios de inclusão e exclusão, durante a iteração 1, foram selecionados 90 estudos. Após a leitura completa de todos os artigos e aplicação da técnica de *Snowballing*, a iteração 2 resultou em 42 artigos relevantes.

A partir da leitura completa de todos os artigos selecionados, as informações consideradas relevantes para responder a pergunta de pesquisa foram selecionadas e extraídas. Todos os resultados obtidos passaram por dois ciclos de revisão, ajustes e validação. Todos os artigos selecionados foram considerados relevantes, os dados extraídos são estruturados, analisados e interpretados a seguir.

3.7 ANÁLISE

A análise e interpretação dos dados coletados durante a RSL visa responder a pergunta geral de pesquisa. Durante este passo, os artigos selecionados foram categorizados por meio da técnica de *Reciprocal translation* (KEELE *et al.*, 2007), que consiste na síntese dos resultados a partir de estudos semelhantes, traduzindo cada um dos outros casos em categorias comuns. Os resultados foram agrupados em quatro categorias: Contexto do Estudo, Métricas Utilizadas, Estratégia de Implementação e Resultados Observados. Apresentados na tabela a seguir:

Figura 9 – Estado Da Arte

Classificação		Estudos Seleccionados	
Q1 - Qual o contexto de uso das métricas?	Q1.1 Contexto de uso	Indústria	[1], [2], [4], [12], [13], [14], [17], [18], [19], [24], [29], [36], [37], [42], [46], [55], [62], [63], [65], [67], [69], [70], [72], [79], [82], [84], [86], [89] - 28
		Academia	[5], [8], [30], [48] - 4
		Não aplicado	[11], [15], [21], [25], [26], [27], [32], [34], [51], [52] - 10
			total 42
	Q1.2 Método Ágil	Scrum	[1], [8], [17], [18], [19], [24], [30], [32], [36], [37], [55], [63], [67], [69], [72], [84] - 16
		Kanban	[13] - 1
		XP	[5], [14], [46], [86] - 4
		Lean	[2], [4], [11], [12], [29], [82] - 6
		Safe	[70] - 1
	Q2 - Qual o perfil das organizações de software que utilizam?	Q2.1 Tamanho Equipe	5 a 9 pessoas
10 pessoas			[5]
Diversas equipes			[2], [12], [13]
Q2.2 Experiência (média em anos)		1 a 5 anos	[1], [46], [48]
		Experientes	[2], [12]
Q2.3 Domínio de aplicação		Educacional	[79]
		Financeiro	[65]
		Geral	[1], [2], [4], [8], [12], [13], [14], [17], [18], [19], [24], [29], [36], [37], [42], [46], [48], [55], [72], [86] - 20
		Segurança	[26]
		Telecomunicações	[70]
Q2.4 Tamanho do produto		Grande	[1], [2], [4], [8], [13], [14], [19], [29], [65] - 9
		Médio	[17], [24], [46]
		Pequeno	[36], [37], [48]
Q2.5 Tipo produto		Aplicação	[4], [14], [17], [18], [37], [46], [48] - 7
		Misto	[1], [2], [19], [29]
		Mobile	[8], [55]
		Outro	[5]
		Web	[13], [24], [36], [70], [79] - 5
Q2.6 Quantidade Organizações	1 a 2 organizações	[5], [8], [13], [14], [18], [19], [24], [29], [36], [37], [46], [55], [65], [70], [72], [79], [82], [84], [17] - 19	
	3 a 4 organizações	[42], [62], [89] - 3	
	10 a 30 organizações	[25], [34], [63] - 3	

Fonte: (SILVA LEAL, Em construção)

Figura 10 – Estado Da Arte

Classificação		Estudos Seleccionados	
Q3 - Quais são as métricas utilizadas?	Q3.1 Métricas	Schedule and Progress	[1], [11], [13], [15], [17], [18], [19], [25], [32], [37], [46], [48], [51], [70], [89] - 15
		People	[1], [5], [14], [30] - 4
		Product	[1], [2], [4], [11], [12], [13], [14], [18], [24], [27], [29], [36], [51], [65], [89] - 15
		Process	[2], [4], [12], [15], [18], [19], [32], [37] - 8
		Technology	[4], [8], [12], [14], [15], [17], [21], [24], [25], [26], [29], [34], [42], [46], [52], [56], [72], [79], [86] - 19
		Customer	[14], [18] - 2
Q4 - Como as métricas foram selecionadas?	Q4.1 Como as métricas foram selecionadas	Estudo da literatura	[1], [4], [14], [15], [19], [21], [26], [27], [34], [36], [42], [46], [51], [52], [55], [65] - 16
		Entrevistas (equipes)	[2], [5], [12], [17], [19], [25], [29], [62], [65], [84] - 10
		Especialista/Grupo de pesquisa	[4], [8], [13], [15], [18], [24], [30], [32], [37], [48], [72], [82], [86] - 13
Q5 - Como as métricas são classificadas?	Q5.1 Estrutura Classificação	Equipe	[1], [2], [4], [5], [11], [12], [13], [14], [15], [17], [18], [19], [24], [25], [26], [27], [29], [30], [32], [36], [37], [42], [46], [48], [51], [52], [55], [62], [65]
		Qualidade	[1], [2], [4], [8], [11], [13], [15], [17], [18], [19], [21], [24], [25], [26], [27], [29], [32], [34], [36], [37], [42], [46], [48], [51], [52], [62], [70], [89]
		Processo	[15], [18], [24], [25], [26], [29], [36], [51], [52], [55], [89]
		Produto	[15], [36]
		Organizacional	[18], [89]
Q6 - Qual o impacto do uso de métricas	Q6.1 Área impactada	Equipe	[14], [15], [17], [18], [36], [55]
		Qualidade processo	[1], [2], [5], [8], [15], [18], [32], [36], [37], [42], [46], [48], [51], [52], [55], [62], [70], [72], [89]
		Qualidade produto	[1], [2], [4], [8], [11], [12], [13], [15], [17], [18], [19], [21], [24], [25], [26], [27], [29], [30], [32], [34], [36], [42], [46], [51], [52], [72]
		Qualidade serviço	[4], [89]
Q7 - Qual o público-alvo das métricas?	Q7.1 Público-alvo das métricas	Equipe	[1], [2], [5], [8], [11], [12], [13], [14], [15], [17], [18], [19], [21], [24], [25], [26], [27], [29], [30], [32], [34], [36], [42], [46], [48], [51], [55], [62]
		Gerência	[1], [2], [18], [24], [25], [26], [27], [29], [30], [32], [36], [42], [46], [48], [51], [52], [55], [62]
		Cliente	[1], [4], [5], [11], [15], [17], [18], [42], [52]

Fonte: (SILVA LEAL, Em construção)

Alguns estudos não apresentaram informações referentes a classificação definida, em função disso, alguma categorias não equivalem ao total de estudos selecionados. Na sequência, as questões de análise são apresentadas e relacionadas com as classificações e categorias apresentadas na Tabela Extração dos Dados.

3.7.1 Q1: Qual o contexto de uso das métricas?

Contexto de Uso: 64,2% (27) dos artigos abordaram a aplicação de métricas em um contexto industrial, 9,5% (4) dos estudos foram aplicados na academia e 23,8% (10) dos estudos não envolvem a aplicação das métricas.

Método Ágil: 38% (16) dos artigos analisados utilizam a abordagem Scrum, seguido pelo Lean com 14,2% (6), XP com 9,5% (4), e Kanban e Safe ambos com 4,2%, (1) estudo cada. Demais estudos não definiram explicitamente a abordagem ágil empregada.

3.7.2 Q2: Qual o perfil das organizações de software que utilizam?

Tamanho Equipe: A maioria dos estudos selecionados não apresentam dados de tamanho dos times envolvidos na aplicação, dentre os resultados encontrados, tem-se: 7% (3) estudos relataram envolver diversas equipes, seguido por 1 estudo com time de 5 a 9 pessoas e outro artigo abordando uso de métricas com time de 10 pessoas.

Experiência do Time: 7% (3) artigos informaram lidar com times com experiência de 1 a 5 anos e outros dois estudos informam trabalhar com times experientes. Demais artigos não definiram a experiência dos times envolvidos.

Domínio de Aplicação: 47,6% (20), se identificaram como Geral, ou seja softwares para diferentes áreas, seguido pelas áreas de Educacional, Financeiro, Segurança e Telecomunicações, todos com 4,2% (1) estudo cada.

Tamanho do Produto: 21,4% (9) identificaram o produto como sendo de tamanho grande, seguido por produtos de médio e pequeno porte, ambos com 7% (3) estudos cada.

Tipo do Produto: 16,6% são produtos de aplicação (7), seguido por produtos web 11,9% (5), produtos misto 9,5% (4) e para mobile 4,7% (2).

Quantidade de Organizações: 45% (19) abordaram de 1 a 2 organizações, seguida por 7% (3) dos estudos envolvendo 3 a 4 organizações e outros 7% (3) envolvendo 10 a 30 organizações.

3.7.3 Q3: Quais são as métricas utilizadas?

Para melhor compreensão e análise as métricas foram agrupadas de acordo com a divisão do PSM: Schedule and progress, People, Product, Process, Technology e Customer, dentro desses grupos tem-se as seguintes métricas extraídas dos estudos:

- a) Schedule and progress: 35,7% (15) estudos. Métricas apresentadas: horas, esforço, velocidade, lead time, cycle time, WIP, porcentagem de lançamentos entregues no prazo, projeto entregue no prazo, trabalho em progresso, gestão do risco, número de pontos entregues, time to market, flow efficiency, capacidade ideal

- b) People: 9,5% (4) estudos. Métricas apresentadas: colaboração do time, eficiência (em termos de decisões corretas), nível de motivação, performance (em nível de aprendizado), fator foco, tamanho do time, comunicação, experiência no papel, tamanho do time (desenvolvedores), nível de experiência do time, domínio de expertise, turnover, fatores morais.
- c) Product: 35,7% (15) estudos. Métricas apresentadas: qualidade, custo, ROI, visibilidade do valor, quantidade de tarefas, sucesso entregas, throughput
- d) Process: 19% (8) estudos. Métricas apresentadas: cerimônias de acompanhamento planejadas e realizadas, reuniões com cliente foram realizadas, quantidade de mudanças realizadas, alteração de prioridade, nível de maturidade no ágil, número de membros alterados, crescimento do escopo, pontos entregues não planejados, quantidade de itens acumulados, diagrama de fluxo acumulado (mede a quantidade de itens por estado no fluxo), tarefas reabertas, precisão das estimativas, desperdício, impedimentos
- e) Technology: 45,2% (19) estudos. Métricas apresentadas: code churn, severidade dos defeitos, quantidade de bugs, horas em bugs, velocidade da correção dos problemas, média de testes, performance build diária, performance dos testes, número de deploys em produção, contagem de dependência
- f) Customer: 4,7% (2) estudos. Métricas apresentadas: porcentagem de aceitação do cliente, indicador de satisfação do cliente, comunicação com cliente

3.7.4 Q4: Como as métricas foram selecionadas?

Sobre a seleção das métricas, pode-se observar os seguintes resultados: 38,5% (16) estudos buscaram e selecionaram as métricas com base em pesquisas da literatura, 30,9% (13) estudos utilizaram de especialistas e comitês de pesquisa dentro das organizações para definir as métricas e 23,8% (10) estudos realizaram entrevistas com as equipes para selecionar as métricas que foram utilizadas posteriormente na aplicação.

Em [1], um estudo da literatura foi realizado para coletar informações e explorar o campo do desenvolvimento ágil de software, governança, KPIs e intervenções. [4], [18], [82], [84], e [86] utilizaram o método *Goal-Question-Metric* (GQM) para derivar um conjunto de métricas das metas dos patrocinadores do projeto, com base em uma revisão da literatura sobre os fatores de sucesso de um projeto.

[1] realizou também um estudo de caso para coletar uma primeira impressão sobre o desenvolvimento ágil de software adotado na prática, além da coleta de informações por meio de entrevistas e análise de documentos. O objetivo era descobrir KPIs e intervenções usadas na empresa para gerenciar o processo. Em [2] as métricas foram selecionadas através da coleta de dados por meio de uma série de fóruns,

incluindo grupos de foco, entrevistas, discussões, análise de ferramentas e análise de documentos para então definir as métricas adequadas para os times. Em [8] foram criados grupos de para discussão de objetivos e explicação ao público-alvo (time, líderes e gerência), depois, fluxos de trabalho foram descritos e as métricas selecionadas para medir a adoção e uso. No [12], [19] e [25] foram realizadas entrevistas e enviados surveys para definir as métricas do estudo.

Em [17] foi realizado um processo iterativo com as equipes, utilizando *Query Domain Language*, assim que os participantes estavam satisfeitos com as métricas eles implantavam para garantir que funcionem no contexto dos projetos. Em [24] uma equipe conjunta de pesquisadores-praticantes selecionou 25 métricas de processo candidatas que passaram a ser alimentadas com os dados da empresa. As métricas de processo resultantes foram usadas pela equipe durante suas reuniões para refletir sobre o desempenho do processo e a qualidade do produto, bem como para estimar tarefas (execução de ações). Posteriormente, as métricas do processo foram avaliadas em uma sessão retrospectiva com a equipe. Por fim, a equipe aprendeu o subconjunto de métricas de processo que são mais eficazes para os problemas diagnosticados (especificando aprendizado).

Em [62] o conjunto de métricas foi selecionado após uma série de workshops com as 4 empresas envolvidas no estudo de caso. Em [15], [30], [32], [34], [37], [46], [48] e [72] os autores utilizaram sua experiência e conhecimento para definir as métricas.

3.7.5 Q5: Como as métricas são classificadas?

Com relação aos grupos de métricas, a maior parte dos estudos selecionaram e aplicaram métricas para acompanhamento das informações relacionadas ao time e a qualidade do produto. Seguido de métricas para acompanhamento do processo.

3.7.6 Q6 - Qual o impacto do uso de métricas

Área impactada: A maior parte dos estudos citou produto e processo como áreas mais impactadas, seguido por equipe e qualidade do serviço.

Descrição do impacto:

Em [1] as métricas voltadas para o time e as métricas individuais tornaram visível a sobrecarga de trabalho que o time estava ocorrendo, e o impacto do estresse de um membro no time como um todo, outro ponto observado foi que as métricas também apoiaram na correta distribuição das tarefas, tornando possível dar atenção para o desenvolvimento profissional da equipe.

Ainda em [1] as métricas voltadas para tarefas apoiaram no gerenciamento das mesmas, bugs críticos passaram a receber preferência em detrimento as tarefas de evolução e tarefas inacabadas passaram a ser movidas para o próximo release. A qualidade foi impactada positivamente, as métricas trouxeram transparência para o

software construído evidenciando para toda a empresa os testes aprovados e bugs resolvidos, assim a equipe tornou-se mais motivada a evitar problemas de qualidade e resolve-los rapidamente, quando existiam.

Em [8] descobriu-se que mesmo que alguns indicadores apontassem um problema, a equipe tinha razões para trabalhar da maneira que trabalhavam. Nesses casos, está relacionado a um dos princípios-chave do manifesto ágil: "Indivíduos e interações acima de processos e ferramentas", e não foi forçada as equipes a mudarem sua abordagem. Os instrutores e líderes do projeto ficaram satisfeitos com a planilha de métricas porque permitiu terem uma visão geral rápida sobre o status do projeto. Ao observar uma irregularidade, por ex. um grande número de compilações com falhas, eles poderiam ir para os respectivos sistemas e investigar as causas raízes ou conversar com os membros responsáveis da equipe de desenvolvimento. Portanto, as métricas foram, muitas vezes, um bom primeiro indicador e provaram ser uma ferramenta poderosa para supervisionar vários projetos ao mesmo tempo.

Já em [11] o uso das métricas teve como objetivo principal o apoio às equipes de desenvolvimento lean e ágil, permitindo que elas avaliassem seu desempenho e se sentissem mais motivadas para alcançarem um nível de qualidade mais alto. Prezando pelo melhor, as equipes precisaram fazer experiências, receber feedback e agir. Além disso, a fim de avaliar se uma ação criou valor e ajudou a equipe, foi feito um de-para dos casos para ser medido também. O modelo de indicadores de desempenho ágeis fornece uma solução ampla para medir a situação da equipe para que melhorias possam ser feitas.

