



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CAMPUS DE QUIXADÁ**  
**ENGENHARIA DE SOFTWARE**

**ALLAN PEIXOTO MARQUES**

**MAPEAMENTO SISTEMÁTICO DE MÉTRICAS DE QUALIDADE PARA**  
**SISTEMAS CRÍTICOS**

**QUIXADÁ**  
**2018**

ALLAN PEIXOTO MARQUES

MAPEAMENTO SISTEMÁTICO DE MÉTRICAS DE QUALIDADE PARA SISTEMAS  
CRÍTICOS

Monografia apresentada ao Curso de Engenharia de Software do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Orientadora: Prof<sup>a</sup> Ma. Jéssyka Flavyanne Ferreira Vilela

QUIXADÁ

2018

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

M315m Marques, Allan Peixoto.  
Mapeamento sistemático de métricas de qualidade para sistemas críticos : Mapeamento sistemático /  
Allan Peixoto Marques. – 2018.  
67 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá,  
Curso de Engenharia de Software, Quixadá, 2018.  
Orientação: Prof. Me. Jéssyka Flavyanne Ferreira Vilela.

1. Software - Controle de qualidade. 2. Métrica de software. I. Título.

CDD 005.1

---

A Deus.

Aos meus pais, Ana Sheila e Antônio Zênio.

A minha irmã, Aline Peixoto

Aos meus amigos.

## **AGRADECIMENTOS**

Queria agradecer primeiramente a Deus por ter me dado forças para suportar cada dia dessa jornada, e que sem ele com certeza não teria conseguido chegar até onde cheguei. Também queria agradecer a minha família, especialmente meus pais, Zênio e Sheila, que sempre deram forças, me apoiaram e fizeram de tudo pra que eu pudesse ter uma boa educação e ter uma boa postura como cidadão.

Queria agradecer a Juliana, ao Clerton e ao Pedro Júnior que contribuíram para a realização desse trabalho. E agradecer principalmente a minha orientadora Jéssyka por todo o apoio por toda instrução e por não desistir de mim. Muito obrigado!

Queria agradecer a todos os professores que tive em toda minha vida acadêmica, que sempre contribuíram para minha formação cidadã, para a evolução de meus conhecimentos e para que me tornasse um bom profissional. Também queria agradecer aos meus amigos que sempre se fizeram presente nos momentos bons e sempre estenderam a mão nos momentos difíceis. Queria agradecer a todos que acreditaram em mim, me deram apoio para que hoje eu pudesse estar aqui.

“A verdadeira motivação vem de realização, desenvolvimento pessoal, satisfação no trabalho e reconhecimento.”

(Frederick Herzberg)

## RESUMO

Contexto: Softwares nos quais uma falha acarreta em danos ao meio ambiente ou as pessoas são chamados de Sistemas Críticos. A qualidade desse tipo de software é um fator fundamental quando se trata de um sistema onde sua falha pode ter um alto grau de severidade. Um dos aspectos abordados pela área de qualidade de *software* é a definição de métricas para avaliação do processo de desenvolvimento bem como dos atributos de qualidade do sistema a ser desenvolvido. Entretanto, a avaliação da qualidade de sistemas críticos por meio de métricas é pouco explorada. Objetivo: este trabalho visa realizar um mapeamento sistemático de métricas para a avaliação de sistemas críticos. Método: as métricas e suas fórmulas de cálculo bem como os atributos de qualidade relacionados foram obtidas a partir do mapeamento sistemático da literatura. Resultados: o mapeamento retornou 805 trabalhos dos quais foram selecionados sete trabalhos que resultaram em 67 métricas de qualidade. Conclusões: Esta categorização pode contribuir para auxiliar os stakeholders como gerentes de projetos, analistas de segurança e engenheiros de software na tomada de decisão para a construção desse tipo de sistema.

**Palavras-chave:** Sistemas Críticos. Métricas. Qualidade de Software.

## **ABSTRACT**

Context: Software in which a failure causes damage to the environment or people are called Critical Systems. The quality of this type of software is a key factor when it comes to a system where its failure can have a high degree of severity. One of the aspects addressed by the area of software quality is the definition of metrics to evaluate the development process as well as the quality attributes of the system to be developed. However, the evaluation of the quality of critical systems by means of metrics is little explored. Objective: This work aims to perform a systematic mapping of metrics for the evaluation of critical systems. Method: the metrics and their calculation formulas as well as the related quality attributes were obtained from the systematic mapping of the literature. Results: The review returned 805 papers from which seven papers were selected which resulted in 67 quality metrics. Conclusions: This categorization can contribute to help stakeholders such as project managers, security analysts and software engineers in decision making for the construction of this type of system.

**Keywords:** Critical Systems. Metrics. Software quality.

## LISTA DE FIGURAS

Figura 1 – Pilares da confiança de um sistema.....	11
Figura 2 – Processo de mapeamento sistemática da literatura.....	14
Figura 3 – Metodologia.....	15
Figura 4 – Resultado da busca automática nas bibliotecas digitais.....	23
Figura 5 – Artigos por ano.....	26

## LISTA DE TABELAS

Tabela 1 – Comparação com trabalhos relacionados. ....	8
Tabela 2 – Perguntas de pesquisa. ....	16
Tabela 3 – Palavras-chave .....	17
Tabela 4 – Base de dados.....	18
Tabela 5 – Critérios de Seleção. ....	18
Tabela 6 – Formulário de extração dos dados. ....	19
Tabela 7 – Primeiros dados da aplicação do primeiro filtro. ....	23
Tabela 8 – Primeiros dados da aplicação do primeiro filtro. ....	23
Tabela 9 – Resultados da segunda etapa da leitura dos artigos.....	24
Tabela 10 – Informações de artigos da IEEE.....	25
Tabela 11 – Informações de artigos da Science. ....	26
Tabela 12 – Informações de artigos da Scopus.....	26
Tabela 13 – Informações de artigos da Springer.....	27
Tabela 14 – Categorização das métricas de Desempenho e Responsividade.....	28
Tabela 15 – Categorização de métricas de Disponibilidade. ....	31
Tabela 16 – Categorização de métricas de Exatidão. ....	33
Tabela 17 – Categorização de métricas de Manutenibilidade. ....	35
Tabela 18 – Categorização de métricas de Confiabilidade e Segurança.....	43

## LISTA DE ABREVIATURAS E SIGLAS

BLOC	<i>Bugs Per Line Of Code</i> (Bugs por Linha de Código)
CC	<i>Cyclomatic Complexity</i> (Complexidade Ciclomática)
CEG	<i>Cause graphic and effect</i> (Gráfico de causa e efeito)
CRU	<i>Computer Resource Utilization</i> (Utilização de Recursos de Computador)
CMM	<i>Capability Maturity Model Integration</i> (Modelo de maturidade de capacidade de software)
DD	Density of defects (Densidade de Defeitos)
FD	<i>Number of days of failure</i> (Número de dias de Falhas)
FP	<i>Function Point</i> (Análise de pontos de função)
MTTUF	<i>Mean Time to Unsafe Failure</i> (Tempo Médio para Falha Insegura)
PICOC	<i>Population, Intervention, Comparison, Outcome, Context</i> (População, Intervenção, Comparação, Resultados, Contexto)
RSCR	<i>Request for Request Stop</i> (Pedido de Parada de Requisição)
RT	<i>Requirements Traceability</i> (Requisitos de rastreabilidade)
SCS	<i>Safety-Critical Systems</i> (Sistemas críticos de segurança)
SDC	<i>Complexity of Systems Design</i> (Complexidade de Design de Sistemas)
TC	<i>Test Coverage</i> (Cobertura de Teste)
LOC	<i>Lines of code</i> (Linhas de código)
SLOC	<i>Source Lines of code</i> (Linhas de código de origem)

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>5</b>
1.1	Objetivos .....	6
1.1.1	<i>Objetivo geral</i> .....	6
1.1.2	<i>Objetivos específicos</i> .....	6
1.2	Trabalhos Relacionados .....	7
1.2.1	<i>Indicadores de Safety</i> .....	7
1.2.2	<i>Métricas de Safety</i> .....	7
1.2.3	<i>Segurança de processos nas indústrias</i> .....	8
1.2.4	<i>Métricas de dependabilidade e de security</i> .....	8
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>10</b>
2.1	Sistemas críticos .....	10
2.2	Qualidade de Software.....	11
2.3	Métricas de Qualidade.....	12
2.4	Mapeamento Sistemático.....	13
<b>3</b>	<b>PROCEDIMENTOS METODOLÓGICOS</b> .....	<b>15</b>
3.1	Realização do mapeamento sistemático da literatura.....	15
3.2	Categorização de Métricas .....	16
<b>4</b>	<b>PROTOCOLO</b> .....	<b>17</b>
4.1	Definição de objetivo.....	17
4.2	PICOC .....	17
4.3	Palavras-chave e Sinônimos .....	18
4.4	String de busca .....	18
4.5	Bases de dados .....	19
4.6	Critérios de Seleção.....	19
4.7	Formulário de extração de dados .....	20
4.8	Ameaças à validade .....	20
<b>5</b>	<b>RESULTADOS</b> .....	<b>22</b>
5.1	Primeira etapa da leitura dos artigos (Título e <i>Abstracts</i> ).....	23
5.2	Segunda etapa da leitura dos artigos (Leitura completa) .....	24
5.3	Visão Geral dos estudos.....	25
5.4	Extração das Métricas .....	27
5.5	Categorização das Métricas .....	28
5.5.1	<i>Métricas de desempenho e responsividade</i> .....	28
5.5.2	<i>Métricas de Disponibilidade</i> .....	31
5.5.3	<i>Métricas de Exatidão</i> .....	33
5.5.4	<i>Métricas de Manutenibilidade</i> .....	34
5.5.5	<i>Métricas de Segurança e Confiabilidade</i> .....	42
<b>6</b>	<b>CONCLUSÃO</b> .....	<b>55</b>
	<b>REFERÊNCIAS</b> .....	<b>58</b>

## 1 INTRODUÇÃO

A tecnologia vem crescendo a cada dia, fornecendo novidades e formas de facilitar cada vez mais as tarefas cotidianas. A automatização de atividades por meio de *softwares*, vem poupando a muitos de um trabalho desgastante, e assim vem ganhando um grande espaço em meio a organizações, instituições e muitos outros ambientes. Em alguns desses casos, *softwares* são responsáveis por controles financeiros, gestão de riscos e outras atividades que movimentam e cuidam de artefatos valiosos, assim sendo um grande mecanismo de segurança.

Apesar da sua ampla aceitação e uso, os *softwares* estão sujeitos a falhas (LEVESON, 2011). A gravidade de uma falha de *software* é relativa. Existem falhas com as quais usuários podem conviver, a tal ponto que o sucesso de aplicação de um produto não seja afetado. Por exemplo, uma falha ao carregar uma página, ou uma falha qualquer que não comprometa a segurança do sistema. Em outros casos, uma falha no programa representa um completo fracasso comercial. Como, por exemplo, uma falha em um sistema bancário onde possa gerar uma grave prejuízo financeiro.

*Softwares* nos quais uma falha acarreta em danos a algo ou alguém, são chamados de Sistemas Críticos, do inglês *Safety-Critical Systems* (SCS) (LEVESON, 2011). Dessa forma, existe um alto custo associado ao ocorrer uma falha nesse tipo de sistema, como por exemplo: (i) A máquina de radioterapia controlada por computador denominada Therac-25, que provocou overdose de seis pessoas (LEVESON, 1995); (ii) A queda de um avião turco DC-10, resultando em 346 mortes (LEVESON, 1995); (iii) o satélite *Milstar* que foi colocado em uma órbita elíptica baixa, incorreta e inutilizável, em oposição à órbita geossíncrona pretendida (LEVESON, 2011); (iv) A contaminação bacteriana de um abastecimento público de água que resultou na morte de sete pessoas e provocou doenças em metade das pessoas de uma cidade de 4.800 (LEVESON, 2011).

Nesses acidentes e muitos outros incidentes relacionados à segurança reportados, requisitos inadequados ou mal compreendidos foram reconhecidos como a causa principal (não a codificação ou a implementação (Lutz, 2000) de uma proporção significativa de acidentes (Simpson, 2002) e catástrofes relacionadas com a segurança (LEVESON, 2002).

Um pilar fundamental em um sistema crítico é a segurança, e para garantir a confiabilidade de um sistema, ele deve ser tolerante a falhas. A tolerância a falhas em um sistema considerado crítico deve considerar todos os aspectos, seja *Hardware*, *Software* ou qualquer peça do conjunto de produção (componentes e subsistemas por exemplo) e interação que possa gerar um problema futuro (LEVESON, 2011).

Para a construção de um sistema considerado crítico, devem ser utilizadas boas práticas da engenharia de *software* para desenvolver um sistema de qualidade. A qualidade em um *software* é um fator fundamental quando se trata de um sistema onde sua falha pode ter um alto grau de severidade. A garantia da qualidade vem de técnicas de uma das mais diversas áreas da

Engenharia de *Software*, que é a Qualidade de *Software*. Segundo a NBR ISO 9000:2005 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2005), qualidade “é o grau no qual um conjunto de características inerentes satisfaz aos requisitos”.

Um dos aspectos abordados pela área de qualidade de *software* é a definição de métricas para avaliação do processo de desenvolvimento bem como dos atributos de qualidade do sistema a ser desenvolvido. A definição de métricas é essencial para avaliação de qualquer tipo de sistema, mas quando se trata de sistemas críticos, é algo de fundamental importância. Segundo Amorim (2017), isto ocorre, uma vez que a adoção de métricas contribui para o sucesso do projeto bem como para tornar o sistema mais confiável. Além disso, auxiliam na tomada de decisão em umas das mais importantes etapas da construção de um sistema, que é o planejamento.

O estudo de métricas para a qualidade de software vem sendo investigado em diversos trabalhos. Saraiva et al. (2015) analisa métricas para a avaliação da manutenibilidade de *software*; Bezerra (2016) define medidas para avaliação da manutenibilidade do modelo de *features* de linhas de produto de *software* tradicionais e dinâmicas; e, Nascimento (2017) propõe um catálogo de métricas para avaliação da qualidade de ambientes integrados de computação em nuvem, redes definidas por *software* e virtualização de funções de rede.

Entretanto, a avaliação da qualidade de sistemas críticos por meio de métricas é pouco explorada (CRUICKSHANK, MICHAEL e SHING, 2009). Sendo assim, o objetivo desse trabalho é realizar um mapeamento sistemático de métricas para a avaliação de sistemas críticos. O resultado contribuirá para auxiliar os *stakeholders* como gerentes de projetos, analistas de segurança e engenheiros de *software* na tomada de decisão para a construção desse tipo de sistema.

## **1.1 Objetivos**

Nessa seção são apresentados o objetivo geral e os objetivos específicos trabalho.

### **1.1.1 Objetivo geral**

Este trabalho tem como objetivo mapear o estado da arte sobre métricas de qualidade para sistemas críticos.

### **1.1.2 Objetivos específicos**

- Conduzir um mapeamento sistemática da literatura sobre métricas de qualidade para sistemas críticos;
- Obter métricas de qualidade relacionadas a sistemas críticos;

- Propor uma categorização das métricas;

## 1.2 Trabalhos Relacionados

A literatura de sistemas críticos apresenta alguns trabalhos sobre métricas de sistemas críticos referentes ao processo de desenvolvimento, de produto ou de formas de trabalho em indústrias. Entretanto, não foi encontrado trabalhos que realizem uma sistematização das métricas existentes. Sendo assim, os trabalhos serão discutidos com relação a essas perspectivas.

### 1.2.1 Indicadores de Safety

Na literatura de indicadores de *Safety*, Mendeloff et al. (2013), em seu artigo, examinam os possíveis usos de métricas de eventos da segurança de processos propostas pela indústria de petróleo americana publicadas na norma ANSI/API 754. Pasman e Rogers (2014) mostram que ao utilizar uma nova técnica de redes bayesianas de gestão de riscos, o progresso de melhoria de indicadores pode ser realizado. Dessa forma, segundo os autores, parece possível relacionar taxas de falhas técnicas com fatores de risco que atuam ao longo do tempo e agir antes que algum evento indesejado aconteça. Swuste et al. (2015), em seu artigo, realizam uma diferenciação entre indicadores de segurança que estimulam (*leading*) e aqueles que atrapalham (*lagging*). Além disso, os autores questionam essa distinção, assim como a quantificação de indicadores de segurança.

### 1.2.2 Métricas de Safety

Na literatura sobre métricas de *Safety*, Cruickshank, Michael e Shing (2009), em seu artigo, apresentam um framework de métricas de validação, que se baseia em um novo modelo para validação de requisitos de segurança de *software*. Karanikas (2016) discute vários pontos de vista referentes a aplicabilidade, viabilidade e adequação de ferramentas para medição de desempenho de segurança. Pontos de vista de gerentes e especialistas em segurança foram considerados em uma pesquisa que foi conduzida em uma grande organização de aviação. Varkoi (2013) estuda o escopo do processo aplicável a requisitos de segurança para *software* com base em um novo conceito de características de qualidade e atributos de qualidade de processo para segurança.

### 1.2.3 Segurança de processos nas indústrias

Na literatura de segurança de processos industriais, Knegetering e Pasman (2008), em seu artigo, abordam o aparecimento de muitas falhas fora do escopo das técnicas de segurança de processos em indústrias, surgindo, assim a necessidade de um novo gerenciamento de controle de segurança. Nesse trabalho ele realiza um embasamento de observações, ilustra características típicas da situação atual e indicam direções que poderiam levar a resolver novos problemas de segurança do processo.

### 1.2.4 Métricas de dependabilidade e de security

Na literatura ao que se refere a métricas de dependabilidade e de *security*, DeLong, Smith e Johnson (2005) em seu artigo, argumentam que para a avaliação de um sistema crítico, métricas como confiabilidade e tempo médio de falha, são bem aceitos como um meio para quantificar os atributos tolerantes a falhas de um sistema com uma taxa de falha associada. Assim, os autores desenvolveram duas métricas que podem ser usadas na avaliação de sistemas críticos, a segurança em estado estacionário (*steady-state safety*) e o tempo médio para falha não-segura, do inglês *Mean Time To Unsafe Failure* (MTTUF).

Tabela 1 – Comparação com trabalhos relacionados.

<b>Critério/ Trabalho</b>	<b>DeLong, Smith e Johnson (2005)</b>	<b>Karanikas (2016)</b>	<b>Trabalho Proposto</b>
<b>Domínio</b>	Safety	Safety	Safety
<b>Tipo de proposta</b>	Propõe duas métricas	Conjunto de métricas	Conjunto de métricas
<b>Abordagem</b>	Métricas de produto	Métricas de processo	Métricas de produto, processo, projeto
<b>Método de Pesquisa</b>	Baseado na esperiencia	Revisão não sistemática da literatura e estudo de caso	Mapeamento sistemático da literatura

Fonte: Autor

Esse trabalho está estruturado da seguinte forma: no Segundo capítulo, são discutidos os principais trabalhos relacionados; no, são apresentados os objetivos gerais e específicos; o

Segundo sumariza a fundamentação teórica; o Terceiro descreve os procedimentos metodológicos. No Quarto, é apresentado o protocolo do mapeamento sistemático. O Quinto discute os resultados obtidos e o Sexto as conclusões e trabalhos futuros

## 2 FUNDAMENTAÇÃO TEÓRICA

Nessa seção serão apresentados os conceitos-chaves abordados nessa pesquisa. A seção 2.1 apresentará o conceito de Sistemas Críticos; a seção 2.2 contempla Qualidade de *software*; e, na seção 2.3, são discutidas Métricas de Qualidade.

### 2.1 Sistemas críticos

*Softwares* são utilizados para diversas finalidades comerciais, poupando serviços manuais e os substituindo por serviços automatizados. Esses *softwares*, em muitas das vezes, são críticos, do inglês *safety-critical*, onde uma falha nesses sistemas pode provocar danos financeiros, físicos ou ambientais. Portanto, é imprescindível que esses tipos de sistemas sejam bem planejados e implementados para fornecer a segurança necessária para lidar com falhas que venham ocorrer.

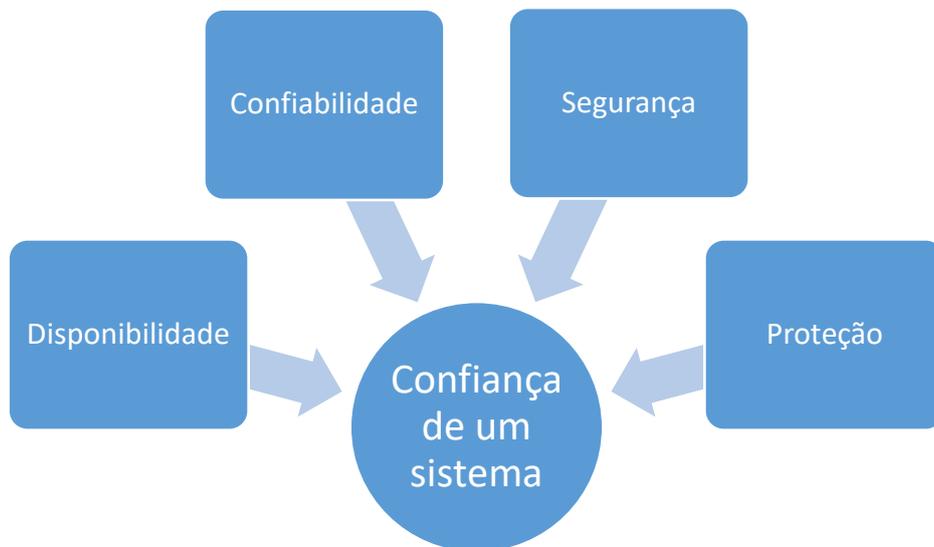
A segurança do sistema tem por objetivo minimizar o risco de acidentes e, se possível, evitá-los. Isso deve ser feito com custos razoáveis usando uma abordagem sistemática. Testar a segurança no sistema não é suficiente para alcançar a segurança. Um sistema deve ser projetado para ser seguro desde o começo do processo de desenvolvimento (LEVESON, 1995). Como benefícios de um sistema confiável, pode-se citar a redução de danos ao meio ambiente, salvar vidas humanas e economia de recursos uma vez que as consequências de acidentes ou incidentes são geralmente muito caras e podem originar custos indiretos (perda de reputação, etc.).