Em [13] o uso de métricas possibilitou uma análise sobre o WIP dos times de desenvolvimento. Um WIP baixo usualmente reduz o tempo de espera de uma issue para a troca de etapa de desenvolvimento, o que usualmente costuma ser positivo, entretanto segundo o descrito no estudo o WIP baixo pareceu reduzir a produtividade da equipe, pode-se concluir então que no caso do estudo a equipe precisa equilibrar o WIP dependendo da prioridade do tempo de entrega versus produtividade. Equipes bem estruturadas e com domínio da aplicação tendem a lidar bem com WIP maior, e tal análise só foi possível por fazerem uso de métricas de medição.

Em [14] houve uma redução de 65% nas taxas de defeito de pré-lançamento do produto, uma redução de 35% na taxa de defeito no período de liberação e uma melhoria de 50% na produtividade (medido pela saída do código), todas melhorias pautadas no uso de métricas no dia-a-dia do time.

Em [17] foi identificado que as métricas conseguiram compreender a incerteza e responderam à dinâmica inerente ao contexto de projeto ágil. Apoiou fortemente os clientes considerando os diferentes contextos causados pelas mudanças, em vez de apenas um único ou três cenários (o melhor, o provável ou o pior caso) e facilitou a repriorização conforme as realidades do cliente foram mudando ao longo do tempo. O uso das métricas ágeis permitiram realizar gastos incrementais enquanto o foco

se dava nas partes críticas da funcionalidade do software, essenciais para cumprir o objetivo do projeto. O time conseguiu compreender que nem todos os requisitos e as opções de projeto de arquitetura têm o mesmo valor.

Em [18] o uso de métricas tornou transparente o débito técnico existente, e evidenciou que a visibilidade do valor de negócio e questões de estimativas eram as necessidades mais urgentes da organização. O débito técnico pareceu impactar principalmente na produtividade da equipe, eficiência e na qualidade do produto.

Em [19], [46], [70] e [72] a produtividade do time foi impactada positivamente pelo uso de métricas, as squads puderam observar como melhorar a taxa de conclusão de histórias planejadas e reduzir o tempo de ciclo. As squads conseguiram reduzir seu tempo de ciclo mantendo um espaço aberto no planejamento da sprint (por exemplo, aumentando sua proporção de tempo restante). Finalmente, reduzindo equipes de trabalho não planejadas pôde aumentar sua previsibilidade de entrega (por exemplo, taxa de conclusão de histórias planejadas).

Em [24] foram observadas grandes vantagens no uso das métricas de processo, como a eficácia do processo e na produtividade da equipe. A solução proposta melhorou a maneira como os desenvolvedores relatavam o tempo gasto em problemas/tickets e permitiram a comparação com o esforço planejado. Ao incluir painéis e as métricas de processo observou-se uma melhora no desenvolvimento de software, disposição e eficiência na geração de relatórios, tempo gasto e planejamento do esforço. Além disso, a lacuna entre o esforço planejado e o gasto seguiu diminuindo continuamente, o que indica que houve uma melhora nas estimativas. As métricas propostas permitiram um rastreamento eficiente de tarefas e questões no projeto, por desenvolvedor e por sprint.

Em [29] a análise das métricas tornou possível a detecção precoce de falhas estruturais, antecipando o tratamento de problemas mais críticos. Também foi possível detectar problemas de qualidade de código que apresentassem riscos de curto prazo e limitações arquitetônicas de longo prazo, através da análise de qualidade estrutural. Durante o primeiro trimestre, os dados agregados entre os aplicativos indicaram que a transformação começou a entregar substancialmente mais funcionalidades sem diminuição da qualidade. Os aplicativos alcançaram entre 30% a 48% de melhoria na qualidade quando medidos por defeitos críticos. Uma média de 28% de ganhos de produtividade foi alcançada pelas equipes que implementaram práticas ágeis repetíveis, análise automatizada de qualidade estrutural, automação de teste e uso consistente da plataforma DevOps para integração e entrega contínuas. A equipe percebeu uma redução de 41% nos defeitos, juntamente com essa melhoria de qualidade, a equipe entregou mais funcionalidade em períodos de tempo mais curtos, resultando em uma melhoria de 30% na produtividade.

Em [36] destaca que o uso de métricas durante as retrospectivas fornece à equipe uma fonte diferente de informações para a melhoria contínua.

3.7.7 Ameaças à validade

Sendo métricas um termo de larga abrangência, foram retornados muitos estudos abordando o assunto nas mais variadas formas. Para diminuir o risco de pesquisas relevantes não serem consideradas, os termos de pesquisa selecionados continham conceitos relacionados a questão de pesquisa e seus sinônimos. Desta forma apenas os estudos com alto grau de relevância foram considerados.

4 PROCESSO ATUAL

Neste capítulo será apresentado o processo atual e o contexto de aplicação do estudo de caso.

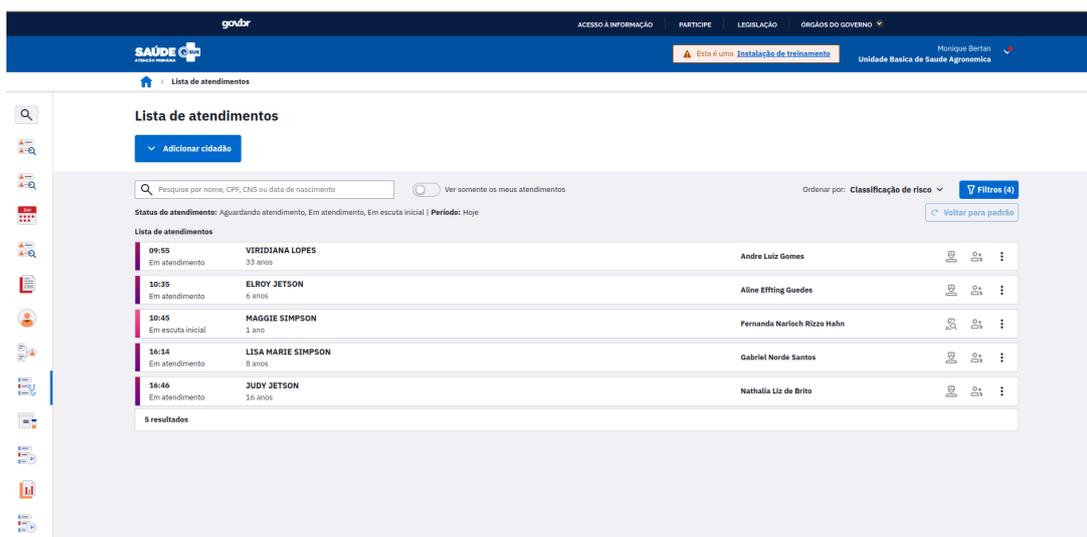
4.1 CONTEXTO

O Laboratório Bridge é um laboratório integrado ao Centro Tecnológico (CTC) e ao Centro de Ciências da Saúde (CCS) da Universidade Federal de Santa Catarina (UFSC), criado em 2013, atua na pesquisa e desenvolvimento de soluções tecnológicas inovadoras para a área da saúde e educação, em parceria com o ministério da saúde e ministério da educação (BRIDGE, 2022). Atualmente o laboratório possui mais de 160 colaboradores e produtos na área da saúde e educação em desenvolvimento.

4.1.1 Produto

A unidade organizacional escolhida para a aplicação do estudo de caso foi o e-SUS APS PEC, responsável pelo desenvolvimento do e-SUS APS PEC, um sistema de prontuário eletrônico do cidadão destinado ao registro clínico do cidadão e auxílio no processo administrativo de trabalho de profissionais da atenção primária em saúde, atuando desde especialidades médicas até a gestão das unidades básicas de saúde (BRIDGE, 2022).

Figura 11 – e-SUS APS



The screenshot displays the 'Lista de atendimentos' (Appointment List) interface. At the top, there is a navigation bar with 'goubr' logo and menu items: 'ACESSO À INFORMAÇÃO', 'PARTICIPE', 'LEGISLAÇÃO', and 'ORÇÃOS DO GOVERNO'. Below the navigation bar, there is a search bar and a button 'Adicionar cidadão'. The main content area shows a table of appointments with columns for time, status, patient name, age, and provider name. The table is filtered by 'Hoje' (Today) and shows 5 results.

Time	Status	Nome	Idade	Profissional
09:55	Em atendimento	VIRIDIANA LOPES	33 anos	Andre Luiz Gomes
10:35	Em atendimento	ELROY JETSON	6 anos	Aline Effling Guedes
10:45	Em escuta inicial	MAGGIE SIMPSON	3 anos	Fernanda Narloch Rizzo Hahn
16:14	Em atendimento	LISA MARIE SIMPSON	9 anos	Gabriel Norde Santos
16:46	Em atendimento	JUDY JETSON	16 anos	Nathalia Liz de Brito

Fonte: Autora - 2022

Atualmente a unidade organizacional conta com 8 equipes ágeis, cada uma é composta por 1 Scrum master, 1 Product Owner, e o time de desenvolvimento. Há também um 1 Gerente de projeto, 4 designers compartilhados entre as equipes e um

Grupo de Trabalho (GT), que faz o papel de stakeholder do produto. O GT é formado por colaboradores do ministério da saúde que atuam diretamente em Brasília, o grupo possui vasto conhecimento de negócio e acesso e conhecimento das necessidades do cliente, o ministério da saúde.

4.1.2 Equipes

As equipes selecionadas atuam na unidade organizacional e-SUS APS e foram escolhidas levando em consideração a proximidade da autora do trabalho com o membros das equipes e os SMs responsáveis por cada equipe.

Para a realização deste trabalho, um estudo de caso será aplicado em duas equipes ágeis da unidade organizacional e-SUS APS PEC, sendo elas:

A equipe "Foundation" é constituída por 6 integrantes, ocupando os seguintes cargos:

Tabela 2 – Membros da equipe F

Cargo	Formação	Experiência na área
SM	Bacharel em Ciências da Computação	4 anos
PO	Graduando em Ciências da Computação	3 anos
Desenvolvedor e tech lead	Bacharel em Ciências da Computação	5 anos
Bolsista de desenvolvimento	Graduando em Ciências da Computação	1 anos
Bolsista de desenvolvimento	Graduando em Sistemas de Informação	2 anos
Desenvolvedor	Graduando em Sistemas de Informação	2 anos

Fonte: Autora (2022).

A equipe S é constituída por 7 integrantes, ocupando os seguintes cargos:

Tabela 3 – Membros da equipe S

Cargo	Formação	Experiência na área
SM - QA	Graduando em Sistemas de Informação	4 anos
PO	Bacharel em Ciências da Computação	4 anos
Desenvolvedor e tech lead	Bacharel em Ciências da Computação	4 anos
Desenvolvedor	Graduando em Sistemas de Informação	1,5 anos
Desenvolvedor	Graduando de Design	4 anos
Bolsista de desenvolvimento	Graduando em Ciências da Computação	0,5 anos
Bolsista em QA	Graduando em Ciências da Computação	2 anos

Fonte: Autora (2022).

4.1.3 Processo de desenvolvimento

O processo de desenvolvimento executado na unidade organizacional e-SUS APS segue os princípios da metodologia Scrum, apresentada no capítulo 2.2.1 deste documento.

A Sprint se inicia com a reunião de planejamento da Sprint, o Product Owner

seleciona um conjunto de tarefas previamente refinadas e priorizadas, o time de desenvolvimento avalia a descrição de cada tarefa, as regras de negócio e os critérios de aceitação definidos pelo Product Owner e define uma pontuação de esforço, baseado na complexidade analisada, previsão de tempo de desenvolvimento e testes necessários para sua conclusão. Utilizando como parâmetro pontos de esforço concluídos em Sprints passadas e com o auxílio do Scrum Master, um conjunto de tarefas é definido para ser executado durante a Sprint, o conjunto de tarefas selecionadas constitui o que é chamado de Sprint Backlog.

Durante as próximas 2 semanas, a equipe executa as tarefas do Sprint Backlog e diariamente o time de desenvolvimento se reúne para a reunião de Daily Scrum, durante a reunião cada membro deve falar rapidamente sobre a tarefa que está realizando e possíveis impedimentos. Semanalmente o Scrum Master de cada equipe escreve um breve texto sobre as atividades da equipe durante a última semana, o texto é compartilhado em um canal com todos os SMs, o objetivo é verificar se não há dependências ou impactos nas tarefas de outras equipes.

Ao final da Sprint outras duas cerimônias são realizadas: Sprint Retrospective e Sprint Review. Durante a retrospectiva o time faz uma análise do decorrer da sprint, determinando possíveis aspectos que podem ser melhorados durante as próximas sprints. A reunião de review apresenta o resultado da sprint para o Product Owner e para o gerente de projeto, afim de verificar se o resultado final do produto está de acordo com o definido no início da Sprint.

4.1.4 Roda ágil

Trimestralmente todas as equipes do laboratório realizam a aplicação da roda ágil. A cerimônia é de responsabilidade do Scrum Master de cada equipe, que define o numero de reuniões necessárias e alinha com o time os melhores dias e horários. Durante a cerimônia cabe ao Scrum Master conduzir a leitura de cada categoria e a discussão entre os membros da equipe sobre a percepção de maturidade da equipe com relação a categoria em discussão, posteriormente os membros do time votam simultaneamente uma nota entre 1 e 5 para avaliar a categoria discutida. Caso a maioria tenha votado a mesma nota, aquela nota deve ser atribuída a categoria avaliada, se houver empate uma nova rodada de discussões é iniciada e a votação se repete, até que haja uma maioria concordante. O processo é repetido ao longo de todos os 4 grupos e 20 categorias presentes na roda ágil.

Ao final da votação, o resultado é uma planilha com as notas atribuídas a cada categoria, neste momento a equipe pode visualizar dentre as categorias existentes quais possuem notas mais baixas e podem ser melhoradas. Um conjunto de categorias, cujo a quantidade varia entre equipes, é escolhido para o desenvolvimento de planos de ação. Cada categoria escolhida deve possuir ao menos um plano de ação definido,

cada plano de ação tem um membro da equipe como responsável interno e deve ser descrito de forma a alcançar um objetivo.

Mensalmente o andamento dos planos de ação de cada equipe são apresentados durante uma reunião entre todos os Scrum Masters das equipes, o Gerente do projeto e um integrante da frente de melhoria contínua, da equipe de gestão. Mensalmente a progressão dos planos de ação novamente é apresentada ao Gerente de projeto durante a reunião de Sprint Review, neste momento a equipe também está presente e pode contribuir com a sua visão sobre o andamento dos planos de ação. Ao final do trimestre os planos são revistos por cada equipe ágil e pela equipe de Scrum Masters do projeto, o sucesso, falha e/ou impacto dos planos de ação são discutidos e possíveis melhorias são endereçadas.

A cada trimestre o ciclo se repete com uma nova aplicação e novos planos de ação, espera-se que as categorias em que houveram intervenção recebam uma nota melhor que a anterior.

5 PLANEJAMENTO DO ESTUDO DE CASO

Neste capítulo é apresentado o planejamento do estudo de caso utilizando a abordagem proposta por Runeson e Höst (2009). Segundo o autor, um estudo de caso é composto por 5 atividades principais, sendo elas: planejamento do estudo de caso, preparação para a coleta de dados, coleta de dados, análise dos dados coletados e o relatório final do estudo de caso. Este capítulo aborda a primeira atividade, o planejamento do estudo de caso.

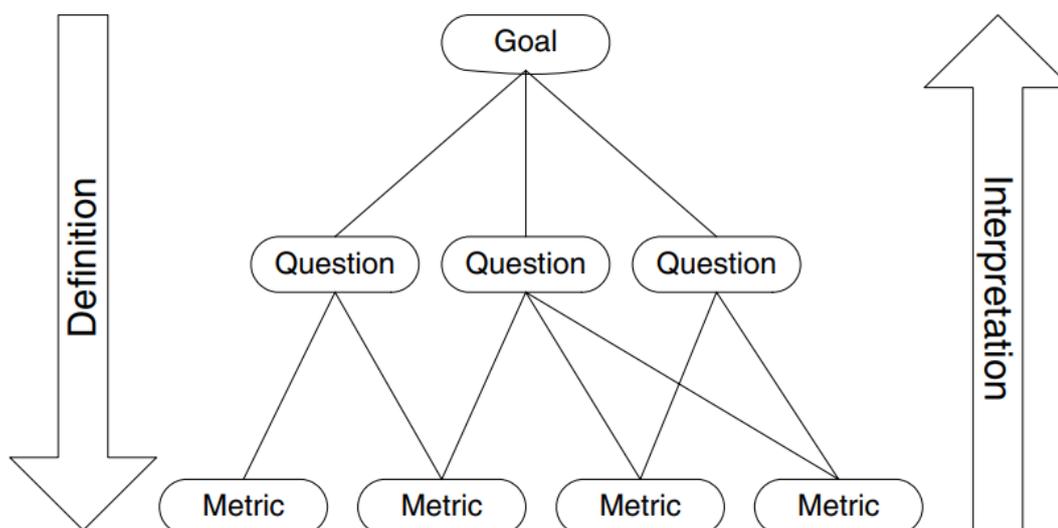
5.1 OBJETIVO DO ESTUDO DE CASO

Como definido no capítulo 1.1, o objetivo deste trabalho é avaliar a seleção e aplicação de métricas ágeis em um ambiente organizacional através de um estudo de caso na unidade organizacional e-SUS APS do Laboratório Bridge. A partir desse objetivo, a seguinte pergunta de pesquisa foi definida: *"Como a aplicação de métricas ágeis impactam no processo de acompanhamento de planos de ação de um time de desenvolvimento de software?"*.