Segundo LEVESON (1995), a segurança do sistema pode ser afetada de duas maneiras:

- pode exibir comportamento em termos de valores de entrada e temporização que contribuem para que o sistema atinja um nível perigoso (LEVESON, 1995).
- pode falhar em reconhecer ou lidar com falhas de *hardware* quando é necessário que responda de maneira eficiente (LEVESON, 1995).

A confiança em um sistema se divide em quatro pilares fundamentais (SOMMERVILLE, 2003) conforme ilustrado na Figura 1. O primeiro desses pilares fundamentais é a disponibilidade, que é a capacidade do sistema estar disponível sempre que for necessário. O segundo é a confiabilidade, que é a capacidade do sistema de realizar as funções de acordo com o esperado, isto é, cumprir de forma confiável o que tem que ser realizado. Segurança é a capacidade de operar sem falhas catastróficas. Finalmente, a proteção, é a capacidade de proteção contra invasão acidental.

Figura 1 – Pilares da confiança de um sistema.



Fonte: Elaborada pelo Autor adaptado de (SOMMERVILLE, 2003).

Esses sistemas que não podem ser propensos a falhas, são chamados de sistemas críticos que podem ser classificados em três tipos (SOMMERVILLE, 2003):

- *Safety-critical*: sistemas nos quais uma falha pode acarretar em algum dano físico, ambiental, entre outros.
- Críticos a Missão: quando não se completa uma missão ou comportamento desejado por falha do mesmo.
- Críticos ao Negócio: onde sua falha resulta em prejuízos financeiros que podem resultar no fracasso em organizações ou outro tipo de entidade.

## 2.2 Qualidade de Software

No contexto de desenvolvimento de *software*, qualidade pode ser entendida como um conjunto de características a serem satisfeitas em um determinado grau, de modo que o produto de *software* atenda às necessidades explícitas e implícitas de seus usuários (ROCHA, 1994).

Na medida em que cresce a demanda por sistemas complexos, com grande responsabilidade no contexto das organizações, a qualidade desponta como um fator essencial no desenvolvimento de *software* (DUARTE e FALBO, 2000). Dessa forma, cada vez mais vem aumentando os motivos para se investir em qualidade de *software*. Contudo, o que ainda continua sendo uma das maiores dificuldades para a implantação de programas de qualidade, é realmente compreender o que é a qualidade de *software*.

Para que a qualidade seja avaliada, é preciso haver meios dela ser medida. Ou seja, é preciso obter uma medida que especifique em quantidade o grau de uma característica de qualidade. Contudo, para computar uma característica de qualidade, é necessário determinar uma métrica capaz de quantificá-la e fazer uma medição para determinar a medida, resultado da aplicação da métrica. Por exemplo, supondo que se deseja saber o tamanho de determinado produto de *software*. Então, a métrica utilizada poderia ser “a quantidade de linhas de código”. A medição poderia ser “contar as linhas de código, desconsiderando todos os comentários, de cada programa contido no *software*”. Dessa forma, a medida seria o número obtido na medição, uma quantidade, por exemplo, “5000 linhas de código” (DUARTE e FALBO, 2000).

### 2.3 Métricas de Qualidade

Na engenharia de qualquer sistema de *hardware*, o termo "melhor prática de engenharia" é aplicado repetidamente em todos os aspectos do desenvolvimento do sistema, seja ele uma ponte, um automóvel ou um grande edifício (MUNSON, 2003). De acordo com o autor, a melhor prática de engenharia incorpora uma longa tradição de experimentação, análise e medição. É a aplicação de princípios científicos para a solução de um projeto complexo de projeto e desenvolvimento de *hardware*. No núcleo da melhor prática de engenharia, está a medição e a validação empírica dos resultados do projeto.

Métricas são formas de medir determinadas características de um *software* ligado à sua estrutura ou funcionamento. O objetivo de utilizá-las é avaliar e revisar dados e informações (SOMMERVILLE, 2011).

Segundo MUNSON (2003), ao se tratar de métricas de *software*, deve-se observar que há duas pré-condições para a viabilidade desse tipo de pesquisa. A primeira é se essas métricas são capazes de serem reproduzidas, ou seja, é possível a repetição das informações extraídas na

verificação. A segunda é a validação que permite verificar se é viável o uso dessas métricas para a tomada de decisão em um projeto.

De acordo com Pressman (2011), as métricas de *software* são definidas em 3 tipos:

- **Métricas de Produto** - são usadas para medir propriedades do *software* contribuindo para melhorar a qualidade dos componentes existentes do *software*. Alguns exemplos dessas métricas são as métricas de tamanho, complexidade, pontos de função, entre outras;
- **Métricas de Processo** - usadas na gestão, sendo utilizadas para medir certos atributos e propriedades do processo usado para se obter o *software* a fim de fornecer um conjunto de indicadores de processo que leve ao aperfeiçoamento do mesmo. Exemplos dessas métricas são as métricas de custo e esforço que podem auxiliar a verificar se a execução do projeto está de acordo com o cronograma;
- **Métricas de Projeto** - usadas de forma mais táticas, por um gerente ou equipe de *software*, podendo ser usadas ao longo de todo o desenvolvimento do projeto.

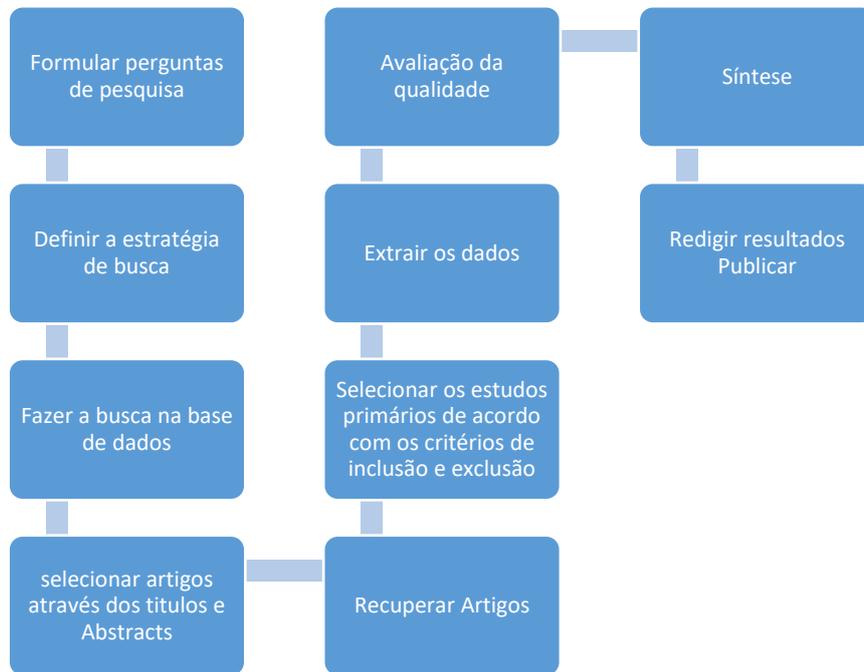
O método de pesquisa adotado neste trabalho é o mapeamento sistemático da literatura definida na seção a seguir.

## 2.4 Mapeamento Sistemático

Estudos secundários visam revisar estudos primários relativos a certas questões de pesquisa, com o objetivo específico de integrar e sintetizar evidências relacionadas a essas questões. De maneira geral, o objetivo de um estudo secundário é prover a pesquisadores uma visão geral de uma área de pesquisa e ajudar a identificar lacunas na pesquisa nessa área (WHOLIN et al., 2013). Revisão Sistemática e Mapeamento Sistemático são tipos de estudos secundários que seguem um processo de pesquisa metodologicamente bem definido para identificar, analisar e interpretar as evidências disponíveis relacionadas a um particular conjunto de questões de pesquisa, tópico ou fenômeno de interesse, de uma maneira não tendenciosa e, até certo grau, repetível (KITCHENHAM; CHARTERS, 2007).

O método de mapeamento sistemático da literatura é realizado por meio de uma série de etapas essenciais para que a pesquisa seja bem-sucedida. A Figura 2 ilustra a sequência de passos realizados em um mapeamento sistemático.

Figura 2 – Processo do mapeamento sistemático da literatura.



Fonte: Adaptado de (KITCHENHAM, CHARTERS, 2007).

O processo de um mapeamento sistemático começa por formular as perguntas de pesquisa. Logo após, deverão ser definidas as estratégias que serão utilizadas para realizar a busca pelos estudos. Após ser definida a estratégia de busca, são realizadas as buscas nas bases de dados de pesquisas científicas. Em seguida, ocorre o processo de identificação dos estudos obtidos, pelo título e *abstract*, e também ocorre a recuperação dos estudos. O próximo passo é selecionar os estudos de acordo com os critérios de inclusão e exclusão pré-definidos anteriormente. Então, devem ser extraídos os dados desejados da pesquisa, a partir dos estudos selecionados. Para que a pesquisa seja bem-sucedida, esses dados devem ser avaliados e validados, e só após devem ser publicados.

Um mapeamento sistemático é uma forma de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma questão de pesquisa particular. Uma das razões para a realização de revisões sistemáticas é que esta resume as evidências existentes em relação a um tratamento ou tecnologia (KITCHENHAM, 2004).

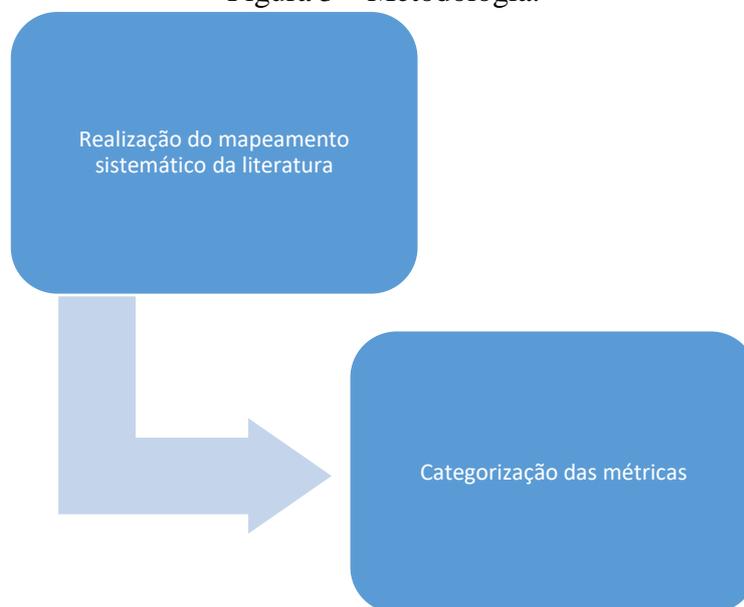
O mapeamento sistemático da literatura é o método de pesquisa empírico adotado neste trabalho para coleta dos dados. No próximo capítulo, a metodologia deste trabalho é descrita.

### 3 PROCEDIMENTOS METODOLÓGICOS

O processo de construção desse trabalho foi realizado em 2 (duas) etapas: Realização do mapeamento sistemático da literatura na qual foi analisado o estado da arte referente a métricas de qualidade de sistemas críticos junto com a consolidação dos dados na qual as métricas serão analisadas e normalizadas.

A segunda etapa é de Categorização das métricas para obter os resultados. Ao decorrer desse capítulo, estas etapas serão explicadas de forma mais detalhada. A Figura 3 apresenta a divisão dessas etapas do processo da pesquisa.

Figura 3 – Metodologia.



Fonte: Autor.

#### 3.1 Realização do mapeamento sistemático da literatura

O mapeamento sistemático da literatura é realizado para extrair, de estudos relacionados, informações relevantes para uma busca específica, assim, conseguindo um conjunto de informações que tenham fundamento. Nessa etapa, serão analisados estudos sobre

os diferentes tipos métricas de qualidade para sistemas críticos disponíveis. O planejamento e condução do mapeamento sistemático será apoiado pela ferramenta Parsifal<sup>1</sup>.

Ao fim do procedimento do mapeamento, já contendo as métricas, será realizada a submissão das mesmas ao processo de consolidação dos dados. O termo consolidação nesse trabalho deve ser considerado como um resumo consistente dos dados extraídos no mapeamento sistemática. Ambiguidades e inconsistências na nomenclatura das métricas é uma questão importante reportada na literatura como no trabalho de Saraiva et al. (2015).

O processo de consolidação visa remover a ambiguidade e inconsistência de nomes de as métricas obtidas. Segundo Saraiva et al. (2015), em casos desse tipo de estudo, são encontrados dois tipos de inconsistência, que em muitas das vezes são métricas com nomes diferentes, mas que possuem o mesmo significado; e métricas que com nomes semelhantes, mas com sentidos totalmente diferentes. Nesse caso, será realizado uma análise para tornar os dados mais consistentes, reduzir a ambiguidade e consolidar os dados, para que estejam preparados para serem categorizados.

### **3.2 Categorização de Métricas**

Após a etapa de consolidação dos dados, as métricas de qualidade relacionadas a sistemas críticos estarão disponíveis. Entretanto, apenas uma lista de métricas não é suficiente para auxiliar na tomada de decisão na construção de um sistema. Para isso, é necessário obter um conjunto que se torne uma informação útil para a adoção dessas métricas, e esse conjunto pode ser obtido por meio de uma categorização dessas métricas.

Segundo Saraiva et al. (2015), a ideia principal para a categorização, é analisar as métricas encontradas no mapeamento sistemático como ponto de partida para o processo de apoio a tomada de decisão sobre quais métricas podem ser adotadas em um projeto.

---

<sup>1</sup><https://parsif.al/>

## 4 PROTOCOLO

O protocolo em um mapeamento sistemático serve como o processo a ser conduzido o mapeamento, para que seja concluída com êxito. Nessa seção, será discutido em detalhes o passo a passo do protocolo que foi utilizado na realização do mapeamento sistemático deste trabalho.

### 4.1 Definição de objetivo

O objetivo deste mapeamento é identificar estudos que contenham métricas de qualidade para sistemas críticos.

### 4.2 PICOC

O método PICOC (População, Intervenção, Comparação, Resultados, Contexto), do inglês *Population, Intervention, Comparison, Outcome, Context*, foi utilizado no protocolo para conduzir a determinação das perguntas de pesquisas.

- **População:** Publicações revisadas aos pares que apresentam métricas de qualidade para sistemas críticos.
- **Intervenção:** Coletar evidencia empírica em relação as perguntas de pesquisa.
- **Comparação:** Não se aplica, pois, os estudos primários não serão comparados.
- **Resultados:** Respostas das perguntas de pesquisa.
- **Contexto:** Qualquer contexto que apresente métrica de qualidade para sistemas críticos.

Tendo em vista o propósito deste mapeamento sistemático e os critérios PICOC, foram respondidas as perguntas de pesquisa definidas na Tabela 2.

Tabela 2– Pergunta de pesquisa.

Pergunta de Pesquisa	Objetivos e motivações
P1. Quais são as métricas de qualidade usadas no desenvolvimento de sistemas críticos?	O objetivo da questão é identificar as métricas, seus atributos, formas de cálculo e características de qualidade que são apresentados no estudo selecionado.

Fonte: Autor.

### 4.3 Palavras-chave e Sinônimos

Palavras-chave são consideradas o principal instrumento de um mapeamento sistemático. São termos que são compostos por uma ou mais sinônimos usados para maximizar os resultados da busca. A Tabela 3 apresenta as palavras chaves que foram utilizadas no mapeamento.

Tabela 3 – Palavras-chave.

<b>Palavras-Chave</b>	<b>Sinônimos</b>
<i>Software Quality</i>	<i>Software engineering</i>
Quality Evaluation	Quality Attribute, Quality Characteristic, Quality indicator, Quality Measure, Quality metric
Safety-critical systems	Safety-Critical <i>Software-Intensive Systems</i> , <i>software-intensive Systems</i>

Fonte: Autor.

### 4.4 String de busca

A *string* de busca é um termo gerado a partir da concatenação das palavras-chave e esse termo que é utilizado para a busca realizadas nas bases de dados. Inicialmente, foram realizados alguns testes interagindo com as bases de dados para aprimorar a *string*. Foram retiradas algumas palavras nas quais sua inclusão não retornava documentos relevantes para a pesquisa. Em seguida foi utilizada a seguinte *string* para a busca automática nas bases de dados:

*("software quality" OR "software engineering") AND ("quality attribute" OR "quality characteristic" OR "quality indicator" OR "quality measure" OR "quality metric") AND ("safety-critical systems" OR "safety-critical software-intensive systems" OR "software-intensive systems")*

#### 4.5 Bases de dados

Todas as pesquisas foram realizadas em bases de dados renomadas que armazenam conteúdo científico. A Tabela 4 apresenta as bases de dados nas quais foram realizadas as buscas deste mapeamento sistemático.

Tabela 4 – Base de dados.

Fontes	URL
ACM	<a href="https://dl.acm.org/">https://dl.acm.org/</a>
IEEE	<a href="https://ieeexplore.ieee.org">https://ieeexplore.ieee.org</a>
Science direct	<a href="https://www.sciencedirect.com/">https://www.sciencedirect.com/</a>
Scopus	<a href="https://www.scopus.com">https://www.scopus.com</a>
Springer	<a href="https://link.springer.com">https://link.springer.com</a>

Fonte: Autor.

#### 4.6 Critérios de Seleção

O objetivo de ter critérios de seleção é obter um grupo de estudos que possam ajudar a responder às perguntas de pesquisa. A Tabela 5 apresenta os critérios de inclusão e exclusão que foram utilizados nesta pesquisa.

Tabela 5 – Critérios de Seleção.

Critérios de inclusão	<ul style="list-style-type: none"> <li>• Estudo publicado em qualquer ano até julho de 2018;</li> <li>• Estudos primários;</li> <li>• Estudos que apresentam métricas de qualidade para sistemas críticos;</li> <li>• Estudos que respondam alguma pergunta de pesquisa.</li> </ul>
Critérios de exclusão	<ul style="list-style-type: none"> <li>• Estudos claramente irrelevantes para a pesquisa, levando em conta as perguntas de pesquisa;</li> <li>• Estudos cujo foco não seja métricas de qualidade para sistemas críticos;</li> <li>• Estudos duplicados (apenas uma cópia de cada estudo será incluída);</li> <li>• Estudos secundários;</li> <li>• Estudos curtos (<math>\leq 3</math> páginas);</li> <li>• Estudo redundante de mesma autoria;</li> <li>• Publicações cujo texto não estava disponível (através de bases de busca ou entrando em contato com os autores);</li> <li>• Estudos não escritos em Inglês.</li> </ul>

Fonte: Autor.

#### 4.7 Formulário de extração de dados

Para armazenar todas as informações necessárias para responder às perguntas da pesquisa foi preparado o formulário de extração de dados apresentado na Tabela 6.

Tabela 6 – Formulário de extração dos dados.

Formulário de extração de dados	Descrição	Pergunta de Pesquisa
Fonte do Artigo	ACM, Springer, IEEE, Science Direct, Scopus	Visão geral do estudo
Autores, ano, título, país		Visão geral do estudo
Filiação		Visão geral do estudo
Cálculo da métrica		P1
Referência		P1
Unidade métrica		P1
Nome da métrica		P1
Tipo da métrica		P1

Fonte: Autor.

#### 4.8 Ameaças à validade

A classificação de ameaças à validade descrita por (WOHLIN et al., 2000) foi utilizada para discutir as ameaças deste trabalho. Esta classificação define quatro tipos de ameaças de validade, sendo elas, ameaças de conclusão, internas, de construção e de validade externa.

*Ameaças à validade de constructo:* Este tipo de validade diz respeito à generalização do resultado para o conceito ou teoria por trás da execução do estudo. Com o objetivo de minimizar ameaças dessa natureza, foi utilizado sinônimos para as principais palavras chaves.

*Ameaças de validade interna:* estão relacionadas a uma possível conclusão errada sobre as relações causais entre o tratamento e o resultado (WOHLIN et al., 2000). Decisões subjetivas

podem ocorrer durante a seleção de artigos e extração de dados uma vez que é comum estudos primários não fornecerem uma descrição clara ou objetivos e resultados apropriados, dificultando a aplicação objetiva dos critérios de inclusão/exclusão ou a imparcialidade. A fim de minimizar erros de seleção e extração, o processo de seleção será realizado de forma iterativa de forma que quando ocorrer dúvida na aplicação de algum critério, o estudo não será eliminado e passará para a próxima fase. Além disso, o processo de seleção será realizado de forma colaborativa pelo aluno e por colaboradores de forma que conflitos sejam discutidos e solucionados pelos alunos em conjunto com a orientadora. Dessa forma, objetiva-se atenuar as ameaças devido ao viés pessoal na compreensão do estudo.

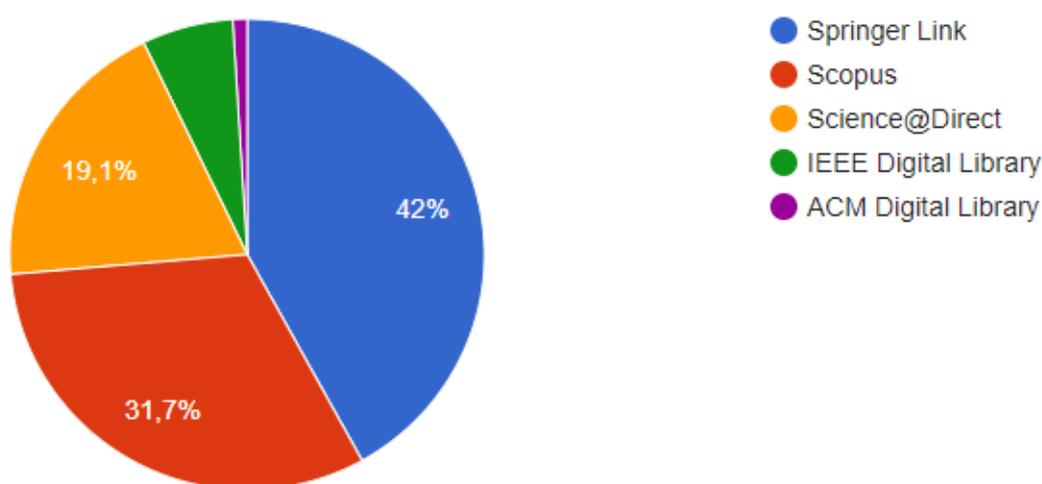
*Ameaças à validade externa:* está relacionada ao grau em que os estudos primários serão representativos para o tópico do mapeamento. Se tratando de um mapeamento sistemático da literatura, a validade externa depende da literatura identificada: se a literatura identificada não é válida externamente, tampouco a síntese do seu conteúdo é citada. Esta ameaça será mitigada devido a utilização do critério de exclusão para eliminar, da pesquisa, os estudos provenientes de literatura cinza. Além disso, para mitigar as ameaças externas, o protocolo de pesquisa foi definido iterativamente e, validado, com o consenso do autor, dos colaboradores e da orientadora.