O estudo de caso será executado entre os meses de outubro e novembro de 2022, em duas equipes da unidade organizacional e-SUS APS, descritas no capítulo 4. Para que possa ser feita a avaliação de um estudo de caso, deve-se primeiramente definir os objetivos da avaliação. Neste trabalho, para definição dos objetivos de medição, foi utilizada a abordagem GQM – Goal/Question/Metric (KOZIOLEK, 2008). O uso dessa abordagem auxilia na definição dos objetivos de medição do trabalho e também na derivação das medidas a serem coletadas posteriormente durante o estudo de caso.

A abordagem GQM consiste na definição de um objetivo, que então será derivado em questões e cada questão resultará em uma ou mais medidas, que podem ou não estar relacionadas entre si. Ao final da medição espera-se que os dados coletados durante a fase de medição possam ser interpretados e então gerem uma resposta para a questão definida, assim elucidando se o objetivo foi atingido.

Figura 12 – Paradigma GQM



Fonte: (KOZIOLEK, 2008)

- **Objetivo de medição 1:** Avaliar o esforço despendido pelas equipes participantes para a seleção, coleta e análise das métricas definidas no estudo de caso.
- **Objetivo de medição 2:** Avaliar o acompanhamento dos planos de ação.
- **Objetivo de medição 3:** Avaliar o custo-benefício da implementação de métricas aos planos de ação.

5.2 PERGUNTAS E MEDIDAS

Como mencionado na pergunta de pesquisa, o estudo de caso pretende observar o impacto da aplicação de métricas ágeis para o acompanhamento de planos de ação definido pelo time. A coleta de dados de medição será realizada por meio da aplicação de questionário, utilizando a ferramenta *Google Forms*¹.

Tabela 4 – Objetivo de Medição 1

Objetivo de Medição 1	Avaliar o esforço despendido pelas equipes participantes para a seleção, coleta e análise das métricas definidas no estudo de caso.
Pergunta 1.1	Qual o esforço total despendido pelas equipes para a seleção, coleta e análise das métricas?
Medição 1.1	Esforço total em horas

¹ <https://docs.google.com/forms>

Tabela 5 – Objetivo de Medição 2

Objetivo de Medição 2	Avaliar o acompanhamento dos planos de ação.
Pergunta 2.1	Considero que o uso de métricas tornou mais claro a evolução do plano de ação.
Medição 2.1	Escala Likert de 1 a 5, sendo 1 discordo totalmente e 5 concordo totalmente.
Pergunta 2.2	Você gostaria de comentar algo sobre o uso de métricas ágeis aplicadas aos planos de ação?
Medição 2.2	Impressão subjetiva da experiência com a aplicação de métricas ágeis para o acompanhamento dos planos de ação.

Tabela 6 – Objetivo de Medição 3

Objetivo de Medição 3	Avaliar o custo-benefício da implementação de métricas aos planos de ação.
Pergunta 3.1	Considero que o uso de métricas aos planos de ação da roda ágil possui um bom custo-benefício.
Medição 3.1	Escala Likert de 1 a 5, sendo 1 discordo totalmente e 5 concordo totalmente.
Pergunta 3.2	Você gostaria de comentar algo sobre custo-benefício do uso de métricas ágeis aplicadas aos planos de ação?
Medição 3.2	Impressão subjetiva da experiência com a aplicação de métricas ágeis para o acompanhamento dos planos de ação.

5.3 PLANEJAMENTO DA COLETA

Para a coleta dos dados quantitativos de horas foi utilizado o google agenda² para somar o tempo de reuniões registrados na ferramenta, o restante dos dados foram coletados através de um formulário contendo as perguntas definidas anteriormente.

Tabela 7 – Procedimento de coleta de medidas

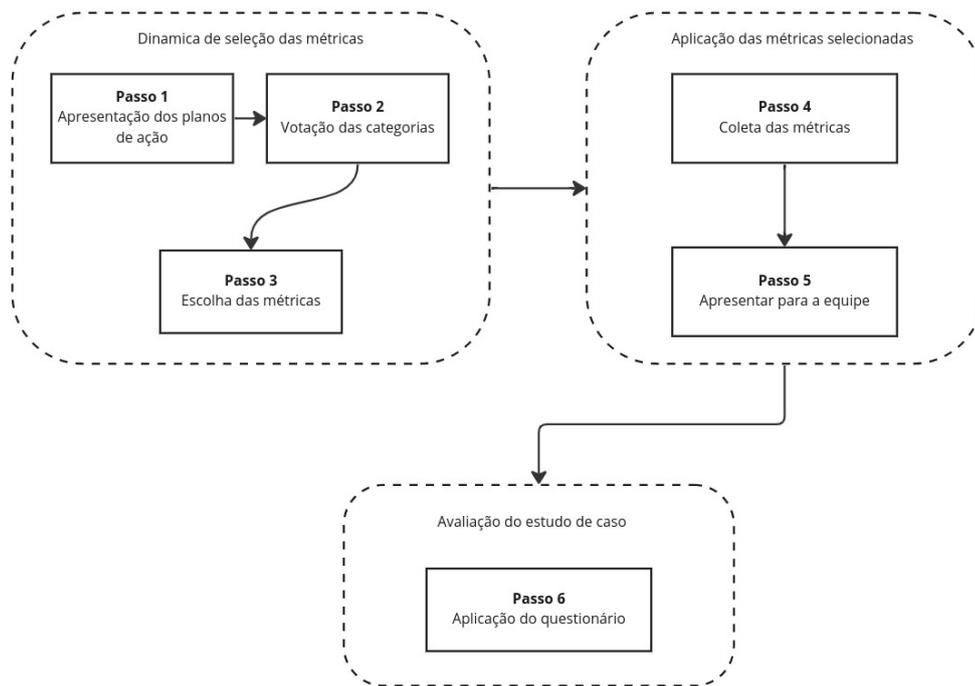
Medida	Algoritmo	Forma de obtenção
Medida 1.1	Esforço total em horas	<i>Google Agenda</i>
Medida 2.1	Pergunta de questionário	<i>Google Forms</i>
Medida 2.2	Pergunta de questionário	<i>Google Forms</i>
Medida 3.1	Pergunta de questionário	<i>Google Forms</i>
Medida 3.2	Pergunta de questionário	<i>Google Forms</i>

² <https://calendar.google.com/>

6 EXECUÇÃO DO ESTUDO DE CASO

Neste capítulo é apresentada a execução do estudo de caso, conforme descrito no capítulo 5. O estudo de caso foi dividido entre seleção de métricas e aplicação das métricas selecionadas e foi executado em duas equipes do Laboratório Bridge. A figura 13 detalha o passo a passo das atividades realizadas durante o estudo de caso.

Figura 13 – Etapas do estudo de caso



miro

Fonte: Autora - 2022

A etapa de seleção utilizou a lista de métricas presentes no artigo "How Agile Organizations Use Metrics: A Systematic Literature Mapping" (LEAL; VIEIRA; BERTAN, 2022), o artigo foi escrito a partir da revisão sistemática da literatura deste trabalho, feita em conjunto com o Grupo de Qualidade de Software da Universidade Federal de Santa Catarina. As métricas foram extraídas dos artigos presentes na revisão bibliográfica e então categorizadas com base nas sete categorias definidas por McGarry *et al.* (2001), sendo elas:

- Satisfação do cliente;
- Efetividade tecnológica;
- Tamanho do produto e estabilidade;
- Planejamento e progresso;

- Qualidade do produto;
- Performance do processo;
- Recursos e custos.

O estudo de caso foi aplicado no período de 10 de outubro á 25 de novembro de 2022 em duas equipes do Laboratório Bridge que atual na unidade organizacional e-SUS APS. A coleta de métricas ocorreu durante duas sprints que compreendem o período de 24 de outubro á 6 de novembro de 2022 e 7 á 21 de novembro de 2022.

6.1 SELEÇÃO DE MÉTRICAS ÁGEIS

Para a definição de quais métricas seriam aplicadas a cada um dos planos de ação, previamente definidos pelas equipes participantes, foi realizada uma dinâmica dividida em dois encontros.

No primeiro encontro, envolvendo o time, cada um dos planos de ação foi lido e a motivação que levou a criação do plano de ação foi lembrada, e também foram discutidos os objetivos que a equipe pretende atingir ao finaliza-lo. Em seguida, levando em consideração o plano de ação criado e as motivações, os membros da equipe utilizaram da ferramenta Padlet¹ para votar em quais categorias o plano de ação em pauta poderia corresponder. Inicialmente a proposta era um voto por membro do time, mas durante a reunião com o primeiro time foi observado que cada objetivo poderia ser classificado em mais de uma categoria, então cada membro da equipe podem votar em uma ou mais categorias, ao final, foi realizado um momento de discussão em que os membros que se sentisse a vontade poderiam expor a categoria que votaram e justificar seu entendimento.

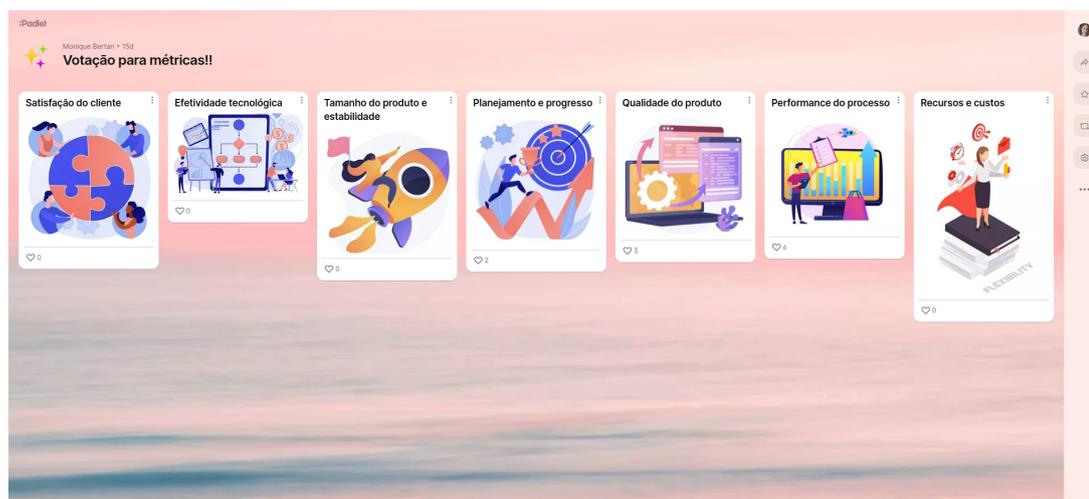
Durante o segundo encontro foi apresentado para cada plano de ação as métricas correspondentes às três categorias mais votadas. A equipe então discutiu e decidiu em conjunto qual métrica ou conjunto de métricas era mais interessante para atingir o objetivo do respectivo plano de ação. Em ambas as equipes foi dado preferência a métricas que pudessem ser automatizadas ou que as medições necessárias já eram realizadas automaticamente.

6.1.1 Equipe F

A seleção de métricas ocorreu entre 17 a 21 de outubro de 2022. Previamente a equipe já havia definido quatro planos de ação, descritos na tabela 8.

¹ <https://padlet.com>

Figura 14 – Painel Padlet



Fonte: Autora - 2022

Tabela 8 – Planos de ação equipe F

ID	Categoria roda ágil	Plano de ação	Motivação
PA01	Qualidade E2E	Rodar o Sonar em todas as histórias finalizadas buscando falhas	A equipe julga que falta testar cenários mais próximos do uso real, como testes de carga, e testes dedicados a segurança.
PA02	Ritmo das sprints	Ter 80% das sprints com aproveitamento maior que 70%	A equipe sente que as issues estão atrasando pois não são acompanhadas, issues sobram no board ao fim da sprint ou faltam issues planejadas durante a sprint.
PA03	Issues e épicos	Ter 70% das issues refinadas	As descrições de issues muito genéricas, algumas vezes faltam critérios de aceitação.
PA04	Colaboração e comunicação	Fazer repasses de reuniões e informações importantes que afetam a equipe de forma estruturada no canal do Slack	Alguns membros que trabalham remoto sentem dificuldade de comunicação e integração com membros do presencial, as vezes há demora para obter respostas e issues ficam bloqueadas esperando definição.

Conforme apresentado, durante o primeiro encontro ocorreu a votação para escolher as categorias correspondentes aos planos de ação, o resultado está descrito na tabela 9.

Tabela 9 – Categorias selecionadas equipe F

ID	Categoria Escolhida	Quantidade de votos
PA01	Qualidade de produto	5 votos
	Recursos e custos	1 votos
	Satisfação do cliente	1 voto
PA02	Planejamento e progresso	5 votos
	Performance do processo	3 votos
PA03	Planejamento e progresso	5 votos
	Recursos e custos	2 votos
	Qualidade do produto	2 votos
PA04	Performance e progresso	6 votos
	Qualidade do produto	3 votos
	Planejamento e progresso	2 votos

Durante a votação, um integrante da equipe considerou que o PA01 pertencia a categoria recursos e custos, devido ao custo e impacto que são despendidos sempre que um bug grave são encontrados em ambiente de produção. Outro integrante da equipe pontuou que a satisfação do cliente também é afetada com o alto índice de bugs encontrados em ambiente de produção, por esse motivo considerou que o PA01 pode ser categorizado também na categoria de satisfação do cliente.

Com relação ao PA03, a equipe como um todo concordou que descrições genéricas das issues e a falta de critérios de aceitação impactam nas categorias planejamento e progresso, custos e recursos e qualidade do produto.

Não houveram comentários sobre a escolha de categorias para o PA02 e PA04.

Após a votação das categorias, em um segundo momento, a equipe se reuniu para escolher uma métrica ou conjunto de métricas para cada plano de ação. Para o PA01 a métrica escolhida foi **número de defeitos**, ao final da medição a equipe deseja comparar o numero de defeitos encontrados após a implementação do SonarQube² com dados históricos de defeitos encontrados em produção para concluir se houve uma diminuição da quantidade de defeitos encontrados em produção após a implementação do SonarQube.

² <https://www.sonarqube.org/>

Tabela 10 – Descrição da métrica número de defeitos

Número de defeitos	
Nome original	Number of defects
Categoria	Qualidade do produto
Necessidade de informação	Cocorrência de falhas
Conceito mensurável	Qualidade do produto
Objetivo da medição	Avaliar a quantidade de defeitos encontrados em um período de tempo
Entidades e atributos	Número de defeitos encontrados Período de tempo
Função de medição	Somar o número de issues corretivas abertas em um período de tempo
Fonte	S04

Para o PA02 foi selecionada a métrica **taxa conclusão de histórias planejadas**. A equipe já utiliza uma métrica similar para reportar o aproveitamento da sprint quinzenalmente para o restante da organização, o objetivo é acompanhar a precisão do planejamento ao longo das sprints e tornar mais visível se mudanças no processo refletem positivamente ou negativamente no aproveitamento da sprint.