*Ameaças à validade de conclusão:* A metodologia descrita por (KITCHENHAM; CHARTERS, 2007) assume que nem todos os estudos primários relevantes possam vir a ser identificados. Para amenizar os efeitos dessa ameaça, o processo do mapeamento foi cuidadosamente elaborado e discutido pelos autores para minimizar o risco de exclusão estudos relevantes. Outro método empregado foi utilizar expressões e palavras sinônimas para os constructos deste mapeamento sistemático, essa técnica objetiva uma maior cobertura de estudos possivelmente importantes a partir da pesquisa automática. Além disso, o processo de seleção do estudo será conduzido em paralelo e de forma independente pelo aluno e por colaboradores. Posteriormente, os resultados foram harmonizados para mitigar o viés pessoal na seleção do estudo causado por revisores individuais. Finalmente, a orientadora supervisionou esse processo.

## 5 RESULTADOS

No mapeamento foram realizadas as buscas nas bases de dados utilizando a *string* de busca definida na Seção 3.4, nas bibliotecas digitais ACM, IEEE, Science direct, Scopus e Springer, que ao total retornaram 805 artigos relacionados. A Figura 4 apresenta a quantidade de trabalhos científicos encontrados em cada base de dados.

Figura 4 – Resultado da busca automática nas bibliotecas digitais.



Fonte: Autor.

Logo após os artigos selecionados para o mapeamento serem retornados pelas bases de dados, os mesmos foram submetidos na *Parsifal*, que foi a ferramenta usada para realizar os passos do mapeamento. Após todas as etapas da condução do mapeamento, foram selecionados um total de 7 artigos, totalizando 73 métricas relacionadas a sistemas críticos. Após a seleção dos artigos e a extração das métricas, foi realizada a consolidação dos dados, onde foi retirada ambiguidades e inconsistências existente entre as métricas coletadas. Na Fase de categorização das métricas, as métricas foram agrupadas por atributos e indicadores de qualidade relacionados ao domínio de sistemas críticos.

As seções a seguir apresentam o passo a passo da condução do mapeamento sistemático, onde foi realizado a leitura dos artigos, a extração das métricas e a categorização das mesmas.

### 5.1 Primeira etapa da leitura dos artigos (Título e *Abstracts*)

A primeira filtragem realizada na condução do mapeamento consistiu na leitura dos títulos e *abstracts* dos 805 artigos retornados pela *string* de busca. Dentre esses artigos, 718 eram artigos não duplicados que foram lidos por todos participantes do mapeamento de forma independente. A Tabela 7 mostra os dados numéricos da primeira etapa de seleção.

Tabela 7 – Primeiros dados da aplicação do primeiro filtro.

<b>Base de Dados</b>	<b>Artigos</b>	<b>Duplicados</b>	<b>Leitura dos <i>Abstracts</i></b>
<b>ACM</b>	8	6	2
<b>IEEE</b>	51	8	43
<b>Science</b>	156	3	153
<b>Scopus</b>	260	29	231
<b>Springer</b>	330	41	289
<b>Total</b>	805	87	718

Fonte: Autor.

Após o termino da leitura de cada base de dados, os participantes do mapeamento em conjunto, analisaram e compararam os artigos aceitos por cada um, e foi calculado o percentual de concordância entre os artigos rejeitados. A Tabela 8 mostra os resultados extraídos da primeira etapa da leitura dos títulos e abstracts dos artigos.

Tabela 8 – Resultados da primeira etapa da leitura dos artigos.

<b>Base de dados</b>	<b>Etapa 1: Leitura de título e abstract</b>				<b>Resultado</b>	
	<b>P1 (rejeitou)</b>	<b>P2 (rejeitou)</b>	<b>P3 (rejeitou)</b>	<b>% concordância</b>	<b>Rejeitados</b>	<b>Próxima fase</b>
<b>ACM</b>	2	2	2	100%	2	0

<b>IEEE</b>	36	35	37	97,67%	36	7
<b>Science</b>	129	125	124	94,77%	129	24
<b>Scopus</b>	201	204	206	98,27%	205	26
<b>Springer Link</b>	260	264	248	95,16%	262	27
<b>Total</b>	624	630	617	96,3%	634	84
<b>Legenda: P= Participante</b>						

Fonte: Autor.

Após o termino da etapa de leitura dos títulos e abstracts, foram definidos os artigos selecionados para a fase de leitura completa. A seção a seguir apresenta os passos dessa etapa e os resultados obtidos.

## 5.2 Segunda etapa da leitura dos artigos (Leitura completa)

A segunda etapa do processo de seleção dos artigos foi realizada por todos os participantes do mapeamento, mas diferente da primeira etapa, os participantes dividiram a leitura dos artigos por base de dados, assim, cada um realizando a leitura de forma individual em base de dados diferentes. Após o termino da leitura dos artigos, os participantes em conjunto analisaram os critérios de aceitação uns dos outros e definiram os artigos aptos a passar para a fase da extração das métricas. A Tabela 9 apresenta os resultados da segunda etapa da leitura dos artigos.

Tabela 9 – Resultados da segunda etapa da leitura dos artigos

Base de dados	Artigos	Resultados				
		P1 (rejeitou)	P2 (rejeitou)	P3 (rejeitou)	Rejeitados	Próxima fase
<b>ACM</b>	<b>0</b>	-	-	-	-	-
<b>IEEE</b>	<b>7</b>	-	5	-	5	2
<b>Science</b>	<b>24</b>	-	22	-	22	2
<b>Scopus</b>	<b>26</b>	-	-	24	24	2

<b>Springer Link</b>	<b>27</b>	26	-	-	26	1
<b>Total</b>	<b>84</b>	26	27	24	77	7

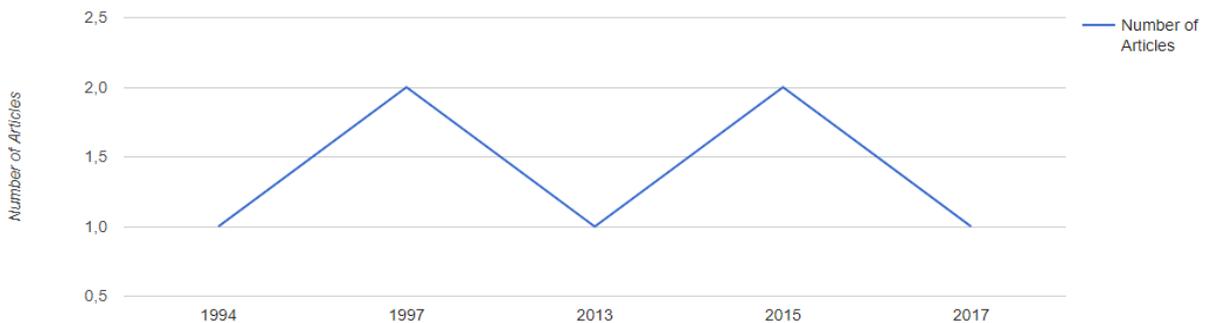
Fonte: Autor.

A seção a seguir descreve o procedimento adotado para extração das métricas a partir dos sete artigos selecionados.

### 5.3 Visão Geral dos estudos

Dentre os artigos selecionados, Figura 5 mostra os artigos publicados por ano.

Figura 5 – Artigos por ano



Fonte: Autor.

Objetivando manter rastreabilidade das informações coletadas, são apresentadas nas tabelas a seguir os títulos dos artigos selecionados, autores e suas afiliações. A Tabela 10 apresenta informações sobre os estudos selecionados da base de dados IEEE.

Tabela 10 – Informações de artigos da IEEE

<b>Título</b>	<b>Autor</b>	<b>Filiação</b>
<i>Formalize the software quality measurement for heterogeneous requirements</i>	E. Mit and C. W. Shiang	Faculty of Computer Science and Information, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Malaysia
<i>Software metrics model for quality control</i>	N. F. Schneidewind	Naval Postgraduate Sch., Monterey, CA, USA

Fonte: Autor.

A Tabela 11 apresenta informações dos artigos extraídos da base de dados Science Direct.

Tabela 11 - Informações de artigos da Science.

<b>Science@Direct</b>		
<b>Título</b>	<b>Autor</b>	<b>Filiação</b>
<i>A software metric set for program maintenance management</i>	George E Stark and Louise C Kern and C.W Vowell	The MITRE Corporation, Houston, Texas U.S.A, and NASA Johnson Space Center, Houston, Texas U.S.A
<i>A three-stage analysis of IDS for critical infrastructures</i>	Lorena Cazorla and Cristina Alcaraz and Javier Lopez	Computer Science Department, University of Malaga, Spain

Fonte: Autor.

A Tabela 12 apresenta informações sobre os artigos extraídos da base de dados Scopus.

Tabela 12 - Informações de artigos da Scopus

<b>Scopus</b>		
<b>Título</b>	<b>Autor</b>	<b>Filiação</b>
<i>Taxonomy of quality metrics for assessing assurance of security correctness</i>	Ouedraogo, M. and Savola, R.M. and Mouratidis, H. and Preston, D. and Khadraoui, D. and Dubois, E.	Service Science and Innovation Department (SSI), Public Research Center Henri Tudor, 1855 Kirchberg, Luxembourg; School of Architecture, Computing and Engineering, University of East London, London, United Kingdom; VTT Technical Research Centre of Finland, Oulu, Finland
<i>Metric-based software reliability prediction approach and its application</i>	Shi, Y. and Li, M. and Arndt, S. and Smidts, C.	NASA Goddard Space Flight Center, Greenbelt, MD, United States; U.S. Nuclear Regulatory Commission, Washington, DC, United States; The Ohio State University, 281 W. Lane Ave, Columbus, OH, United States

Fonte: Autor.

A Tabela 13 apresenta informações dos artigos extraídos da base de dados Springer.

Tabela 13 - Informações de artigos da Springer.

<b>Springer Link</b>		
<b>Título</b>	<b>Autor</b>	<b>Filiação</b>
<i>The Role of Measurement in Software Safety Assessment</i>	Fenton, Norman	Centre for <i>Software Reliability</i> City University UK

Fonte: Autor.

#### 5.4 Extração das Métricas

Após a última etapa de seleção dos artigos, foram extraídas as métricas e as suas respectivas informações. Após selecionadas, as métricas foram submetidas à etapa de categorização, onde foram avaliadas e agrupadas conforme o planejado.

A extração das métricas foi realizada a partir dos critérios de serem métricas relacionadas ao domínio de sistemas críticos de segurança e que apresente o mínimo de informações relevantes para o estudo. A maioria dos artigos selecionados apresentam características de qualidade relacionadas a essas métricas. Portanto, o objetivo da categorização foi agrupar essas métricas de acordo com suas características de qualidade reportadas nos artigos.

Para cada métrica, foram especificadas as seguintes informações baseadas nas informações gerais do formulário de extração de métricas proposto por Bezerra (2016):

- **Nome da Métrica:** É o nome que identifica a métrica categorizada.
- **Tipo da Métrica:** Classifica a métrica por seu tipo, que pode ser métrica de produto, métrica de processo e métrica de projeto.
- **Descrição da Métrica:** Descreve a métrica em função do aspecto avaliado, esclarecendo sua característica e seu objetivo de medição.
- **Formula de Cálculo:** Demonstra em forma de equações, quando disponíveis, a forma em que essas métricas são calculadas, definindo os atributos participantes da mesma.