Tabela 11 – Descrição da métrica taxa de conclusão de histórias planejadas

Taxa de conclusão de histórias planejadas	
Nome original	Planned stories completion ratio
Categoria	Planejamento e progresso
Necessidade de informação	Progresso do projeto
Conceito mensurável	Produtividade do projeto
Objetivo da medição	Avaliar a porcentagem de histórias planejadas que foram efetivamente concluídas no período da sprint
Entidades e atributos	Número de histórias de usuário concluídas Número de histórias de usuário planejadas
Função de medição	Somatório da pontuação das histórias concluídas/somatório de pontuação das histórias planejadas
Fonte	S27

Para o PA03 a equipe escolheu a métrica **comparação de fluxo de tarefas concluídas**, a métrica consiste em calcular a porcentagem de issues bloqueadas dentre as issues planejadas para a sprint, o objetivo da equipe é manter as issues melhores descritas e definidas antes do início da sprint e assim alcançar a porcentagem abaixo de 30% de issues bloqueadas.

Tabela 12 – Descrição da métrica comparação de fluxo de tarefas concluídas

Comparação de fluxo de tarefas concluídas	
Nome original	Task finished flow comparison
Categoria	Planejamento e progresso
Necessidade de informação	Capacidade de planejamento da equipe
Conceito mensurável	issues bloqueadas
Objetivo da medição	Avaliar a porcentagem de issues bloqueadas durante uma sprint
Entidades e atributos	Número de issues bloqueadas durante a sprint Número de issues planejadas durante a sprint
Função de medição	Número de issues bloqueadas/número de issues planejadas
Fonte	S29

Com relação ao PA04, a equipe não encontrou uma métrica dentre as selecionadas da literatura que cumprisse o propósito da medição, visto que o objetivo da equipe

é melhorar a comunicação e relacionamento dos membros da equipe, um conceito bastante subjetivo para ser medido.

6.1.2 Equipe S

A seleção de métricas na equipe S ocorreu entre os dias 17 à 21 de outubro de 2022, os planos de ação anteriormente definidos pela equipe estão descritos na tabela 13.

Tabela 13 – Planos de ação equipe S

ID	Categoria roda ágil	Plano de ação	Motivação
PA01	Testes automatizados	Compreender como as equipes estão aplicando TDD dentro do laboratório	A equipe pretende observar e estudar a aplicação de TDD dentro do laboratório para avaliar se a técnica pode trazer benefícios para o seu contexto.
PA02	Métricas	Criar um gráfico de Breakdown	Identificar e compreender os impedimentos que atrapalham a fluidez da equipe e possíveis gargalos no processo de desenvolvimento.
PA03	Métricas	Criar um gráfico de Burndown	Acompanhar o progresso da sprint em andamento, afim de identificar issues atrasadas e tomar ações antecipadas.

Durante o primeiro encontro ocorreu a votação para escolher as categorias correspondentes aos planos de ação, o resultado pode ser observado na tabela 14.

Tabela 14 – Categorias selecionas equipe S

ID	Categoria Escolhida	Quantidade de votos
PA01	Qualidade de produto	7 votos
	Efetividade tecnológica	2 votos
PA02	Planejamento e progresso	6 votos
	Performance do processo	5 votos
	Tamanho do produto e estabilidade	2 votos
PA03	Planejamento e progresso	7 votos
	Performance do processo	4 votos
	Tamanho do produto e estabilidade	1 voto

Dois integrantes do time pontuaram que por TDD se tratar de uma técnica de

desenvolvimento, o PA01 poderia ser classificado também na categoria efetividade tecnológica. Um integrante considerou que o atraso das issues, motivação do PA03, pode afetar a estabilidade do produto, por esse motivo votou na categoria tamanho do produto e estabilidade. Não houveram comentários sobre a votação da categoria para o PA02.

Em um encontro posterior, as métricas das categorias selecionadas foram apresentadas para a equipe e uma métrica ou conjunto de métricas foi escolhido para o acompanhamento de cada um dos planos de ação.

A equipe não conseguiu definir uma métrica para o PA01, visto que o objetivo do plano de ação definido é estudar a aplicação da técnica TDD dentro da organização, dentre as métricas selecionadas na revisão sistemática não havia nenhuma que cumprisse o objetivo da equipe.

Para o PA02 o objetivo da equipe é identificar e compreender os impedimentos que atrapalham a fluidez da equipe e possíveis gargalos no processo de desenvolvimento, para isso o time escolheu as métricas apresentadas no **diagrama de fluxo cumulativo**.

Tabela 15 – Descrição da métrica diagrama de fluxo acumulado

Diagrama de fluxo cumulativo	
Nome original	Cumulative Flow Diagram
Categoria	Qualidade do produto
Necessidade de informação	Progresso do projeto
Conceito mensurável	Produtividade do projeto
Objetivo da medição	Avaliar o andamento das entradas e saídas ao longo do processo ou suas partes
Entidades e atributos	Tarefas do projeto Data em que cada tarefa mudou de status
Função de medição	O CFD é uma representação gráfica. de uma curva decrescente que a auxilia o acompanhamento do trabalho realizado, que mostra as alterações no status das tarefas a serem realizadas ao longo do tempo, sendo que o eixo horizontal representa o tempo percorrido, enquanto o eixo vertical apresenta o número de features ou histórias trabalhadas dentro de cada um dos estágios de um item de trabalho, ou seja, no gráfico podemos ver diversas curvas plotadas, inclusive por vezes se sobrepondo, mostrando, por exemplo o trabalho em backlog, em desenvolvimento, em testes e completo.
Fonte	S02

A equipe havia definido o PA03 como "Criar um gráfico de Burndown", com o objetivo de identificar melhor o progresso da sprint, foi então revisado as outras métricas disponíveis e escolhido a métrica apresentada no gráfico de **burnup**, muito parecida com o burndown, porém acrescentando ao gráfico uma linha com a projeção do escopo e com o gráfico ascendente, enquanto o o burndown possui linhas descendentes.

Tabela 16 – descrição da métrica burnup

Burnup	
Nome original	Burnup
Categoria	Performance do processo
Necessidade de informação	Progresso do projeto
Conceito mensurável	Produtividade do projeto
Objetivo da medição	Avaliar a o andamento da equipe durante um período de tempo
Entidades e atributos	Número de histórias de usuárona coluna de backlog do board ao final de cada dia da sprint Número de histórias de usuárona coluna de done do board ao final de cada dia da sprint Data de início e fim da sprint
Função de medição a estimativa de pontos planejadas e realizadas, buscando o valor 0 como diferença, contudo, por não se tratar na essência de um cálculo e sim de uma representação gráfica, o burnup ocorre ao plotar os itens de backlog pontuados a serem realizados no eixo Y, bem como os itens realizados e os dias percorridos no eixo X.	Pode-se obter o resultado do burnup comparando
Fonte	

6.2 APLICAÇÃO DAS MÉTRICAS SELECIONADAS

Esta etapa foi realizada entre os dias 24 de outubro à 20 de novembro de 2022, durante o período ocorreram duas sprints de duas semanas cada, sendo a primeira entre os dias 24 de outubro à 6 de novembro e a segunda sprint compreendeu o período entre os dias 7 à 20 de novembro de 2022. A autora e os respectivos scrum masters de cada equipe se responsabilizaram de coletar as métricas selecionadas na seção anterior, sendo que cada scrum master ficou responsável por apresentar para o time o resultado das métricas ao menos uma vez durante a sprint.

6.2.1 Equipe F

As métricas selecionadas da etapa anterior foram Número de defeitos, Taxa de conclusão de histórias planejadas e Comparação de fluxo de tarefas concluídas.

A motivação para a escolha da métrica foi comparar o número e defeitos encontrados utilizando os testes já executados pela equipe versus os defeitos encontrados após rodar a ferramenta SonarQube. Infelizmente durante o período do estudo de caso a equipe não conseguiu finalizar o estudo e configuração da ferramenta. Os dados da métrica foram coletados, mas não foi possível fazer o comparativo esperado.

Tabela 17 – Número de defeitos - Equipe F

Número de defeitos	
Sprint 1	0 Issues
Sprint 2	2 Issues

A métrica "taxa de conclusão de histórias planejadas" foi selecionada com o objetivo de tornar mais visível o aproveitamento da equipe, visto que o plano de ação definido era "Ter 80% das sprints com aproveitamento maior que 70%.", pode-se notar que durante a primeira sprint do estudo de caso o objetivo foi atingido, enquanto na segunda sprint não.

Tabela 18 – Taxa de conclusão de histórias planejadas- Equipe F

Taxa de conclusão de histórias planejadas	
Sprint 1	88%
Sprint 2	51%

O PA03 foi descrito pela equipe como "Ter 70% das issues refinadas", para o acompanhamento do plano de ação foi escolhido a métrica comparação de fluxo de tarefas concluídas, que calcula a taxa de tarefas bloqueadas durante a sprint, o objetivo da equipe foi manter a métrica abaixo de 30%, em nenhuma das duas sprints a meta foi atingida.

Tabela 19 – Comparação de fluxo de tarefas concluídas - Equipe F

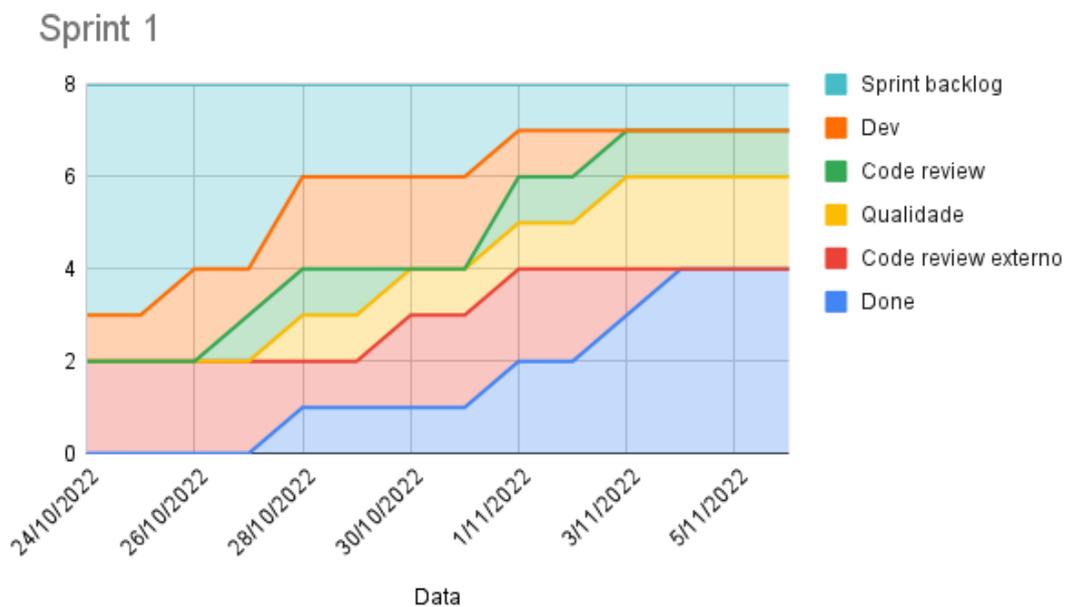
Comparação de fluxo de tarefas concluídas	
Sprint 1	33%
Sprint 2	40%

6.2.2 Equipe S

As métricas selecionadas da etapa anterior foram dos diagramas de Fluxo Cumulativo e de *Burnup*.

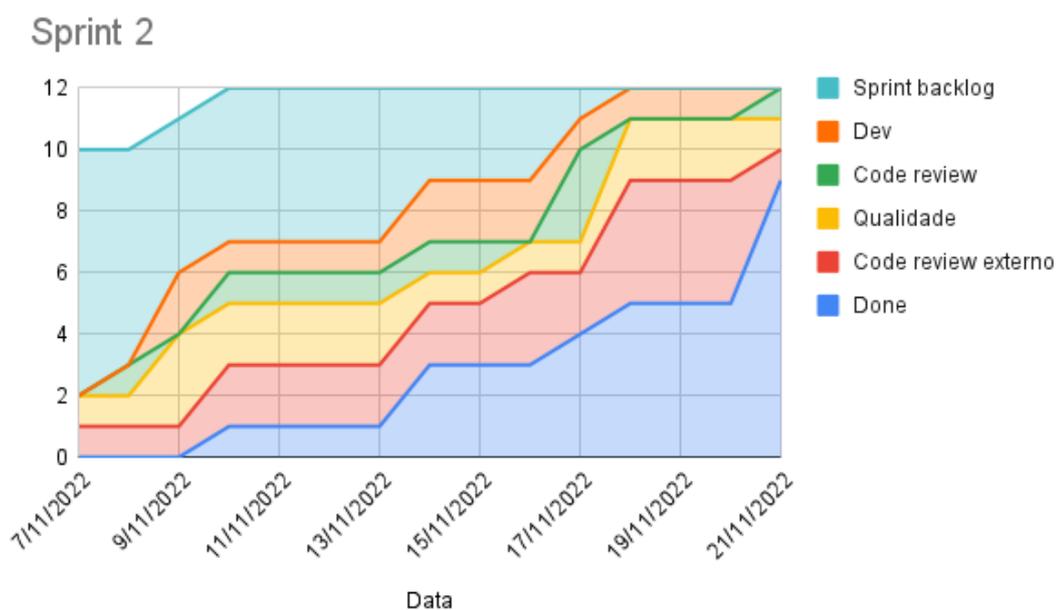
Observando o diagrama de fluxo cumulativo é possível notar que durante a segunda sprint a altura das faixas permanece mais regular, aumentando significativamente apenas a faixa de code review externo a partir do dia 18/11/2022 ao contrário da primeira sprint em que todas as colunas houveram momentos de ociosidade ao longo da sprint.

Figura 15 – CFD - Sprint 1



Fonte: Autora - 2022

Figura 16 – CFD - Sprint 2



Fonte: Autora - 2022

Sobre o gráfico de *burnup*, pode-se concluir que durante as duas sprints a linha de trabalho realizado ficou a todo momento abaixo da linha de trabalho esperado, sendo assim, pode-se inferir ao longo da sprint que, se o *throughput* da equipe não

aumentar e transpassar a linha vermelha o trabalho planejado no início da sprint não será finalizado em sua totalidade ao final da sprint.

Figura 17 – Burnup - Sprint 1



Fonte: Autora - 2022

Figura 18 – Burnup - Sprint 2



Fonte: Autora - 2022

6.3 COLETA E ANÁLISE DOS RESULTADOS

Ao final do estudo de caso foi aplicado um questionário com as equipes participantes para avaliação do estudo, ao total foram obtidas 10 respostas, 6 respostas da equipe S, equivalente a 85% da equipe, e 4 respostas da equipe F, equivalente a 66% da equipe.

Objetivo de medição 1: Avaliar o esforço despendido pelas equipes participantes para a seleção, coleta e análise das métricas definidas no estudo de caso.

Foram necessárias duas reuniões por equipe, de aproximadamente 30 min cada, para realizar a dinâmica de seleção de métricas. A coleta foi realizada pela autora do tcc, em conjunto com os SMs de cada uma das equipes, sendo que para coletar as métricas da equipe F foi necessário 30 min por sprint, visto que a equipe já possui uma planilha contendo quase todos os dados necessários. Para coletar as métricas da equipe S foi necessário cerca de uma hora a cada sprint, pois a equipe ainda não possuía os dados necessários sendo coletados automaticamente. A análise das métricas foi feita semanalmente, durante as reuniões de daily scrum das equipes, o tempo despendido especificamente na análise não foi coletado.

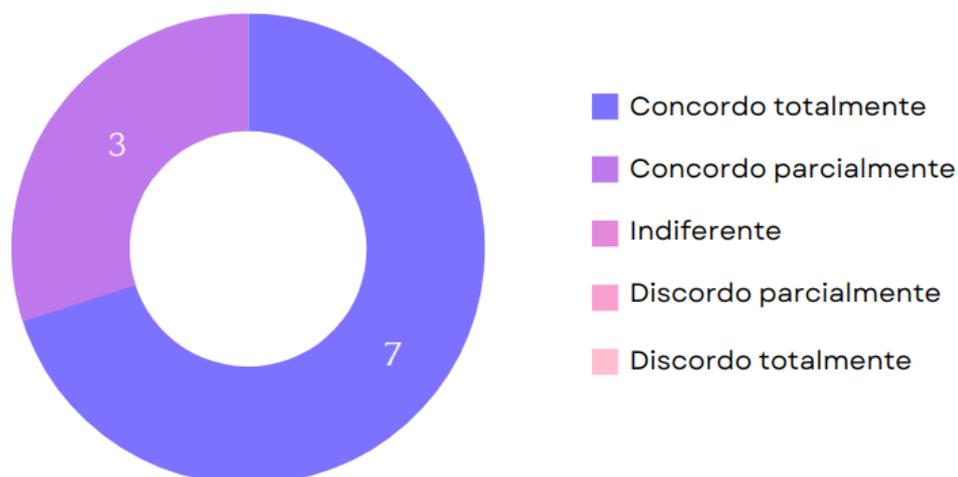
Objetivo de medição 2: Avaliar o acompanhamento dos planos de ação.

70% dos respondentes concordam totalmente com a afirmação "O uso de métricas tornou mais claro a evolução do plano de ação", figura 19. Dois integrantes pontuaram que o uso das métricas tornou mais fácil de compreender os objetivos dos planos de ação e como alcança-los.

Durante umas das reuniões com as equipes foi observada dificuldade por parte de um dos integrantes do time de interpretar o diagrama de fluxo cumulativo devido a condição de daltonismo do integrante.

Figura 19 – Pergunta 1

CONSIDERO QUE O USO DE MÉTRICAS TORNOU MAIS CLARO
A EVOLUÇÃO DO PLANO DE AÇÃO



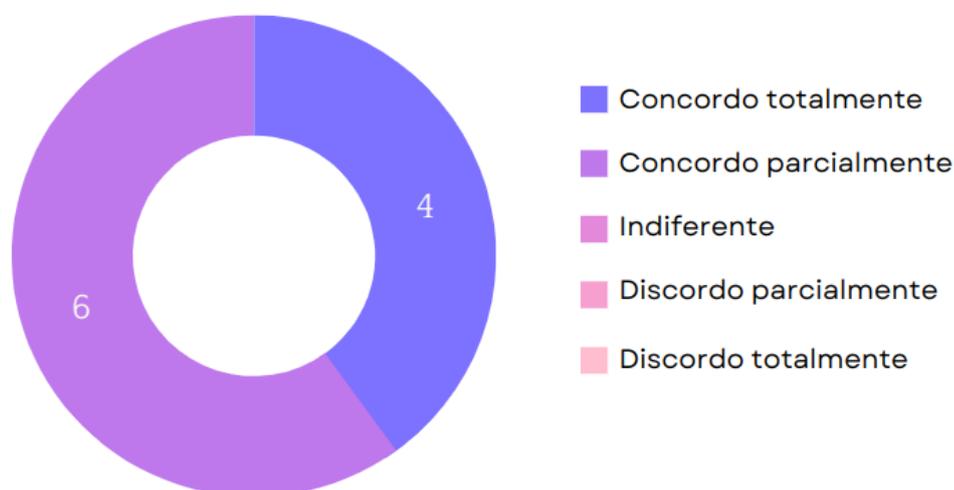
Fonte: Autora - 2022

Objetivo de medição 3: Avaliar o custo-benefício da implementação de métricas aos planos de ação.

60% dos respondentes concordam totalmente com a afirmação "Considero que o uso de métricas aos planos de ação da roda ágil possui um bom custo-benefício", 40% dos respondentes concordaram parcialmente com a afirmação, 20. Um integrante pontuou que caso a dinâmica para a escolha de métricas fosse mais estruturada o acompanhamento durante um ciclo completo da roda ágil faz com que a implementação tenha um ótimo custo-benefício.

Figura 20 – Pergunta 2

CONSIDERO QUE O USO DE MÉTRICAS AOS PLANOS DE AÇÃO DA RODA ÁGIL POSSUI UM BOM CUSTO-BENEFÍCIO



Fonte: Autora - 2022

6.4 DISCUSSÃO

A partir dos resultados obtidos pode-se observar que em sua maioria as equipes julgaram como benéfico a aplicação de métricas aos planos de ação. A aplicação das métricas propiciou aos membros uma visão mais clara da evolução do objetivo proposto em cada plano de ação. Contudo, observou-se que a dinâmica de escolha de métricas demandou um esforço considerado excessivo pelas equipes, que preferiam uma dinâmica assíncrona.

Houve dificuldade para interpretar os gráficos da equipe S no primeiro momento, em especial o CFD que exigiu uma explicação mais detalhada para todos os membros da equipe que nunca haviam utilizado esta métrica anteriormente. A equipe F não encontrou dificuldade para interpretar as métricas selecionadas. Além disso, o tempo de estudo foi considerado curto pelos participantes, que consideraram que é necessário acompanhar as métricas por pelo menos três meses, um ciclo da roda ágil, para avaliar o real ganho frente ao tempo despendido para a seleção das métricas e coleta de dados.

6.5 AMEAÇAS A VALIDADE

A pesquisa possui algumas ameaças à sua validade devido a alguns pontos na coleta e avaliação dos resultados o estudo de caso, o que pode representar ameaça à

validade do estudo quanto à tentativa de generalização dos resultados observados.

Primeiramente, o estudo de caso foi aplicado em apenas uma organização, além disso, nem todos os participantes do estudo de caso responderam os questionários enviados.

Outro ponto a ser considerado foi a falta de descrição detalhada da totalidade de métricas encontradas na literatura, apenas uma parte das métricas encontradas na literatura possuíam descrição de implementação, o estudo se limitou a considerar apenas esta parcela de métricas.

Por fim, pode-se considerar também como ameaça o envolvimento direto da autora, que fazia parte de uma das equipes envolvidas no estudo de caso, portanto pode haver certo viés na interpretação dos resultados obtidos.

7 CONCLUSÃO

Este trabalho descreve a seleção e aplicação de métricas ágeis em um ambiente organizacional de desenvolvimento de software através de um estudo de caso aplicado no projeto eSUS-APS do laboratório Bridge da Universidade Federal de Santa Catarina.

O trabalho iniciou com um estudo na literatura dos conceitos abordados ao longo do trabalho, como medição e métodos ágeis, em seguida foi realizada uma análise do estado da arte, na forma de uma revisão sistemática da literatura, que deu origem ao artigo "How Agile Organizations Use Metrics: A Systematic Literature Mapping"(LEAL; VIEIRA; BERTAN, 2022).

Com base nos artigos encontrados durante a análise do estado da arte, foi realizado o diagnóstico do estado atual da organização Laboratório Bridge e comparado o processo executado no laboratório com os resultados encontrados na literatura.

A partir da análise do estado atual, oportunidades de melhoria foram levantadas e houve então o planejamento do estudo de caso e definição dos objetivos do estudo de caso.

Com o intuito de obter uma análise sobre a seleção e aplicação de métricas ágeis, foi realizado um estudo de caso com duas equipes da organização, que aplicaram uma dinâmica de seleção de métricas e em seguida acompanharam as métricas selecionadas pelo período de duas sprints, aproximadamente um mês.

Como resultado do estudo de caso pode-se concluir que todos os participantes do estudo de caso que responderam o formulário de avaliação consideraram o custo-benefício bom ou ótimo, entretanto, a dinâmica de seleção de métricas deve ser melhorada, afim de despende menos tempo das equipes e tornar-se um processo mais estruturado, idealmente assíncrono.

Como trabalhos futuros sugere-se o acompanhamento das métricas selecionadas por um período maior de tempo e em outras organizações com produtos de outras áreas, diferentes de saúde. Além disso, o processo de seleção de métricas pode ser melhor estruturado, gerando um framework para seleção de métricas em diferentes contextos.

REFERÊNCIAS

- ALPEROWITZ, Lukas; DZVONYAR, Dora; BRUEGGE, Bernd. Metrics in agile project courses. *In: PROCEEDINGS of the 38th International Conference on Software Engineering Companion*. [S.l.: s.n.], 2016. P. 323–326. Citado na p. 14.
- ANDERSON, David J; CARMICHAEL, Andy. **Essential kanban condensed**. [S.l.]: Blue Hole Press, 2016. Citado na p. 29.
- BECK, Kent *et al.* **Manifesto for Agile Software Development**. [S.l.: s.n.], 2001. Disponível em: <<https://agilemanifesto.org/>>. Acesso em: 7 nov. 2021. Citado nas pp. 14, 24, 36.
- BOEG, Jesper. Kanban em 10 passos. **Tradução de Leonardo Campos, Marcelo Costa, Lúcio Camilo, Rafael Buzon, Paulo Rebelo, Eric Fer, Ivo La Puma, Leonardo Galvão, Thiago Vespa, Manoel Pimentel e Daniel Wildt**. C4Media, 2010. Citado na p. 29.
- BOERMAN, Martin P *et al.* Measuring and monitoring agile development status. *In: IEEE. 2015 IEEE/ACM 6th International Workshop on Emerging Trends in Software Metrics*. [S.l.: s.n.], 2015. P. 54–62. Citado na p. 14.
- BRIDGE, Laboratório. **Quem Somos**. [S.l.: s.n.], 2016. Disponível em: <<https://bridge.ufsc.br/quem-somos>>. Acesso em: 21 mar. 2021. Citado na p. 14.
- BRIDGE, Laboratório. **Quem Somos**. [S.l.: s.n.], 2022. Disponível em: <<https://portal.bridge.ufsc.br/quem-somos/>>. Acesso em: 1 mar. 2022. Citado na p. 47.
- CARDOZO, Luiz Fernando *et al.* UM GUIA PARA SELEÇÃO DE MÉTRICAS ÁGEIS DE GERENCIAMENTO DE PROJETOS PARA ORGANIZAÇÕES DE DESENVOLVIMENTO DE SOFTWARE. Florianópolis, SC, 2020. Citado na p. 14.
- COHEN, David; LINDVALL, Mikael; COSTA, Patricia. An introduction to agile methods. **Adv. Comput.**, v. 62, n. 03, p. 1–66, 2004. Citado na p. 14.
- DENNING, Peter J; GUNDERSON, Chris; HAYES-ROTH, Rick. The profession of IT Evolutionary system development. **Communications of the ACM**, ACM New York, NY, USA, v. 51, n. 12, p. 29–31, 2008. Citado na p. 31.

ISO/IEC/IEEE International Standard - Systems and software engineering—Measurement process. **ISO/IEC/IEEE 15939:2017(E)**, p. 1–49, 2017. DOI: 10.1109/IEEESTD.2017.7907158. Citado nas pp. 18, 20–23.

KEELE, Staffs *et al.* **Guidelines for performing systematic literature reviews in software engineering**. [S.l.], 2007. Citado nas pp. 36, 38.

KOZIOLEK, Heiko. Goal, Question, Metric. *In: Dependability Metrics: Advanced Lectures*. Edição: Irene Eusgeld, Felix C. Freiling e Ralf Reussner. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. P. 39–42. ISBN 978-3-540-68947-8. DOI: 10.1007/978-3-540-68947-8_6. Disponível em: <https://doi.org/10.1007/978-3-540-68947-8_6>. Citado nas pp. 51, 52.

LARMAN, Craig; VODDE, Bas. **Large-scale scrum: More with LeSS**. [S.l.]: Addison-Wesley Professional, 2016. Citado nas pp. 32, 33.

LEAL, S; VIEIRA, G; BERTAN, M. How Agile Organizations Use Metrics: A Systematic Literature Mapping. **XXI Simpósio Brasileiro de Qualidade de Software**, 2022. Citado nas pp. 14, 54, 69.

LEFFINGWELL, Dean. **SAFe 4.5 reference guide: scaled agile framework for lean enterprises**. [S.l.]: Addison-Wesley Professional, 2018. Citado nas pp. 33–35.

LEFFINGWELL, Dean. **Scaling software agility: best practices for large enterprises**. [S.l.]: Pearson Education, 2007. Citado na p. 31.

MCGARRY, John *et al.* **Practical software measurement: objective information for decision makers**. [S.l.]: Addison-Wesley Professional, 2001. Citado nas pp. 18, 19, 21–23, 54.

MOORE, Erik; SPENS, John. Scaling agile: Finding your agile tribe. *In: IEEE. AGILE 2008 Conference*. [S.l.: s.n.], 2008. P. 121–124. Citado na p. 31.

PADMINI, KV Jeeva; BANDARA, HMN Dilum; PERERA, Indika. Use of software metrics in agile software development process. *In: IEEE. 2015 Moratuwa Engineering Research Conference (MERCon)*. [S.l.: s.n.], 2015. P. 312–317. Citado na p. 14.

POPPENDIECK, M.; POPPENDIECK, T.; POPPENDIECK, T.D. **Lean Software Development: An Agile Toolkit**. [S.l.]: Addison-Wesley, 2003. (Agile software

development series). ISBN 9780321150783. Disponível em:
<<https://books.google.com.br/books?id=Ua1KAgAAQBAJ>>. Citado na p. 31.

PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software-8ª Edição**. [S.l.]: McGraw Hill Brasil, 2016. Citado na p. 29.

RUNESON, Per; HÖST, Martin. Guidelines for conducting and reporting case study research in software engineering. **Empirical software engineering**, Springer, v. 14, n. 2, p. 131–164, 2009. Citado na p. 51.

SCRUM.ORG. **The 2020 Scrum Guide**. [S.l.: s.n.], 2020. Disponível em:
<<https://scrumguides.org/scrum-guide.html>>. Acesso em: 7 nov. 2021. Citado nas pp. 25–28.

SILVA LEAL, Stéphanie da. **Em construção**. Florianópolis, SC: [s.n.], Em construção. Citado nas pp. 39, 40.

SOFTEX, ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – **MPS.BR – Guia de Implementação de Software – Parte 2**. [S.l.: s.n.], 2016. Disponível em: <www.softex.br>. Acesso em: 30 out. 2021. Citado nas pp. 18, 19.

SOMMERVILLE, Ian. **Engenharia de Software**. [S.l.]: Pearson Education do Brasil, 2011. Citado nas pp. 29–32.

VERSIONONE. **14th Annual State of Agile Report**. [S.l.: s.n.], 2020. Disponível em:
<<https://stateofagile.com/#>>. Acesso em: 4 mai. 2021. Citado nas pp. 14, 25.

VERSIONONE. **15th Annual State of Agile Report**. [S.l.: s.n.], 2021. Disponível em:
<<https://stateofagile.com/#>>. Acesso em: 13 mar. 2022. Citado nas pp. 36, 37.

WAZLAWICK, Raul. **Engenharia de software: conceitos e práticas**. [S.l.]: Elsevier Editora Ltda., 2019. Citado nas pp. 18, 25–31.

WOHLIN, Claes. Guidelines for snowballing in systematic literature studies and a replication in software engineering. *In*: PROCEEDINGS of the 18th international conference on evaluation and assessment in software engineering. [S.l.: s.n.], 2014. P. 1–10. Citado na p. 37.

Apêndices

APÊNDICE A – ARTIGO

How Agile Organizations Use Metrics: A Systematic Literature Mapping

Author 1

Department Name
Institution/University Name
City State Country
email@email.com

Author 2

Department Name
Institution/University Name
City State Country
email@email.com

Author 3, Author 4

Department Name
Institution/University Name
City State Country
email@email.com

ABSTRACT

Agile Software Development Methods (ASDMs) have addressed the complexity of managing software development processes through small increments, continuous and frequent delivery, openness to change, and work visibility. These practices have been complemented by the use metrics adapted to ASDMs context, generating a large number and types of metrics. Thus, this secondary study presents a Systematic Literature Mapping to understand how agile organizations use metrics. Seven research questions were defined, and 33 primary studies related to the use of agile metrics were selected. The results indicate that most organizations that use agile metrics use Scrum, develop large size general-purpose software products, desktop, or web applications, in small teams. The metrics are selected by primary studies based on literature search, interviews with the team members or expert groups. Most studies defined Technology, Schedule & Progress or Product metrics. The primary studies results indicate that using agile metrics can positively impact the quality of processes, products, and services.

CCS CONCEPTS

• Insert CCS text here • Insert CCS text here • Insert CCS text here

KEYWORDS

Metrics; Agile Software Development; Systematic Literature Mapping (SLM)

1 Introduction

Agile Software Development Methods (ASDM) have the ability to respond to change and to deal with an uncertain and turbulent environment [31]. ASDMs have helped software organizations to manage priority changes, accelerate software delivery and increase team productivity [22]. These characteristics, in line with the need

for speed of the organizational processes nowadays, have ensured the continuous growth in the adoption of ASDMs since their formalization in 2001 [31][62]. Among the existing Agile Methods, the most used by software organizations have been Scrum, ScrumBan, Hybrid models, Scrum+XP and Kanban [62].

However, despite its various advantages, software projects that use ASDMs also fail for several reasons such as: team capacity, lack of customer involvement and lack of project management [14][58][59]. In order to increase the success rate of software projects in agile methods, the project management has been supported by the use of metrics [45]. In the traditional or prescriptive software development methods, the measurement process [18] is typically used to support project management. The traditional measurement process, however, can be cumbersome and bureaucratic for organizations that adopt agile methods [40].