- **Retorno da Métrica:** Define o tipo de retorno que pode ser obtido a partir de uma métrica.
- **Referência:** É a forma de referenciar a métrica ao artigo no qual ela foi extraída.

A seção a seguir descreve o procedimento adotado na categorização e apresenta os conjuntos de métricas categorizados.

## 5.5 Categorização das Métricas

Após a extração das métricas, as mesmas foram separadas e agrupadas. Os artigos selecionados apresentavam características de qualidade relacionadas as medidas de qualidade. Todas as características eram relacionadas à segurança e aos seus respectivos domínios. Tendo isso em vista, as métricas foram separadas e categorizadas em grupos que atendessem a essas características.

Em alguns dos artigos selecionados, não são especificadas todas informações referentes a métrica. Dentre elas, a mais ausente foi a forma de cálculo das métricas, sendo apresentadas apenas em uma pequena parte dos artigos selecionados. As seções a seguir, apresentam os conjuntos de métricas separadas por suas devidas categorias.

### 5.5.1 Métricas de desempenho e responsividade

A Tabela 14 apresenta um conjunto de métricas categorizadas como métricas de desempenho e responsividade, que são duas características que possuem definições em relação ao tempo muito semelhantes.

Tabela 14 – Categorização das métricas de Desempenho e Responsividade.

<b>INFORMAÇÕES GERAIS</b>	
<b>Nome da Métrica</b>	<b>Precisão de eventos</b>
<b>Tipo da Métrica</b>	Produto
<b>Descrição da Métrica</b>	É a precisão do tempo em que um ou mais eventos ocorrem.

<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Latência</b>
<b>Tipo da Métrica</b>	Produto
<b>Descrição da Métrica</b>	É o tempo gasto para fornecer um serviço solicitado ou permitir acesso a recursos. Esta medida fornece informações sobre o atraso do sistema.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Tempo de resposta</b>
<b>Tipo da Métrica</b>	Produto
<b>Descrição da Métrica</b>	É o tempo necessário para responder inicialmente a uma solicitação de um serviço ou para acessar recursos.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Agendamento</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	É o grau em que os eventos podem ser agendados e, em seguida, ocorrerem nos horários programados.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Tempo de Serviço</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Mede o tempo dispendido para o sistema entregar um serviço ao ator que o solicita.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Comprimento do caminho de instrução.</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Mede o número de instruções de código de máquina necessárias para executar uma seção de um programa de computador.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)

<b>Nome da Métrica</b>	<b>Tempo de conclusão</b>
<b>Tipo da Métrica</b>	Produto
<b>Descrição da Métrica</b>	Mede a quantidade de tempo necessária para executar e concluir uma determinada tarefa.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)

Fonte: Autor.

### 5.5.2 Métricas de Disponibilidade

A Tabela 15 apresenta o conjunto de métricas categorizadas como métricas de disponibilidade. Essa é uma das características frequentemente citadas nos artigos do mapeamento como de grande importância para sistemas críticos de segurança.

Tabela 15 – Categorização de métricas de Disponibilidade.

<b>INFORMAÇÕES GERAIS</b>	
<b>Nome da Métrica</b>	<b>Diversidade.</b>
<b>Tipo da Métrica</b>	Processo.
<b>Descrição da Métrica</b>	Mede o número de implementações diferentes da mesma especificação, quanto mais diversos for um sistema, mais resiliente a falhas ele é.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Replicação.</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Mede o número de sistemas replicados que estão presentes no sistema. Ele também pode ser aplicado a um componente, a fim de identificar seu nível de replicação.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
	<b>Tempo de atividade</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É a medida do tempo que um sistema está funcionando e disponível, representando o tempo que ele pode trabalhar sem parar e sem manutenção.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
	<b>Tempo de inatividade</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Oposto ao tempo de atividade, representa os períodos de tempo em que o sistema não está disponível ou está on-line.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
	<b>Disponibilidade</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	É o tempo que o sistema não falha e não está sob reparos. Nesse conjunto, normalmente encontra-se métricas relacionadas à sustentabilidade.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)

Fonte: Autor.

### 5.5.3 Métricas de Exatidão

A Tabela 16 apresenta um conjunto de métricas categorizadas como métricas de exatidão que são geralmente relacionadas à conformidade de sua operação com o comportamento que se espera.

Tabela 16 – Categorização de métricas de Exatidão.

<b>INFORMAÇÕES GERAIS</b>	
<b>Nome da Métrica</b>	<b>Sensibilidade</b>
<b>Tipo da Métrica:</b>	Produto.
<b>Descrição da Métrica</b>	Também condiz com taxa positiva verdadeira, mede a proporção de positivos reais corretamente identificados como tal.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Precisão</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É uma característica inerente ao sistema de medição, estatisticamente é definida como a dispersão de dados quantitativos, independentemente de sua precisão.

<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
<b>Nome da Métrica</b>	<b>Atualidade</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Em termos de corretude, atualidade, do inglês <i>currency</i> , é definida como o grau em que os dados permanecem atuais, ou seja, não obsoletos.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
<b>Nome da Métrica</b>	<b>Especificidade</b>
<b>Tipo da Métrica</b>	Produto
<b>Descrição da Métrica</b>	Mede a proporção de negativos que são corretamente identificados como tal.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual
<b>Referência</b>	CAZORLA (2015)

Fonte: Autor.

#### 5.5.4 Métricas de Manutenibilidade

A Tabela 17 apresenta um conjunto de métricas classificadas como métricas de Manutenibilidade, que segundo informações extraídas no mapeamento, é uma característica de qualidade de grande importância no contexto de criticidade de sistemas.

Tabela 17 – Categorização de métricas de Manutenibilidade.

<b>INFORMAÇÕES GERAIS</b>	
<b>Nome da Métrica</b>	<b>Manutenção Planejada</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Refere-se aos eventos de manutenção preventiva programados para verificar a operação correta de um sistema.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Tempo de Reparo</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Tempo de Reparo é o tempo médio necessário para reparar um sistema que falhou.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Custos</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Refere-se ao custo médio de substituição de um determinado componente ou sistema.
<b>Fórmula de Cálculo</b>	Não possui.

<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Tempo de Restauração</b>	
<b>Nome da Métrica</b>	
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É o tempo estimado necessário para que um sistema seja restaurado para sua operação original, no caso de falha.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Falhas ao longo do tempo</b>	
<b>Nome da Métrica</b>	
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Ou taxa de falhas, é a frequência com que um sistema falha.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	CAZORLA (2015)
<b>Quantidade de horas de trabalho de Manutenção</b>	
<b>Nome da Métrica</b>	
<b>Tipo da Métrica</b>	Processo.
<b>Descrição da Métrica</b>	Refere-se à mão de obra necessária para manter o sistema.
<b>Fórmula de Cálculo</b>	Não possui.

<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Taxa de eventos de atualização</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Refere-se à frequência estimada de eventos de atualização necessários em um sistema.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Tamanho do <i>Software</i></b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É o número de linhas de código de origem, do inglês <i>Source lines of code</i> (SLOC), mantidas pelo projeto. Essa métrica pode ser usada pelo gerenciamento de projetos.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Quantidade de horas envolvidas na manutenção de <i>software</i></b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Essa métrica é o número de horas de pessoal gastas por mês pelo pessoal de engenharia e gerenciamento de <i>software</i> diretamente envolvido com manutenção de <i>software</i> . A métrica fornece suporte ao gerenciamento dos dados para prever os futuros requisitos de pessoal.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Processamento de Solicitação de Manutenção</b>
<b>Tipo da Métrica</b>	Projeto.
<b>Descrição da Métrica</b>	A métrica de processamento de solicitação de manutenção monitora o fluxo de trabalho de manutenção de <i>software</i> e o nível de satisfação do cliente.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Agendamento de aprimoramento de <i>software</i></b>
<b>Tipo da Métrica</b>	Processo.
<b>Descrição da Métrica</b>	Rastreia o período de tempo para fechar uma solicitação de aprimoramento e o esforço de engenharia gasto em aprimoramentos. Duas primitivas de dados são usadas para calcular essa métrica: (1) o número planejado e real de horas de engenharia gastas no aprimoramento e; (2) a quantidade de tempo de calendário desde o envio da solicitação até que o aprimoramento seja liberado para o recurso.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.

<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Utilização de recursos de computador</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A métrica CRU ( <i>Computer Resource Utilization</i> ) rastreia o uso real dos recursos do sistema.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Densidade de Falha</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	O número de relatórios de discrepância fechados com uma correção de <i>software</i> por 1000 linhas de código de origem ao longo do tempo define a métrica de densidade de falhas.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Volatilidade de <i>Software</i></b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A métrica de volatilidade de <i>software</i> é uma proporção do número de módulos alterados devido a uma solicitação de manutenção de <i>software</i> para o número total de módulos em um release ao longo do tempo.

<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	
	<b>Duração de abertura do relatório de discrepância</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A métrica de duração de abertura do relatório de discrepância, do inglês <i>Discrepancy Reports</i> (DR), monitora a quantidade de tempo necessária para fechar os DRs de <i>software</i> assim que eles são descobertos.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	
	<b>Taxa de quebra / fixação</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A taxa de quebra / correção é a contagem de falhas inseridas na linha de base do <i>software</i> operacional dividida pelo número total de alterações feitas no <i>software</i> .
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	
	<b>Confiabilidade de Software</b>
<b>Tipo da Métrica</b>	Processo.

<b>Descrição da Métrica</b>	A confiabilidade do <i>software</i> é a probabilidade de que o <i>software</i> não falhe por um período de tempo especificado em um ambiente especificado.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	
	<b>Complexidade de design</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A complexidade do design rastreia o número de módulos com uma medida de complexidade maior do que a diretriz estabelecida.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	
	<b>Distribuição de tipo de falha</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Esta métrica acompanha o fechamento do relatório de discrepância de três maneiras: (1) por código de fechamento (ou seja, hardware, <i>software</i> , humano, incapaz de duplicar, etc.); (2) pelo tipo de problema encontrado (isto é, lógica, computacional, interface, entrada de dados, etc.); e (3) pelo processo que introduziu o problema (ou seja, requisitos, design, código, teste).
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.

<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Número de instâncias em ADA</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É uma contagem do número de unidades genéricas da linguagem de programação Ada projetadas e codificadas durante o curso de manutenção, o tamanho das unidades genéricas (em SLOC) e uma contagem do número de vezes que a unidade genérica é instanciada. O objetivo dessa métrica é rastrear o número e o tamanho dos componentes reutilizáveis desenvolvidos por um projeto para uso dentro do projeto e de outros projetos.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	STARK, KERN, VOWELL (1994).
<b>Nome da Métrica</b>	<b>Implementação do programa de métrica</b>
<b>Tipo da Métrica</b>	Projeto.
<b>Descrição da Métrica</b>	Depois de definir o conjunto de métricas, decide fazer com que os contratados colem, analisem, relatem e usem as métricas.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	STARK, KERN, VOWELL (1994).

Fonte: Autor.

### 5.5.5 Métricas de Segurança e Confiabilidade

A Tabela 18 apresenta um conjunto de métricas categorizadas como métricas de Segurança, que é uma característica genérica que agrupa métricas relacionadas a *Safety* e *Security*, incluindo

a confiabilidade, que são atributos de qualidade diretamente relacionados à criticidade e presença de falhas em um sistema.