Metrics used in the context of ASDMs have been defined as agile metrics [22]. Agile metrics are measures, variables to which a value is assigned as the result of measurement [27], that emphasis on delivered value [22], affirms and reinforces agile principles, are easy to collect and can provide feedback on a frequent and regular basis [23][46]. The use of agile metrics has presented various benefits in ASDMs, such as helping planning, understanding and improving software quality, fixing software process problems, and motivating people [35][54]. Due to its advantages, a plethora of these agile metrics have been proposed to help to measure various aspects of the software development processes and products [13][35][37].

In a growing trend of interest in agile metrics, some secondary studies, such as [3], [11], [14], [29], [30] and [31], have analyzed different aspects related to agile metrics, but none of them focusing on how metrics are selected, classified and its context of use. From this identified gap arises our research question: "How metrics are used by organizations that adopt agile methods?". To answer this research question, we conducted a Systematic Literature Mapping Study (SLM) [15]. To understand how agile metrics are used, we analyze the context of metrics usage, how they are selected and classified, the target audience of metrics and the impact of their use.

The major contributions of this research are twofold: for researchers we present a comprehensive map of how primary studies have reported the selection, use and observed results of metrics in agile contexts; and practitioners can find a systematized collection of agile metrics and select the most useful ones to improve agile management according to their contexts.

*Article Title Footnote needs to be captured as Title Note

†Author Footnote to be captured as Author Note

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

2 Related Works

The use of metrics in agile methods has attracted secondary studies in recent years. In this section we present secondary studies more directly related to our research question.

In [13] the research protocol search for metrics used in agile development methods, with an inclusion criterion that restricts results to metrics relevant to government domain. From 21 selected primary studies 87 different metrics were extracted. Unlike the present mapping study, study [13] focus on metrics relevant to government and there is also no analysis of the context of use of the metrics and how they are selected and used.

In the secondary study [37] the research protocol searches for specific metrics for Scrum. In 13 selected primary studies, authors found 34 metrics applicable to Scrum, which are classified according to the Scrum ceremonies. Finally, it proposes a set of recommended software metrics to implement Scrum. Unlike the present mapping study, in [37] the study focuses exclusively on metrics for Scrum.

In [44] the research protocol defines a broad literature review, covering metrics used both in agile methods and in traditional development, with a focus on discovering which process and project attributes are measured. In 462 selected primary studies 64 attributes are extracted. The focus of study [44] in metrics for all contexts differs from the focus of the present mapping study, which is on metrics in the agile methods context. There is also no focus on analyzing the context of use of the metrics or how they were selected.

In studies [35] and [36] authors publish the same systematic review in a conference paper and expanded in a journal paper. The research protocol focuses on finding only the metrics used in agile methods applied in the industry, and the reasons and effects of using the metrics. In 30 selected primary studies 102 metrics that have reports on the reasons for use, effect of use, and importance are collected. Unlike this mapping study, the study [36] focuses on exclusive application in the industry, the collected metrics are not unified and there is no analysis of the context, or how are metrics selected and used.

The study [48], despite not explicitly defining its research protocol, it details 40 metrics grouped into 5 different categories and discusses the application of part of these metrics in the most popular agile methods. Unlike the present mapping study, in [48] the secondary study is not performed systematically and no analysis of the context or the metrics selection are presented.

In addition to these secondary studies, other secondary studies can be found in the literature that analyze specific metrics for specific software products or processes, such as effort [61] or software sustainability [8]. This kind of secondary study differs from the present study as they focus only on specific metrics and not for metrics in the agile context.

Thus, to the best of our knowledge, it was not possible to find a secondary study that focus on understanding how agile metrics have been used, analyzing the context of metrics usage, how they are selected and classified, the target audience of metrics and the impact of their use.

3 Methodology

In order to understand how organizations that adopt agile methods use metrics we undertook an SLM study following the procedures suggested by Petersen, Vakkalanka, and Kuzniarz [50], Petersen et al. [51], and Wohlin [64]. An SLM is a literature review method used to structure an area of interest, provide an overview of a research area, and determine the amount and type of research and results available within it [51].

As proposed in the guidelines for SLMs [44],[50],[64], we undertook a series of activities grouped into four main phases: (i) protocol definition, (ii) execution of searches, (iii) study selection and snowballing (iv) data extraction and analysis.

Based on the identified research need, the general research question was defined as: "How metrics are used by organizations that adopt agile methods?". After defining the general research question, the search terms were derived, search strings were developed and data sources were selected. From this research question, the following analysis questions were derived:

- Q1: Where are agile metrics applied?
- Q2: What is the context of use of the agile metrics?
- Q3: How are the agile metrics selected?
- Q4: What are the agile metrics used?
- Q5: What is the target audience for the metrics?
- Q6: What is the impact of using metrics?

4 Protocol Definition

4.1 Search Strategy

In order to apply the SLM we selected digital libraries considered most relevant to the Software Engineering area [30]: IEEEExplore, Scopus and ACM. In addition to these sources, the Snowballing technique was applied on the results [64].

4.2 Inclusion and Exclusion Criteria

The following **inclusion criteria** were defined: primary studies available in full in the selected databases; studies published in workshops, conferences, or journals; studies that report the use of agile metrics in ASDM contexts. The **exclusion criteria** were defined as: studies that only address the description of metrics without presenting their use in ASDM contexts.

4.3 Search Strings

From the main research question, two search terms were results: "metric" and "agile method". In order to find as many relevant primary studies as possible, other terms were defined as synonyms. For the term "agile method", as there is no consensus in the literature for its use, the most used agile methods and other similar methods known by the researchers were used as synonyms.

The search string was validated by all researchers using three previously known primary studies as a reference: [4], [34] and [45]. After these tests the term "software" was added to reduce the volume of false positives from studies from other unrelated areas. The search string, as shown follow, was then adapted to the syntax of each search engine and applied to the *title* and *abstract* fields: (software) AND (metric OR measure OR indicator OR KPI) AND (agile OR scrum OR kanban OR xp OR extreme programming OR scrumban OR lean)).

4.4 Studies Selection

The query execution and studies selection were performed according to the defined protocol. Three cycles were carried out during the fourth quarter of 2021: Iteration 1, Iteration 2 and Snowballing. The first three authors were responsible for carrying out the searches and studies selection in the digital libraries.

The initial search consisted of executing the search string in each of the digital libraries, returning a total of 1,888 studies. Iteration 1 consisted of reading the titles of the results by each of the first three authors, applying the inclusion and exclusion criteria. After the reviewing by the fourth author, 150 studies were selected, discarding repeated. Iteration 2 consisted of the full texts reading distributed among the first three authors. All studies selected or discarded were discussed among the authors until a consensus was reached, resulting in 33 selected studies. At the end, a Snowballing cycle was performed by the first author, with a review by the fourth author. No new primary studies were found using Snowballing, however, three related secondary studies ([9], [37] and [44]) were

found using this technique (see section 2). Table 1 shows the results of selection cycles in digital libraries.

Table 1 - Selection results

Digital Library	Initial	Iteration 1	Iteration 2
<i>ACM</i>	275	24	11
<i>IEEEExplore</i>	444	41	10
<i>Scopus</i>	1,169	85	12

The data that had relevance to answer the research question was then extracted by the first three authors and reviewed by the fourth author. After two cycles of reviews and adjustments in the data collection and classification, all studies initially selected were kept and considered relevant. The complete list of selected primary studies is presented in Table 2.

Although the search was not limited in time, the 33 selected primary studies were found from 2004 onwards. The concentration of selected primary studies between 2016 and 2020 (19 studies - 57%), indicate the growing relevance of this research.

Table 2 - Selected Studies

#	Reference
S01	A comparative study on measuring software functional size to support effort estimation in agile [60]
S02	A Metric-Based Approach to Managing Architecture-Related Impediments in Product Development Flow: An Industry Case Study from Cisco [53]
S03	Agile Enterprise Metrics [20]
S04	Agile Performance Indicators for Team Performance Evaluation in a Corporate Environment [18]
S05	An agile, measurable and scalable approach to deliver software applications in a large enterprise [21]
S06	An empirical study of software metrics for assessing the phases of an agile project [12]
S07	An Empirical Study of WIP in Kanban Teams [55]
S08	An innovative measurement programme for agile governance [52]
S09	Analyses of an agile methodology implementation [26]
S10	Applying Software Design Metrics to Developer Story: A Supervised Machine Learning Analysis [3]
S11	Assessing individual performance in Agile undergraduate software engineering teams [19]
S12	Controlling and Monitoring Agile Software Development in Three Dutch Product Software Companies [10]
S13	Designing and Implementing a Measurement Program for Scrum Teams: What Do Agile Developers Really Need and Want? [38]
S14	Effective Monitoring of Progress of Agile Software Development Teams in Modern Software Companies: An Industrial Case Study [43]
S15	How to measure agile software development [33]
S16	Measurement Based Performance Evaluation of DevOps [5]
S17	Measuring and Improving Agile Processes in a Small-Size Software Development Company [11]
S18	Measuring and Monitoring Agile Development Status [7]
S19	Measuring Best-in-Class Software Releases [24]
S20	Metrics in Agile Project Courses [4]
S21	Metrics to Meet Security & Privacy Requirements with Agile Software Development Methods in a Regulated Environment [63]
S22	Motivations and Measurements in an Agile Case Study [39]
S23	On the measurement of agility in software process [56]
S24	Quality assurance for Product development using Agile [2]
S25	Requirements Engineering Quality Revealed through Functional Size Measurement: An Empirical Study in an Agile Context [16]
S26	Software Metrics for Agile Software Development [34]
S27	Strong Agile Metrics: Mining Log Data to Determine Predictive Power of Software Metrics for Continuous Delivery Teams [25]
S28	Success Factors for Effective Process Metrics Operationalization in Agile Software Development: A Multiple Case Study [54]
S29	Supporting the deployment of ISO-based project management processes with agile metrics [42]

S30 Use of Software Metrics in Agile Software Development Process [29]

S31 Using a Roles Scheme to Derive Software Project Metrics [15]

S32 Using Analytics to Guide Improvement during an Agile-DevOps Transformation [57]

S33 Using Lehman's laws to validate the software evolution of agile projects [32]

5 Data Analysis

In this section, the data collected in the SLM are analyzed, seeking to answer the research questions. The data collected from the primary studies were categorized according to the research questions using the reciprocal translation technique [47], which consists of synthesizing the results from similar studies and translating each of the other cases into common categories. The broad map of the extracted data, grouped by the research questions, is presented in Table 4.

Not all studies present information regarding all research questions, for this reason some research questions do not correspond to the total number of selected studies in Table 4. Next, the extracted data are analyzed according to each research question.

Q1: Where are agile metrics applied?

The purpose of this research question is to understand in what type of environment: industry or academia, the studies were applied, using which agile method, and involving how many organizations.

Regarding the **type of environment** of the use of agile metrics, 63,63% (21) of the selected studies addresses the application of metrics in an industry context and 12,12% (4) of the studies were applied in an academic context, while 24,24% (8) are studies that do not explicitly present the environment type of application, despite some of that would even be possible to infer it.

In terms of the **agile method** used in the studies, 50% (12) use Scrum, followed by Lean with 20,83% (5), XP with 16,66% (4), and Kanban and Safe both with 4,16%, (1) study each. Other studies did not explicitly define the agile method used.

Regarding the **number of organizations** involved in the studies, 4% (17) approached 1 to 2 organizations, followed by 7,6% (3) of the studies involving 3 to 4 organizations and another 7,6% (3) involving 10 to 30 organizations.

Q2: What is the context of use of the agile metrics?

The purpose of this research question is to describe the characteristics of organizations that use agile metrics, as presented in the primary studies in terms of size, teams' experience, product domain, size and type.

Most of the studies did not inform the **sizes** of the teams involved in the applications of agile metrics. Among the studies that present this data, there are: 60% (3) involving several teams, followed by 1 study (20%) with a team of 5 to 9 people and another study (20%) addressing the use of metrics with a team of 10 people.

Regarding the **experience** of the teams involved in the study, 60% (3) papers reported dealing with teams with 1 to 5 years of experience in average and 2 (40%) studies reported working with

experienced teams. The other papers did not define the experience of the teams involved.

Regarding the **product domain**, most of the studies, 86,95% (20), involved software development for general areas, followed by the Educational, Financial, Security and Telecommunications areas, all with 4,34% (1) each.

About the software **product size**, as no common size measure was found in the studies, the textual description of the products made by the studies' authors was used as the basis for this classification. Most of the studies did not present any reference to the size of the software products. The other described the application of metrics with large software products, 47,05% (8), followed by medium and small products, both with 17,64% (3) studies each.

Most of the software **products' types** reported in the studies are desktop applications (35% - 7), followed by web application (25% - 5), mixed products (20% - 4) and mobile applications (10% - 2).

Q3: How are the agile metrics selected?

In order to understand how agile metrics were used in the studies, it is important to understand in what way the metrics were selected. Thus, we found in the studies three different strategies for selecting the metrics to be used: (i) through a study of the literature looking for the most appropriate metrics; (ii) through interviews with organization's team members; or (iii) through the judgment of an (external) expert group.

Among the selected primary studies, 50% (15) studies selected metrics based on **literature search**, 30% (9) studies conducted **interviews** with the teams to select the metrics that were later used in the application, and 20% (13) studies used **expert groups** or committees to define the metrics. Next, we present actions in some studies used to selected the metrics.

In S12 a literature search was carried out to collect information and explore the field of agile software development, governance, KPIs and interventions. After the literature search, a case study was carried out to have a first impression on agile software development adopted in practice. The objective was to discover and select KPIs to be used in the organization. Interviews and document reviews were also used to gather information about the agile development process and KPIs.

In S18, S13 and S15, based on a literature review on project success factors, the Goal-Question-Metric (GQM) approach was used to derive and select a set of metrics to meet the project sponsors' goals. In S02 the metrics were also collected and selected through a series of forums, including focus groups, interviews, discussions, tool analysis and document analysis to then define the appropriate metrics for the teams.

In S20 the organization’s objectives were discussed and explained to the target audience (team, leaders, and management), then workflows were described, and the metrics selected to measure adoption and usage. In S14, S27 and S30 interviews were conducted, and surveys were performed to select the metrics.

In S28 a series of metrics were selected through several workshops with the 4 companies involved in the case study. In S11, S21, S03, S10, S09, S16 and S08 the metrics were selected according to the knowledge and experience of the authors of the studies.

Q4: What are the agile metrics used?

The selected primary studies reported the use of a wide variety of agile metrics with different scopes and objectives. In total, 319 different agile metrics were extracted from the studies.

Among the selected primary studies, a series of possible repeated metrics were found (metrics with different names/titles but possible same meaning). To unify the list of metrics and avoid repetition, we followed a standard procedure among all researchers: (1) we extracted the metrics with the original name indicated in the paper; (2) we performed the analysis of the attribute measured by the metric and the objective of the metric; (3) we performed a comparison of the new metric with all metrics already recorded; (4) if there was already another metric with the same title, or the same measured attribute and objective or meaning, we kept the metric already registered, otherwise a new metric was registered.

Following this procedure, we identified a list of 132 unique metrics. The raw list of the extracted metrics is available at: <https://bit.ly/3vIM1F5> (tab extracted metrics). The unified/merged list of the metrics is presented in Table 3.

As it was not possible to find in the selected primary studies a standard classification scheme for agile metrics that have been created or even used by the primary studies, we decided to classify the metrics into a widely accepted set of software metric categories. Thus, we decided to use a classic software metrics classification standard, PSM [17] that classifies the software metrics, according to the attribute(s) they measure, in: Schedule and progress, People, Product, Process, Technology and Customer.

To group the list of metrics into the chosen categories, we mainly used the attribute measured by each metric and its meaning. The classification of metrics was performed by the first author and revised in detail by the fourth author.

Considering the number of primary studies that presented metrics for each type, the most studies 28 (~26,4%) presented metrics related to Product quality, 23 (~21,7%) for Resources and Cost, 17 (16%) for Process performance, 14 (~13,2%) for Schedule and progress, (11 ~ 10,4%) presented metrics related to Technology Effectiveness, 8 (~7%) for Product size and stability and 5 (~4,7%) metrics for Costumer Satisfaction. Table 3 shows the unified list of extracted metrics, according to the grouped categories.