Tabela 18 – Categorização de métricas de Confiabilidade e Segurança.

<b>INFORMAÇÕES GERAIS</b>	
<b>Nome da Métrica</b>	<b>Exatidão</b>
<b>Tipo da Métrica</b>	Processo.
<b>Descrição da Métrica</b>	Geralmente relacionada ao processo de desenvolvimento do sistema e à conformidade de sua operação com o comportamento esperado.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Eficácia</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Medir a eficácia é geralmente baseado nos mecanismos de segurança, determinando o quão bem eles funcionam e se o sistema mostra sinais de vulnerabilidades.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	<b>Percentual de Perigos de Segurança do Sistema</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Mede os perigos presentes no sistema.

<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual
<b>Referência</b>	CAZORLA (2015)

<b>Nome da Métrica</b>	<b>Bugs por linha de código (BLOC)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	O objetivo desta métrica é fornecer uma estimativa aproximada do número de falhas em um programa.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).

<b>Nome da Métrica</b>	<b>Gráfico de causa e efeito (CEG)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É uma tradução formal da especificação de linguagem natural em suas condições de entrada e resultados esperados e ajuda a identificar requisitos que são incompletos e ambíguos.
<b>Fórmula de Cálculo</b>	$CE\% = 100x(1-(A_{existente})/(A_{total})):$ <p><math>A_{existente}</math>: é o número de ambiguidades em um programa que resta a ser eliminado;  <math>A_{total}</math>: total de ambiguidades a ser identificadas.</p>
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	SHI (2017).

<b>Nome da Métrica</b>	<b>Modelo de maturidade de capacidade de <i>software</i> (CMM)</b>
------------------------	--

<b>Tipo da Métrica</b>	Processo.
<b>Descrição da Métrica</b>	O objetivo desta medida é descrever os princípios e práticas relacionadas a qualidade do processo de <i>software</i> para ajudar as organizações de <i>software</i> a melhorar a maturidade de seus processos de <i>software</i> .
<b>Fórmula de Cálculo</b>	$CMM = i \in \{1, 2, 3, 4, 5\}$
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	FENTON (1997).
<b>Nome da Métrica</b>	<b>Integridade</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Métrica que determina a integridade das especificações de requisitos de <i>software</i> .
<b>Fórmula de Cálculo</b>	$COM = \sum_{i=1}^{10} w_i D_i:$ <p>COM: é a medida de perfeição;  <math>W_i</math>: é o peso da <math>i^{\text{th}}</math> medida derivada;  <math>D_i</math>: é a <math>i^{\text{th}}</math> medida derivada calculada a partir das medidas primitivas <math>B_i</math> (<math>i = 1, \dots, 18</math>).</p>
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Complexidade ciclomática (CC)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Essa medida determina a complexidade estrutural de um módulo codificado e o número de caminhos linearmente independentes através de um módulo.
<b>Fórmula de Cálculo</b>	$CC_i = E_i - N_i + 1:$ <p><math>CC_i</math>: é a medida de complexidade ciclomática do módulo <math>i^{\text{th}}</math>;  <math>E_i</math>: é o número de arestas do módulo <math>i^{\text{th}}</math> (fluxos de programa entre nós);</p>

	$N_i$ : é o número de nós do módulo $i^{\text{th}}$ (grupos sequenciais de instruções de programa).
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Fator de Cobertura (CF)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É explicada como a probabilidade de o sistema se recuperar de uma falha que uma falha ocorre.
<b>Fórmula de Cálculo</b>	$c = Pr(H(g) = 1   g \in G):$ Pr: é a probabilidade de $H(g) = 1$ quando $g \in G$ ; H: é uma variável que caracteriza o manuseio de uma determinada condição defeituosa; $H(g) = (1; \text{ se o mecanismo manuseia corretamente o condicionamento defeituoso}; 0; \text{ de outra forma};$ G é o espaço de entrada completo de um mecanismo de tolerância a falhas; g é uma condição defeituosa ou um ponto no espaço G.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Densidade de defeitos (DD)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É definida como o número de defeitos remanescentes dividido pelo número de linhas de código no <i>software</i> .
<b>Fórmula de Cálculo</b>	$DD = \sum_{i=1}^I D_i / KSLOC;$ $D_i$ : é o número de defeitos únicos detectados durante o projeto ou inspeção de código que ainda permanecem no código.

	KSLOC: é o número de linhas de código de origem (LOC) em milhares.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Número de dias de falha (FD)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Essa medida representa o número de dias que as falhas permanecem no sistema de <i>software</i> de introdução à remoção.
<b>Fórmula de Cálculo</b>	$FD_i = f_{out_i} - fin_i \text{ e } FD = \sum_{i=1}^I FD_i;$ <p>FD: são os dias de falha para o sistema total;            FD<sub>i</sub>: são os dias de falha para a i<sup>th</sup> falha;            fin: é a data em que a falha supra foi introduzida no sistema            fout<sub>i</sub>: é a data em que a i<sup>th</sup> falha foi removida do sistema.</p>
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Análise de pontos de função (FP)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É uma medida projetada para determinar o tamanho funcional do <i>software</i> .
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).

<b>Nome da Métrica</b>	<b>Requisição de mudança de especificação de requisito (RSCR)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É definida como o número de solicitações de mudança que são feitas para especificação dos requisitos.
<b>Fórmula de Cálculo</b>	$RSCR = \sum$ (solicitações de alterações na especificação de requisitos)
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Requisitos de rastreabilidade (RT)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	A medida de rastreabilidade de requisitos ajuda a identificar requisitos que estão faltando ou além dos requisitos originais.
<b>Fórmula de Cálculo</b>	$RT = R_1/R_2 \times 100\%$ ; RT: é o valor da rastreabilidade de requisitos de medida; R <sub>1</sub> : é o número de requisitos atendidos pela arquitetura; R <sub>2</sub> : é o número de requisitos originais.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	<b>Complexidade de Design de Sistemas (SDC)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	É uma função da complexidade média dos vários módulos onde a complexidade do design do módulo é uma combinação de complexidade estrutural.
<b>Fórmula de Cálculo</b>	$SDC = 1/n \sum_{i=1}^n f^2(i) + 1/n \sum_{i=1}^n \vartheta(i) / f(i) + 1$ ; n: é o número total de módulos no sistema; $f_{(i)}$ : é o espalhamento do i <sup>th</sup> módulo;

	$\vartheta_{(i)}$ : é o número de variáveis de E / S no módulo $i^{\text{th}}$ .
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SHI (2017).
<b>Nome da Métrica</b>	
	<b>Precisão da Qualidade</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Calcula a qualidade de determinado documento a partir do modelo fuzzy, que destaca quatro características para a garantia da qualidade.
<b>Fórmula de Cálculo</b>	$Quality = Q_{doc}(n) = \sum PQ + CQ + TQ + LQ / n;$ <p>Qualidade de Precisão, <math>PQ = \{pq1, pq2, \dots, pqn\}</math>  Qualidade de Correção, <math>CQ = \{cq1, cq2, \dots, cqn\}</math>  Qualidade de Consistência, <math>TQ = \{tq1, tq2, \dots, tqn\}</math>  Qualidade de Completude, <math>LQ = \{lq1, lq2, \dots, lqn\}</math></p>
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	MIT e SHIANG (2015)
<b>Nome da Métrica</b>	
	<b>Métrica de comprimento de chave</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	O sistema criptográfico é uma função do comprimento da chave. No entanto, adicionar um bit extra nem sempre é exatamente dobrar o esforço necessário para quebrar a criptografia.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Percentual.
<b>Referência</b>	SAVOLA (2009).
<b>Nome da Métrica</b>	
	<b>Métrica de tempo de ataque</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	O tempo de ataque é definido como o tempo necessário para realizar a invasão do sistema mais rápido conhecido.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SAVOLA (2009).
<b>Nome da Métrica</b>	
	<b>Métrica de etapas de ataque</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	As etapas de ataque são definidas como número de etapas necessárias para executar os melhores ataques.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SAVOLA (2009).
<b>Nome da Métrica</b>	
	<b>Tempo Médio para Falha Insegura (MTTUF)</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Representa o tempo médio que um sistema irá operar com segurança antes da ocorrência de uma falha que produza um estado inseguro do sistema.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	CAZORLA (2015)
<b>Nome da Métrica</b>	
	<b>Eta1</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Número de operadores distintos presentes na função eta usada na matemática na área de teoria analítica dos números.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b><math>\eta_2</math></b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Número de operandos distintos presentes na função eta usada na matemática na área de teoria analítica dos números.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b><math>\eta_1</math></b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Número total de operadores presentes na função eta usada na matemática na área de teoria analítica dos números.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b><math>\eta_2</math></b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Número total de operandos presentes na função eta usada na matemática na área de teoria analítica dos números.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
<b>Instruções</b>	
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Número total de instruções, do inglês <i>statements</i> , (código executável; sem comentários).
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
<b>LOC</b>	
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Total de linhas de código, do inglês <i>lines of code</i> (LOC), não comentadas.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
<b>Comentários</b>	
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Número total de comentários no código.
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b>Nós</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Número total de nós (no gráfico de controle).
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b>Arestas</b>
<b>Tipo da Métrica</b>	Produto.
<b>Descrição da Métrica</b>	Número total de arestas (no gráfico de controle).
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).
<b>Nome da Métrica</b>	
	<b>Caminhos</b>
<b>Tipo da Métrica</b>	Produto.

<b>Descrição da Métrica</b>	Número total de caminhos (no gráfico de controle).
<b>Fórmula de Cálculo</b>	Não possui.
<b>Unidade de Métrica</b>	Valor numérico.
<b>Referência</b>	SCHNEIDEWIND (1997).

Fonte: Autor.

Este capítulo apresentou os resultados obtidos com o mapeamento sistemático realizada para responder a seguinte pergunta de pesquisa:

*Quais são as métricas de qualidade usadas no desenvolvimento de sistemas críticos?*

Na pesquisa inicial foram retornados 805 estudos que após a aplicação dos critérios de exclusão resultaram em sete artigos. A partir dos estudos selecionados foram extraídas 67 métricas que foram categorizadas e agrupadas.

No próximo capítulo são apresentadas as conclusões e discutidos trabalhos futuros.

## 6 CONCLUSÃO

A garantia da qualidade para sistemas de alta criticidade é sempre um problema a ser avaliado, pois é algo essencial para a tolerância a falhas. A qualidade de *software* é a área da Engenharia de *software* que tem diversas atribuições para tratar a garantia da qualidade de um produto. Entre essas atribuições da qualidade, há o uso de métricas para que através de fórmulas, alguns objetivos possam ser medidos e avaliados de forma concreta.

Tendo em vista a grande dificuldade que há na garantia da qualidade de um sistema crítico, o trabalho tem como objetivo trazer um conjunto de métricas categorizadas por características de qualidade que são relacionadas ao domínio de sistemas críticos. As métricas e as informações apresentadas foram todas baseadas em estudos de mapeamento sistemático da literatura. Com as informações extraídas foi possível responder às questões de pesquisa elaboradas no protocolo do mapeamento.

Por meio da aplicação da *string* de busca definida em cinco bases de dados (ACM, IEEE, Science, Scopus e Springer), foram retornados 805 estudos. Após a aplicação dos critérios de exclusão foram obtidos sete artigos. A partir dos estudos selecionados foram extraídas 67 métricas que foram categorizadas e agrupadas.