Table 3 - Classified metrics

Schedule and progress (14 studies)	Number of completed tasks, Scope growth, Priority Shift, Rejected Product Backlog Items, Indicator about risky user stories, Reopened Tickets, Cumulative Flow Diagrams, Number of epics last sprint, WIP (work in progress), Blocked tasks, Merge Request Life Time, Merge Request Review
--	--

Resource and Cost (23 studies)	Individual effectiveness, Weekly working hours of individual, Individual effective available hours, Individual Contribution, Individual Influence, Individual Impact, Individual Impression, Effort remaining, Effort, Team total available hours, Team total effective available hours, Planned hours, Number of Sprints, Sprint duration, Effort estimation accuracy, User Story Points, Number of remaining tasks, Backlog size, Cost Devitation, Earned Business Value, Number of Projects (n), Project size, Project avg. cost, Estimation of user stories, Pairing Frequency, Number of members per team, Role Time Measure (RTM), Role Communication Measure (RCM), Role Management Measure (RMM), Ideal team capacity, Non-working days, Cost model (CM), Team and individual motivational, Number of change team members, ROI, Number of Active Customers, Team adaptability, Team Innovation, Effort spent outside sprint
Product size and stability (8 studies)	Changed Product Backlog Items, Number of code lines, Weak Components
Product quality (28 studies)	Outstanding bugs, Critical bugs, Known bugs, Project avg. time-to-market, Delivery on time, Lead Time, Delivery to customer, Technical debt, Throughput, Velocity, Velocity Deviation, Burndown, Quality rating, Number of commits, Number of builds, Build time, Number of Attributes (NOA), Number of Method (NOM), Number of Parameters (NOP), Depth Inheritance Tree (DIT), Number of Children (NOC), Response for a Class (RFC), Lack of Cohesion of Methods (LCOM), Cohesion Among Methods (CAM), Access to Foreign Data (ATFD), Tight Class Cohesion (TCC), Number of Name-Parts of a method (NNP), Number of Characters (NC), Number of Comment - Lines (CL), Number of Local Variables (NLV), Number of Created Objects (NCO), Number of Referring Objects (NRO), Class-level (Coupling between Objects, Number of Children, Depth of Inheritance Tree), Number of Referring Objects (NRO), Package-level (Cycle Count of Dependency Graph), Requirements clarity index, Bugs Density, Code Scanning (security), Detection Rate (security), Collective Code Ownership, Fan-In, Weighted Fan-In, Fan-Out, Weighted Fan-Out, Reach efficiency In, Reach Efficiency Out, Closeness-In, Closeness-Out, Purchased delivery units quantity, Technical Efficiency, Weighted Methods per Class
Process performance (17 studies)	Tasks attributes quality, Number of development tasks without estimation of effort ("estimated"), Number of development tasks without real effort ("spend"), Fixed bugs, Hours spent on bugs, Time to fix, Hours spent on task, Cycle Time, Total points by use case, Queue impediments size, Downstream Impact, Code changes during sprint, Scrum teams (S), Inspection Frequency, Flow efficiency, Time in meeting, Focus factor
Technology Effectiveness (11 studies)	Test Run Frequency, Test failure rate, Avg test run time, Test Coverage, Test success rate, Running Tested Features Metric (RTF), Security Test Pass Rate (security), Number of deployments, Commit Review Performance
Costumer Satisfaction (5 studies)	NPS

Q5 - What is the target audience for the metrics?

With this research question we aim to understand who the metrics defined in the selected primary studies are intended for. It is important to clarify that this research question differs from what is presented in Table 4, which refers to what type of attribute the metric measures, while this research question refers to whom the

information generated by the metric is intended for or by whom it is used.

In terms of target audience, the following result was obtained: 28 studies (72%) reported that the metrics were intended to focus on the **agile team**, followed by **management** with 17 studies (43.5%) and the **customer** with 9 studies (23%).

It is interesting to note that the greater focus of the metrics presented in the studies being directed to team members is aligned with the agile principles and values [31].

Q6 - What is the impact of using metrics

In this research question, we seek to know the main results of using metrics in a context of agile methods. For this, the main results of the use of metrics reported in the primary studies are described below.

We found in the primary studies three main types of impacts of the use of agile metrics: impacts on the **quality of products**, impacts on the **quality of processes** and on the **quality of services**. Below we present details on the most relevant impacts observed in the primary studies.

In study S12 is presented metrics on the quality of processes, the metrics focused on the team and the individual metrics allowed to observe the work overload that the team was going through, and how the stress of a member impacts the team as a whole. The proposed metrics also supported the correct distribution of tasks, making it possible to pay attention to the professional development of the team. Also, in S12 task-oriented metrics supported task management, where unfinished tasks were moved to the next release and critical bugs took precedence over tasks. And as for quality of products, the metrics brought transparency for well-built and poorly built software by showing to the entire organization tests passed and bugs resolved, so the team strived to avoid problems in the future and resolve issues quickly.

In S20 on the quality of process, when instructors and project leaders observing a deviation, e.g. a large number of failed builds, they could go to the respective systems and investigate root causes or talk to the responsible members of the development team. Study S20 also reports that, sometimes, even when some metrics pointed to a problem, the team could have reasons to keep their focus. These scenarios were related to one of the key values of the agile manifesto: "Individuals and interactions over processes and tools", and teams were not forced to change their approach.

In S04, on the quality of process, the main objective for the use of metrics was to support lean and agile development teams to evaluate their performance and reach a higher quality level with greater motivation. For the best, the teams needed to experiment, receive feedback, and act. Furthermore, in order to assess whether an action created value and helped the team, an analysis of the cases was made to be measured too. The proposed agile performance indicators model provides a comprehensive solution to measure the current status of the team so that improvements can be made.

In S07, on the quality of process, the use of metrics allowed to perform a deeper analysis on the Work in Progress (WIP) of the teams. Based on a comprehensive dataset, a low WIP usually reduces the waiting time, which is positive and consistent with the current claims in the literature and among practitioners. However,

it was observed that a low WIP also appeared to reduce productivity, which was perceived as negative, in contrast to claims in the literature and among practitioners. Well-structured, application-dominant teams tend to handle larger WIP well, and such analysis was only possible by making use of proposed agile metrics.

In S22, on the quality of product, a 65% reduction in pre-release product defect rates, a 35% reduction in release period defect rate, and a 50% improvement in productivity were observed, all improvements based on the use of metrics in the team's day-to-day activities: productivity, number of defects delivery to and reported by the customer, defect density and defect removal efficiency.

In the study S28 it was identified that the proposed quality of process metrics were able to understand the uncertainty and respond to the dynamics inherent to the agile project context. The use of quality of services metrics resulted in strongly supported clients, considering the different contexts caused by changes and facilitated reprioritization as clients' realities changed over time. The use of agile metrics of quality of products also allowed incremental investments while focusing on the critical parts of the software's functionality, essential to fulfill the project's objectives. Using metrics, the team was able to understand that not all architectural design requirements and options are of equal value.

Study S13 reports that agile metrics of quality of products allowed the understanding that technical debt, business value visibility and estimation issues are the most urgent needs that all stakeholders agreed on. From the developers' perspective, technical debt seems to have the worst impact on team productivity, collaboration efficiency and transparency and a strong impact on product quality.

In S27, S09, S05 and S08, on the quality of process, similar improvements in team productivity were reported. The use of agile metrics allowed teams to observe how to improve the completion rate of planned stories and reduce cycle time, splitting deliverables into smaller user stories. Teams were able to reduce their cycle time by keeping an open space in the sprint planning (e.g. increasing their proportion of time remaining). Finally, reducing unplanned work, teams could increase deliver predictability (e.g., completion rate of planned stories).

In study S17 the team also observed great advantages in the use quality of process metrics, such as process effectiveness and team productivity. The proposed approach improved the way developers reported time spent on issues/tickets and allowed comparison with planned effort. By including dashboards and process metrics, software development has improved readiness and efficiency in reporting, time spent, and effort planning. Furthermore, the gap between planned and actual effort continued to shrink, which means that Product Owners, Scrum Masters and developers estimate much better. The proposed metrics allowed efficient tracking of tasks and issues in the project, by each developer and during each Sprint.

Study S11 reports how the use of metrics of quality of process allows for a more accurate analysis of the broad characteristics of the level of involvement, activity and results related to the productivity of an individual in a team. Teams with higher individual levels of contribution and impact had better end-product results. In addition, team members were able to form accurate assessments of each other's contributions, influence, and impact on

the project, given the high visibility of the development process. The use of metrics also provides a basis for direct and objective

feedback to individuals on more detailed qualities in relation to performance after each Sprint.

Table 4 - General map of extracted data

Classification			Selected studies
Q1 - Where are agile metrics applied?	Q1.1 Study environment	Industry	S01, S02, S05, S06, S07, S08, S09, S12, S13, S14, S15, S17, S18, S19, S22, S23, S25, S27, S28, S32, S33
		Academy	S11, S16, S20, S31
		Not informed	S03, S04, S10, S21, S24, S26, S29, S30
	Q1.2 Agile Method	Scrum	S01, S03, S08, S11, S12, S13, S17, S19, S20, S25, S27, S28
		Lean	S02, S04, S14, S18, S32
		XP	S09, S15, S22, S31
		Kanban	S07
		Safe	S05
	Q1.3 Number of participant Organizations	1 to 2	S01, S05, S06, S07, S08, S09, S13, S17, S19, S20, S22, S25, S27, S31, S32, S28
		3 to 4	S23, S33
10 to 30		S10, S30	
Q2 - What is the context of use of the agile metrics?	Q2.1 Team Size	Various teams	S02, S07, S14
		5 to 9 people	S12
		10 people	S31
	Q2.2 Team's Experience	1 to 5 years	S09, S12, S16
		Experienced	S14
	Q2.3 Product domain	Educational	S06
		General	S01, S02, S07, S08, S09, S12, S13, S14, S15, S16, S17, S18, S19, S20, S22, S25, S27, S28, S32, S33
		Security	S21
	Q2.4 Product size	Telecom	S05
		Large	S02, S07, S12, S18, S20, S22, S27, S32
		Medium	S09, S17, S28
	Q2.5 Product type	Small	S16, S19, S25
		Application	S09, S13, S16, S18, S19, S22, S28
		Mixed	S02, S12, S27, S32
		Mobile	S01, S20
Other		S31	
Q3 - How are the agile metrics selected?	Q3.1 How metrics were selected	Web	S05, S06, S07, S17, S25
		Literature study	S24, S01, S07, S09, S10, S12, S21, S22, S24, S25, S26, S27, S29, S31, S33
		Interviews (team)	S02, S13, S14, S15, S18, S20, S28, S30, S32
		Expert Group	S03, S08, S11, S16, S17, S19
Q4 - What are the agile metrics used?	Q4.1 Classification Structure (metrics)	Schedule and Progress	S02, S05, S07, S12, S13, S17, S18, S20, S21, S27, S28, S29, S30, S32
		Resources and Cost	S01, S02, S03, S04, S05, S07, S09, S11, S12, S13, S14, S17, S18, S19, S21, S22, S23, S25, S27, S28, S30, S31, S33
		Product Size and Stability	S06, S09, S10, S18, S22, S26, S32, S33
		Product Quality	S02, S03, S04, S05, S06, S07, S08, S09, S10, S12, S14, S15, S16, S17, S19, S20, S21, S22, S23, S24, S25, S26, S27, S28, S29, S30, S32, S33
		Process Performance	S01, S02, S03, S04, S05, S08, S12, S13, S16, S17, S22, S24, S27, S28, S29, S30, S33
		Technology Effectiveness	S08, S12, S13, S16, S20, S21, S22, S26, S27, S28, S30
		Customer Satisfaction	S09, S13, S28, S30, S33
		Agile Team	S07, S10, S14, S20, S22, S26, S27, S13, S01, S02, S03, S09, S11, S16, S17, S19, S21, S24, S25, S29, S30, S32, S12, S33
Q5 - What is the target audience for the metrics?	Q5.1 Metrics target audience	Management	S13, S01, S02, S03, S09, S11, S16, S17, S19, S21, S24, S25, S29, S30, S32, S12, S33
		Customer	S18, S04, S28, S13, S12, S33
		Team Improvement	S22, S01, S28, S25
Q6 - What is the impact of using metrics?	Q6.1. Impacted area	Process quality	S01, S25, S23, S05, S16, S19, S02, S03, S09, S12, S29, S33, S08, S20, S13
		Product quality	S28, S25, S18, S30, S31, S02, S03, S09, S12, S29, S33, S08, S20, S04, S06, S07, S10, S11, S14, S17, S21, S24, S26, S27, S32, S13
		Quality service	S18

6 Mapping and Discussion

The data collection and analysis presented in Section 6 indicates several possible correlations between the organizational and product characteristics and the type of metrics used. Thus, Table 4 presents these detailed relationships evidenced by the primary studies.

Regarding the **environment** of study application, it is possible to notice that the studies applied in the industry are the ones that

present the most diversity and quantity of types of metrics. In relation to agile methods, it is possible to see that Scrum naturally dominates, due to the greater number of studies. However, no metrics with Scrum were found in the Customer category.

Regarding the **size and experience of the team**, as few studies presented this type of information, it is not possible to carry out very in-depth analyzes. However, it is possible to note that less experienced teams have defined metrics for schedule and people.

Table 5 – Correlation between categories and Metric Types

	<i>Customer Satisfaction</i>	<i>Process Performance</i>	<i>Product Quality</i>	<i>Product Size and Stability</i>	<i>Resources and Cost</i>	<i>Schedule and Progress</i>	<i>Technology Effectiveness</i>	
Environment	Industry	S09, S13, S28, S33	S01, S02, S05, S08, S12, S13, S17, S22, S27, S28, S33	S02, S05, S06, S07, S08, S09, S12, S14, S15, S17, S19, S22, S23, S25, S27, S28, S32, S33	S06, S09, S18, S22, S32, S33	S01, S02, S05, S07, S09, S12, S13, S14, S17, S18, S19, S22, S23, S25, S27, S28, S33	S02, S05, S07, S12, S13, S17, S18, S27, S28, S32	S08, S12, S13, S22, S27, S28
	Acad.		S16	S16, S20		S11, S31	S20	S16, S20
Agile Method	Not Informed	S30	S03, S04, S24, S29, S30	S03, S04, S10, S21, S24, S26, S29, S30	S10, S26	S03, S04, S21, S30	S21, S29, S30	S21, S26, S30
	Scrum	S13, S28	S01, S03, S08, S12, S13, S17, S27, S28	S03, S08, S12, S17, S19, S20, S25, S27, S28		S01, S03, S11, S12, S13, S17, S19, S25, S27, S28	S12, S13, S17, S20, S27, S28	S08, S12, S13, S20, S27, S28
	Lean		S02, S04	S02, S04, S14, S32	S18, S32	S02, S04, S14, S18	S02, S18, S32	
	XP	S09	S22	S09, S15, S22	S09, S22	S09, S22, S31		S22
	Kanban			S07		S07	S07	
	Safe		S05	S05		S05	S05	
	Various		S02	S02, S07, S14		S02, S07, S14	S02, S07	
Team Size	5 to 9		S12	S12		S12	S12	S12
	10					S31		
	1 to 5	S09	S12, S16	S09, S12, S16	S09	S09, S12	S12	S12, S16
Exper.	Exper. Experiências variadas			S14		S14		
	Educat.			S02		S02	S02	
Application	General	S09, S13, S28, S33	S01, S02, S08, S12, S13, S16, S17, S22, S27, S28, S33	S02, S07, S08, S09, S12, S14, S15, S16, S17, S19, S20, S22, S25, S27, S28, S32, S33	S09, S18, S22, S32, S33	S01, S02, S07, S09, S12, S13, S14, S17, S18, S19, S22, S25, S27, S28, S33	S02, S07, S12, S13, S17, S18, S20, S27, S28, S32	S08, S12, S13, S16, S20, S22, S27, S28
	Security			S21		S21	S21	
	Telecom		S05	S05		S05	S05	
Product	Large		S02, S12, S22, S27	S02, S07, S12, S20, S22, S27, S32	S18, S22, S32	S02, S07, S12, S18, S22, S27	S02, S07, S12, S18, S20, S27, S32	S12, S20, S22, S27
	Medium	S09, S28	S17, S28	S09, S17, S28	S09	S09, S17, S28	S17, S28	S28
	Small		S16	S16, S19, S25		S19, S25		S16
Product Type	Applic.	S09, S13, S28	S13, S16, S22, S28	S09, S16, S19, S22, S28	S09, S18, S22	S09, S13, S18, S19, S22, S28	S13, S18, S28	S13, S16, S22, S28
	Mixed		S02, S12, S27	S02, S12, S27, S32	S32	S02, S12, S27	S02, S12, S27, S32	S12, S27
	Mobile		S01	S20		S01	S20	S20
	Other					S31		
Web		S05, S17	S05, S06, S07, S17, S25	S06		S05, S07, S17, S25	S05, S07, S17	

On the **product domain**, for general-domain products, most metrics focus on Product Quality, Resources and Costs. Regarding the **size of the product** is for large products.