As métricas apresentadas neste trabalho foram categorizadas considerando as características de qualidade. Essa decisão foi motivada em virtude da presença desse tipo de relacionamento entre as métricas e suas respectivas características ou atributos de qualidade, como também relacionavam essas tais categorias com a contribuição para a tomada de decisão para o desenvolvimento de um sistema tolerante a falhas.

O trabalho resultou em 5 (cinco) categorias contendo um conjunto de métricas relacionadas a sistemas críticos. A primeira delas é a categoria de desempenho e responsividade, que resultou em 7 (sete) métricas, sendo que nenhuma delas apresenta a forma de cálculo. A segunda categoria foi a de disponibilidade, onde resultou em 5 (cinco) métricas, sendo que nenhuma apresentam forma de cálculo. A terceira categoria é a de exatidão, onde apresenta 4 (quatro) métricas também sem a formula de cálculo. A quarta categoria é a de manutenibilidade, que foi a segunda categoria que mais apresentou métricas, totalizando 21 (vinte e uma) métricas também sem apresentar forma de cálculo. A quinta categoria, que foi a que apresentou o maior número de métricas, foi a categoria de segurança e confiabilidade, onde

retornou 30 (trinta) métricas relacionadas a sistemas críticos, pouco menos de 50% do total das métricas, sendo que 11 delas apresentaram forma de cálculo.

A maioria dos artigos não especifica o domínio no qual as métricas poderiam ser aplicadas. Dessa forma, a mesma métrica pode ser utilizada no desenvolvimento de um carro como de um avião por exemplo. Já em outros casos, os artigos apresentavam métricas para certo tipo de domínio, que permitiu a identificação dos tipos de sistemas críticos no qual a mesma poderia ser aplicada, sendo que maioria das pesquisas utilizavam tais métricas para a tomada de decisão na construção de sistemas críticos de missão.

Foram poucos os artigos que passaram para a etapa da extração das métricas, e nem todos apresentavam o tipo de domínio dos quais as métricas propostas ou apresentadas eram aplicadas, mas tiveram a exceção de três artigos, onde foi proposta a aplicação das métricas. O trabalho de CAZORLA (2015), apresentou as métricas de segurança, confiabilidade, manutenibilidade e exatidão, todas propostas para o domínio de Intrusão em rede. MIT e SHIANG (2015) apresentou métricas de segurança aplicada ao domínio de requisitos de sistemas. Por fim, SHI (2017) apresentou métricas de segurança voltadas para o domínio aéreo e espacial.

Uma métrica permite mensurar algo para auxiliar na qualidade do sistema. A maioria dos estudos selecionados no mapeamento apresentavam apenas uma descrição trazendo o objetivo de tal métrica, mas poucos foram os que apresentam a fórmula de como elas são mensuradas, apenas 16,4% delas apresentaram forma de cálculo. Em alguns casos onde as métricas apresentavam a fórmula de cálculo, apareciam elementos da fórmula em formato de siglas que nem sempre eram descritos no artigo, dificultando assim, sua adoção. As demais foram apresentadas de forma totalmente clara ao ponto de vista do leitor, assim, contribuindo de forma eficaz para o estudo realizado.

Após todo o processo realizado nas etapas anteriores, já será possível ter informações relevantes para o processo de construção de um sistema crítico. Porém, para saber se essas informações são realmente úteis, e obter os resultados finais dessa pesquisa, é necessário que os resultados gerados e categorizados sejam validados. Portanto, a proposta da validação do conjunto de métricas é o trabalho que será realizado futuramente.

Como a ideia é ter um grande volume de dados para se ter mais informações relevantes para a pesquisa, um trabalho futuro será a criação de um catálogo de métricas através de um

mapeamento sistemático sem a exclusão dos estudos secundários, para a obtenção de mais métricas, e logo após disponibilizar esses dados em um portal, assim tendo um maior volume de dados e uma maior contribuição para o estudo.

## REFERÊNCIAS

- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR ISO 9000:2005**. Rio de Janeiro, 2005.
- AMORIM, B. P., SANTOS, D. R., MUNIZ, D. A., COSTA, J. P. D. M., ANTUNES, L. A., CRISTINA, E., ABE, N. Gestão de qualidade no processo de desenvolvimento de software. In: **Revista Gestão em Foco**, São Paulo, 2017.
- ARKSEY, Hilary; O'MALLEY, Lisa. Scoping studies: towards a methodological framework. **International journal of social research methodology**, v. 8, n. 1, p. 19-32, 2005.
- BEZERRA, Carla Ilane Moreira. **Medidas para avaliação da manutenibilidade do modelo de features de linhas de produto de software tradicionais e dinâmicas**. 2016. 202 fl. Tese (Doutorado em Ciência da Computação) - Universidade Federal do Ceará, Fortaleza, 2016.
- CAZORLA, Lorena; ALCARAZ, Cristina; LOPEZ, Javier. A three-stage analysis of IDS for critical infrastructures. **Computers & Security**, v. 55, p. 235-250, 2015.
- CRUICKSHANK, Kristian J.; MICHAEL, James Bret; SHING, Man-Tak. A Validation Metrics Framework for safety-critical software-intensive Systems. In: System of Systems Engineering, 2009. SoSE 2009. **IEEE International Conference on**. IEEE, 2009. p. 1-8.
- DELONG, Todd A.; SMITH, D. Todd; JOHNSON, Barry W. Dependability metrics to assess safety-critical systems. **IEEE Transactions on Reliability**, v. 54, n. 3, p. 498-505, 2005.
- DE AG SARAIVA, Juliana et al. Classifying metrics for assessing object-oriented software maintainability: A family of metrics' catalogs. **Journal of Systems and Software**, v. 103, p. 85-101, 2015.
- DUARTE, Katia Cristina; FALBO, Ricardo A. Uma ontologia de qualidade de software. In: **Workshop de Qualidade de Software, João Pessoa**. 2000. p. 275-285.
- FENTON, Norman. The role of measurement in software safety assessment. In: **Safety and reliability of software based systems**. Springer, London, 1997. p. 217-248.
- GORSCHKEK, Tony et al. A model for technology transfer in practice. **IEEE software**, v. 23, n. 6, p. 88-95, 2006.
- KARANIKAS, Nektarios. Critical review of safety performance metrics. **International Journal of Business Performance Management**, v. 17, n. 3, p. 266-285, 2016.
- KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. [S.l.: s.n], 2007.

KNEGTERING, B.; PASMÁN, H. J. Safety of the process industries in the 21st century: a changing need of process safety management for a changing industry. **Journal of Loss Prevention in the Process Industries**, v. 22, n. 2, p. 162-168, 2009.

KITCHENHAM, B. A., DYBA, T., e JORGENSEN, M. Evidence-based software engineering. In ICSE '04, pages 273–281, Washington, DC, USA. **IEEE Computer Society**, 2004.

KITCHENHAM, B.A., CHARTERS, S. **Guidelines for performing systematic literature reviews in software engineering**. Tech. Rep. EBSE-2007-01, Keele University, 2007.

LEVESON, N. **Safeware: system safety and computers**. [S.l]: ACM, 1995.

LEVESON, N. **Engineering a safer world: systems thinking applied to safety**. [S.l]: Mit Press, 2011.

LUTZ, R. Software engineering for safety: a roadmap. In: **Proceedings of the Conference on The Future of Software Engineering**. ACM, 2000, p. 213–226.

MENDELOFF, John et al. Evaluation of process safety indicators collected in conformance with ANSI/API Recommended Practice 754. **Journal of Loss Prevention in the Process Industries**, v. 26, n. 6, p. 1008-1014, 2013.

MERRIAM, S. B.; TISDELL, E. J. **Qualitative research: a guide to design and implementation**. John Wiley & Sons, 2015

MIT, Edwin; SHIANG, Cheah Wai. Formalize the software quality measurement for heterogeneous requirements. In: **IT in Asia (CITA), 2015 9th International Conference on**. IEEE, 2015. p. 1-4.

MUNSON, JC. **Software Engineering Measurement**. Auerbach Publications, London, New York, 2003.

NASCIMENTO, Tiago da Silva. **Um catálogo de métricas para avaliação da qualidade de ambientes integrados de computação em nuvem, redes definidas por software e virtualização de funções de rede**. 2017. 69 f. TCC (Graduação em Tecnologia de Redes de Computadores) - Universidade Federal do Ceará, Campus Quixadá, Quixadá, CE, 2017.

OUEDRAOGO, Moussa et al. Taxonomy of quality metrics for assessing assurance of security correctness. **Software Quality Journal**, v. 21, n. 1, p. 67-97, 2013.

PASMÁN, Hans; ROGERS, William. How can we use the information provided by process safety performance indicators? Possibilities and limitations. **Journal of Loss Prevention in the Process Industries**, v. 30, p. 197-206, 2014.

ROCHA, ARC da et al. Uma experiência na Definição do Processo de Desenvolvimento e Avaliação de Software segundo as normas ISO. **Relatório Técnico ES-302/94, COPPE/UFRJ**, 1994.

SAMPAIO, R. F. Estudos de revisão sistemática: um guia para síntese criteriosa da evidência científica. **Revista brasileira de fisioterapia**, v. 11, n. 1, p. 83-89, 2007.

SEAMAN, C. B. Qualitative methods in empirical studies of software engineering. **IEEE Transactions on software engineering**, IEEE, v. 25, n. 4, p. 557–572, 1999.

SIMPSON, A.; STOKER, J. Will it be safe? An approach to engineering safety requirements. In: **Components of System Safety**. Springer, 2002, p. 140–164.

SCHNEIDEWIND, Norman F. Software metrics model for quality control. In: **Software Metrics Symposium, 1997. Proceedings., Fourth International**. IEEE, 1997. p. 127-136.

SHI, Ying et al. Metric-based software reliability prediction approach and its application. **Empirical Software Engineering**, v. 22, n. 4, p. 1579-1633, 2017.

STARK, George E.; KERN, Louise C.; VOWELL, C. W. A software metric set for program maintenance management. **Journal of Systems and Software**, v. 24, n. 3, p. 239-249, 1994.

SWUSTE, Paul et al. Process safety indicators, a review of literature. **Journal of Loss Prevention in the Process Industries**, v. 40, p. 162-173, 2016.

VARKOI, Timo. Safety as a process quality characteristic. In: **International Conference on Software Process Improvement and Capability Determination**. Springer, Berlin, Heidelberg, 2013. p. 1-12.

VILELA, J.; CASTRO, J.; MARTINS, L. E. G.; GORSCHKEK, T. Integration between requirements engineering and safety analysis: A systematic literature review. **Journal of Systems and Software**, Elsevier, v. 125, p. 68–92, 2017.

WIERINGA, R. Relevance and problem choice in design science. In: **International Conference on Design Science Research in Information Systems**, 2010.p. 61–76.

WOHLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M. C.; REGNELL, B.; WESSLÉN, A. **Experimentation in software engineering: an introduction**. Norwell, MA, USA: Kluwer Academic Publishers, 2000. ISBN 0-7923-8682-5.

WOHLIN, C., RUNESON, P., SILVEIRA NETO, P.A.M., ENGSTRÖM, E., MACHADO, I.C., ALMEIDA, E.A., On the Reliability of Mapping Studies in Software Engineering. **Journal of Systems and Software**, vol. 86, n. 10, 2013, p. 2594 - 2610.