About the **product types**, the development of Desktop Applications led to the use of metrics for Schedule and Technology, while for Web products and Mixed products, there was a slightly greater predominance of metrics for Product.

It is interesting to note that, for several relationships, the primary studies did not define metrics. These gaps may indicate that metrics are not applicable in these contexts, or also may indicate opportunities where future research may be applied.

7 Threats to validity

We identified potential threats and applied mitigation strategies to minimize their impact on the results following Zhou et al. [65].

Omission of relevant studies. To reduce the risk of relevant studies not being considered, following [51] we defined our search string through a systematic process, in which search terms were selected to describe concepts related to our main research question and also synonyms of the terms were used. The search string was also tested and validated by the four authors using previously known primary studies as a reference.

Missing relevant information. To minimize the risk of not collecting the relevant information from the primary studies, we systematically derived the data to be extracted from the research questions and defined a shared data collection form. The collected data were also reviewed by the fourth author.

Data classification. There is a possible threat to the validity of the subjectivity in classifying data into defined categories. This applies to all categories and metric rankings. Despite this possible threat, all categorizations were peer-reviewed by the authors in order to minimize individual opinions.

8 Conclusion

This study presents an SLM on the use of metrics in agile software organizations. Seven research questions were defined, and, through a strict research protocol, 33 primary studies were selected. Appropriate data were then extracted and analyzed to answer the research questions.

As a result, we observed that most organizations use agile metrics with Scrum, Lean or XP. Metrics are used to support the development of large size, general-purpose, desktop or web applications, through small teams. The selected primary studies select metrics based on literature search, interviews with the team members or expert groups. Most studies defined Product Quality, Resources and Costs metrics.

Among the observed impacts of using agile metrics are improvements on the quality of products, impacts on the quality of processes and on the quality of service. The selected primary studies indicated the feasibility of the use of properly adapted metrics in the context of agile software development methods not identifying any increased effort or other possible negative impact that could undermine the principles and values of agile.

Based on the identification of gaps in the literature, future works may involve the development of specific agile metrics for organizational contexts, agile methods, and software product profiles where no evidence were found in the primary studies.

Availability of Artifacts

The data collected from the primary studies are available at: <https://bit.ly/3vIM1F5>. Data is presented separated by research question, as presented in this paper.

REFERENCES

- [1] Alexandre Bartié. 2002. *Garantia da qualidade de software*. [S.l.]: Gulf Professional Publishing.
- [2] Agarwal, A., Garg, N. K., & Jain, A. (2014, February). Quality assurance for Product development using Agile. In 2014 International Conference on Reliability Optimization and Information Technology (ICROIT) (pp. 44-47). IEEE. doi: 10.1109/ICROIT.2014.6798281
- [3] Algarni, A., & Magel, K. (2019, December). Applying Software Design Metrics to Developer Story: A Supervised Machine Learning Analysis. In 2019 IEEE First International Conference on Cognitive Machine Intelligence (CogMI) (pp. 156-159). IEEE. doi: 10.1109/CogMI48466.2019.00030
- [4] Alperowitz, L., Dzvonyar, D., & Bruegge, B. (2016, May). Metrics in agile project courses. In Proceedings of the 38th International Conference on Software Engineering Companion (pp. 323-326). doi: 10.1145/2889160.2889183
- [5] Batra, P., & Jatain, A. (2020, July). Measurement Based Performance Evaluation of DevOps. In 2020 International Conference on Computational Performance Evaluation (ComPE) (pp. 757-760). IEEE. doi: 10.1109/ComPE49325.2020.9200149
- [6] Bickman L and Rog D., 2018. *Applied Research Design: A practical approach*. The SAGE Handbook of Applied Social Research Methods, pp. 3-43.
- [7] Boerman, M. P., Lubsen, Z., Tamburri, D. A., & Visser, J. (2015, May). Measuring and monitoring agile development status. In 2015 IEEE/ACM 6th International Workshop on Emerging Trends in Software Metrics (pp. 54-62). IEEE.s. doi: 10.1109/WETSoM.2015.15
- [8] Calero, C., Bertoa, M. F., & Moraga, M. Á. (2013, May). A systematic literature review for software sustainability measures. In 2013 2nd international workshop on green and sustainable software (GREENS) (pp. 46-53). IEEE.
- [9] Chakravarty, K., and Jagannath S., 2021. A Study of Quality Metrics in Agile Software Development. *Machine Learning and Information Processing*. Springer, Singapore, 2021. 255-266.
- [10] Cheng, T. H., Jansen, S., & Remmers, M. (2009, May). Controlling and monitoring agile software development in three dutch product software companies. In 2009 ICSE workshop on software development governance (pp. 29-35). IEEE. doi: 10.1109/SDG.2009.5071334
- [11] 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. doi: 10.1109/ACCESS.2020.2990117
- [12] 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. doi: 10.1142/S0218194012500131
- [13] Dantas C. B., 2013. Um mapeamento sistemático de métricas para metodologias ágeis Scrum, Kanban e XP. *Monografia*. Universidade de Brasília.
- [14] Dhir S.; Kumar D. and Singh V. B., 2018. Success and failure factors that impact on Project Implementation Using Agile Software Development methodology. *Advances in Intelligent Systems and Computing*, pp. 647-654.
- [15] Dubinsky, Y., & Hazzan, O. (2004, November). Using a roles scheme to derive software project metrics. In Proceedings of the 2004 workshop on Quantitative techniques for software agile process (pp. 34-39). doi: 10.1145/1151433.1151438
- [16] Dumas-Monette, J. F., & Trudel, S. (2014, October). Requirements engineering quality revealed through functional size measurement: an empirical study in an agile context. In 2014 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement (pp. 222-232). IEEE. doi: 10.1109/TWSM.Mensura.2014.43
- [17] DoD - Department of Defense. Practical Software and Systems Measurement: A Foundation for Objective Project Management, v. 4.0b1. Available at: <https://www.psmc.com/psmguide.asp>
- [18] Ertaban, C., Sarikaya, E., & Bagriyanik, S. (2018, May). Agile performance indicators for team performance evaluation in a corporate environment. In Proceedings of the 19th International Conference on Agile Software Development: Companion (pp. 1-3). doi: 10.1145/3234152.3234156
- [19] Gamble, R. F., & Hale, M. L. (2013, October). Assessing individual performance in Agile undergraduate software engineering teams. In 2013 IEEE Frontiers in Education Conference (FIE) (pp. 1678-1684). IEEE. doi: 10.1109/FIE.2013.6685123
- [20] Greening, D. R. (2015, January). Agile Enterprise Metrics. In 2015 48th Hawaii International Conference on System Sciences (pp. 5038-5044). IEEE. doi: 10.1109/HICSS.2015.597
- [21] 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. doi: 10.1504/IJASM.2016.08156
- [22] Hayes, W., Miller, S., Lapham, M. A., Wrubel, E., & Chick, T., 2014. Agile metrics: Progress monitoring of agile contractors. Carnegie-Mellon Univ. Pittsburgh Pa Software Engineering Inst.
- [23] Hartmann, D., & Dymond, R. (2006, July). Appropriate agile measurement: using metrics and diagnostics to deliver business value. In AGILE 2006 (AGILE'06) (pp. 6-pp). IEEE.

- [24] Huijgens, H., & van Solingen, R. (2013, October). Measuring Best-in-Class Software Releases. In 2013 Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement (pp. 137-146). IEEE. doi: 10.1109/IWSM-Mensura.2013.29
- [25] 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). doi: 10.1145/3106237.3117779
- [26] Ilieva, S., Ivanov, P., & Stefanova, E. (2004, September). Analyses of an agile methodology implementation. In Proceedings. 30th Euromicro Conference, 2004. (pp. 326-333). IEEE. doi: 10.1109/EURMIC.2004.1333387
- [27] International Standardization Organization. 2007. International Organization for et al. Systems and software engineering—Measurement process. ISO/IEC 15939: 2007, v. 1.
- [28] James Johnson. 2018. *CHAOS report: decision latency theory: it is all about the interval*. [S.l.]: Lulu.
- [29] Jeeva KV. Padmini; Dilum HMN Bandara and Indika Perera, Indika. 2015. *Use of software metrics in agile software development process*. In: IEEE. 2015 Moratuwa Engineering Research Conference (MERCOn). [S.l.: s.n.]. P. 312–317.
- [30] Keele, Staffs. Guidelines for performing systematic literature reviews in software engineering. Vol. 5. Technical report, Ver. 2.3 EBSE Technical Report. EBSE, 2007.
- [31] Kent Beck et al. 2001. *Manifesto for agile software development*. Available at: <https://agilemanifesto.org/>
- [32] Kour, G., & Singh, P. (2016, March). Using Lehman's laws to validate the software evolution of agile projects. In 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT) (pp. 90-96). IEEE. doi: 10.1109/ICCTICT.2016.7514558
- [33] Kunz, M., Dumke, R. R., & Schmietendorf, A. (2007). How to measure agile software development. In Software Process and Product Measurement (pp. 95-101). Springer, Berlin, Heidelberg. doi: 10.1007/978-3-540-85553-8_8
- [34] Kunz, M., Dumke, R. R., & Zenker, N. (2008, March). Software metrics for agile software development. In 19th Australian Conference on Software Engineering (aswec 2008) (pp. 673-678). IEEE. doi: 10.1109/ASWEC.2008.4483261
- [35] Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2015). Using metrics in Agile and Lean Software Development—A systematic literature review of industrial studies. Information and software technology, 62, 143-163.
- [36] Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2014, June). Why are industrial agile teams using metrics and how do they use them? In Proceedings of the 5th International Workshop on Emerging Trends in Software Metrics (pp. 23-29).
- [37] Kurnia, Reni, Ridi Ferdiana, and Sunu Wibirama. 2018. *Software metrics classification for agile scrum process: A literature review*. International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). IEEE.
- [38] Ktata, O., & Lévesque, G. (2010, May). Designing and Implementing a Measurement Program for Scrum Teams: What do agile developers really need and want?. In Proceedings of the Third C* Conference on Computer Science and Software Engineering (pp. 101-107). doi: 10.1145/1822327.1822341
- [39] Layman, L., Williams, L., & Cunningham, L. (2004, November). Motivations and measurements in an agile case study. In Proceedings of the 2004 workshop on Quantitative techniques for software agile process (pp. 14-24). doi: 10.1145/1151433.1151436
- [40] Lomio F. Codabux Z.; Birtch D.; Hopkins D. and Taibi D. 2022.. *On the Benefits of the Accelerate Metrics: An Industrial Survey at Vendasta*. IN: Proceedings of 29th IEEE International Conference on Software Analysis, Evolution and Reengineering.
- [41] Mauricio Concha; Marcello Viscont and Hernán Astudillo. 2007. *Agile commitments: Enhancing business risk management in agile development projects*. In: SPRINGER. INTERNATIONAL Conference on Extreme Programming and Agile Processes in Software Engineering. [S.l.: s.n.]. P. 149–152.
- [42] 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. doi: 10.1016/j.csi.2019.103405
- [43] Meding, W. (2017, October). Effective monitoring of progress of agile software development teams in modern software companies: an industrial case study. In Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement (pp. 23-32). doi: 10.1145/3143434.3143449
- [44] Meidan, Ayman, et al. 2018. *Measuring software process: A systematic mapping study*. ACM Computing Surveys (CSUR) 51.3: 1-32.
- [45] Mounir, M., Salah, A., Kamel, A., & Moussa, H. (2020, November). Framework to measure agile software process effectiveness in critical systems development. In Proceedings of the 2020 9th International Conference on Software and Information Engineering (ICSIE) (pp. 25-32).
- [46] Mukker, A. R., Singh, L., & Mishra, A. K. (2014). Systematic review of metrics in software agile projects. Compusoft, 3(2), 533-539.
- [47] Noblit GW, Hare RD. Meta-ethnography: synthesizing qualitative studies, vol. 11. California: Sage Publications; 1988.
- [48] Padhy, N., Panigrahi, R., & Baboo, S. (2015, January). A systematic literature review of an object oriented metric: reusability. In 2015 international conference on computational intelligence and networks (pp. 190-191). IEEE.
- [49] Padmini, K. J., Bandara, H. D., & Perera, I. (2015, April). Use of software metrics in agile software development process. In 2015 Moratuwa Engineering Research Conference (MERCOn) (pp. 312-317). IEEE. doi: 10.1109/MERCOn.2015.7112365
- [50] Petersen, Kai, Sairam Vakkalanka, and Ludwik Kuzniarz. 2015. *Guidelines for conducting systematic mapping studies in software engineering: An update*. Information and software technology, vol. 64, pp. 1-18.
- [51] Petersen, Kai, et al. 2008. *Systematic mapping studies in software engineering*. 12th International Conference on Evaluation and Assessment in Software Engineering (EASE) 12. 2008.
- [52] Poligadu, A., & Moloo, R. K. (2014). An innovative measurement programme for agile governance. International Journal of Agile Systems and Management, 7(1), 26-60. doi: 10.1504/IJASM.2014.059153
- [53] Power, K., & Conboy, K. (2015, May). A metric-based approach to managing architecture-related impediments in product development flow: an industry case study from Cisco. In 2015 IEEE/ACM 2nd International Workshop on Software Architecture and Metrics (pp. 15-21). IEEE. doi: 10.1109/SAM.2015.10
- [54] Ram, P., Rodriguez, P., Oivo, M., & Martínez-Fernández, S. (2019, May). Success factors for effective process metrics operationalization in agile software development: A multiple case study. In 2019 IEEE/ACM International Conference on Software and System Processes (ICSSP) (pp. 14-23). IEEE. doi: 10.1109/ICSSP.2019.00013
- [55] Sjöberg, D. I. (2018, October). An empirical study of WIP in kanban teams. In Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (pp. 1-8). doi: 10.1145/3239235.3239238
- [56] Shen, B., & Ju, D. (2007, May). On the measurement of agility in software process. In International Conference on Software Process (pp. 25-36). Springer, Berlin, Heidelberg. doi: 10.1007/978-3-540-72426-1_3
- [57] Snyder, B., & Curtis, B. (2017). Using analytics to guide improvement during an Agile-DevOps transformation. IEEE Software, 35(1), 78-83. doi: 10.1109/MS.2017.4541032
- [58] Tam C.; Moura E. J.; T. Oliveira, and J. Varajão. 2020. *The factors influencing the success of on-going agile software development projects*. International Journal of Project Management, vol. 38, no. 3, pp. 165–176.
- [59] Tanner M. and U. von Willingh. 2014. *Factors leading to the success and failure of agile projects implemented in traditionally waterfall environments*. In: Human Capital without Borders: Knowledge and Learning for the Quality of Life. Portoroz, Slovenia: Make Learn. 693-701.
- [60] Ugalde, F., Quesada-López, C., Martínez, A., & Jenkins, M. (2020). A comparative study on measuring software functional size to support effort estimation in agile. In ClbSE (pp. 208-221).
- [61] Usman, M., Mendes, E., Weidt, F., & Britto, R. (2014, September). Effort estimation in agile software development: a systematic literature review. In Proceedings of the 10th international conference on predictive models in software engineering (pp. 82-91).
- [62] Versionone, CollabNet. 2021. 15th annual state of agile report. collab.net.
- [63] Wagner, T. J., & Ford, T. C. (2020, February). Metrics to meet security & privacy requirements with agile software development methods in a regulated environment. In 2020 International Conference on Computing, Networking and Communications (ICNC) (pp. 17-23). IEEE. doi: 10.1109/ICNC47757.2020.9049681
- [64] Wohlin, Claes. 2014. *Guidelines for snowballing in systematic literature studies and a replication in software engineering*. Proceedings of the 18th international conference on evaluation and assessment in software engineering.
- [65] Zhou, X., Jin, Y., Zhang, H., Li, S., & Huang, X. (2016). A map of threats to validity of systematic literature reviews in software engineering. In 2016 23rd Asia-Pacific Software Engineering Conference (APSEC), 153-160.