

UNIVERSIDADE FEDERAL FLUMINENSE
INSTITUTO DE COMPUTAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

RODRIGO NUNES LAIGNER

**DESENVOLVIMENTO DE SISTEMAS BIG DATA: UM MAPEAMENTO
SISTEMÁTICO DA LITERATURA**

Niterói
2017

RODRIGO NUNES LAIGNER

**DESENVOLVIMENTO DE SISTEMAS BIG DATA: UM MAPEAMENTO
SISTEMÁTICO DA LITERATURA**

Trabalho de conclusão de curso
apresentado ao curso de Bacharelado em
Sistemas de Informação, como requisito
parcial para conclusão do curso.

Orientador:
Prof. Dr. Rodrigo Salvador Monteiro.

Coorientador:
Prof. Dr. Marcos Kalinowski.

Niterói
2017

Ficha Catalográfica elaborada pela Biblioteca da Escola de Engenharia e Instituto de Computação da UFF

L185 Laigner, Rodrigo Nunes
Desenvolvimento de sistemas *big data* : um mapeamento
sistemático da literatura / Rodrigo Nunes Laigner. – Niterói, RJ :
[s.n.], 2017.
84 f.

Projeto Final (Bacharelado em Sistemas de Informação) –
Universidade Federal Fluminense, 2017.
Orientadores: Rodrigo Salvador Monteiro, Marcos Kalinowski.

1. Engenharia de software. 2. Mineração de texto. 3. Mineração
de dados (Computador). I. Título.

CDD 005.1

RODRIGO NUNES LAIGNER

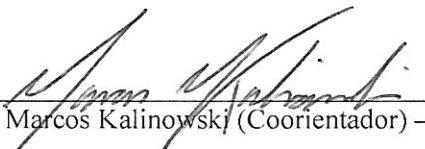
**DESENVOLVIMENTO DE SISTEMAS BIG DATA: UM MAPEAMENTO
SISTEMÁTICO DA LITERATURA**

Trabalho de conclusão de curso
apresentado ao curso de Bacharelado em
Sistemas de Informações, como requisito
parcial para conclusão do curso.

Aprovada em 15 de dezembro de 2017.

BANCA EXAMINADORA


Prof. Dr. Rodrigo Salvador Monteiro (Orientador) - UFF


Prof. Dr. Marcos Kalinowski (Coorientador) – PUC-Rio


Prof. Dr. Daniel de Oliveira - UFF

AGRADECIMENTOS

À Deus, por me sustentar ao longo de toda essa jornada, por estar comigo nos momentos de tristeza, dor e sofrimento.

À minha família, em especial minha mãe, minha dindinha, meu tio Fábio e minha irmã, por todo o apoio durante este período de estudos, estágios, cursos e viagens. Sem a base que me proporcionaram, jamais estaria escrevendo estas linhas.

Ao meu amigo Julio, por me apoiar em momentos difíceis nos últimos anos.

Aos meus amigos, Erick, Leonardo, Lucas e Paulo pelas caronas e momentos muito felizes que passamos ao longo da graduação.

Ao professor Marcos Kalinowski, por ter me aberto as portas para a pesquisa desde o primeiro contato, me orientando de forma paciente.

Ao professor Rodrigo Salvador, que assumiu minha orientação e esteve aberto a contribuir para este trabalho.

À Visagio, que, quando necessário, me deixou confortável para deixar de exercer minhas funções na organização, dessa forma, pude me esforçar para concluir esse trabalho.

RESUMO

De acordo com um estudo realizado em 2015, a atual quantidade de dados gerados nas organizações levou a um maior investimento em desenvolvimento de infraestrutura e data *analytics*. Entretanto, a dimensão do desenvolvimento de aplicações de software é subestimada. Com o objetivo de habilitar o desenvolvimento de aplicações para usuários-finais utilizando big data, o campo da engenharia de software apresenta um conjunto sólido de diretrizes para avaliar diferentes domínios de aplicação, processos de desenvolvimento e engenharia de requisitos. É fundamental a investigação de que esforços tem sido empregados no desenvolvimento de sistemas big data com o objetivo de prover a pesquisadores e profissionais informações que habilitem maiores atividades de pesquisa. Esse estudo objetiva inspecionar a pesquisa existente na área da engenharia de software para big data com o objetivo de identificar abordagens empregadas, estratégias de desenvolvimento e identificar as pesquisas atuais. Um mapeamento sistemático foi realizado baseado em um conjunto de 8 questões de pesquisa. No total, 305 estudos, datados de 2011 a 2016, foram avaliados. Nós propusemos um novo protocolo e recuperamos um conjunto de estudos primários, identificamos uma lista de abordagens e analisamos o atual estado na construção de sistemas de software big data, identificando tendências e lacunas onde novos esforços de estudo podem ser investidos. Os resultados desse mapeamento sistemático podem suportar pesquisadores e profissionais em escolhas de desenvolvimento e pesquisa futura.

Palavras-chave: Sistemas com uso intensivo de dados. Engenharia de software big data. Mapeamento sistemático.

ABSTRACT

According to study made in 2015, the current amount of data generated in organizations have led to an increased investment in infrastructure development and data analytics. However, the applications software development side is underestimated. In order to enable the development of end-user applications utilizing big data, software engineering field presents a solid set of directives to assess different application domains, development processes and requirements engineering. It is fundamental to investigate what efforts in big data systems development have been employed in order to provide both researchers and practitioners with information that enable further research activities. This study aims at surveying existing research on big data software engineering in order to identify approaches employed, development strategies, and identifying current research. A systematic mapping study was performed based on a set of 8 research questions. In total, 305 studies, dated from 2011 to 2016, were evaluated. We proposed a novel protocol and retrieved a set of primary studies, identified a list of approaches, and analyzed the current state on building big data software systems, identifying trends and gaps where new research efforts can be invested. The results of this systematic mapping can support researchers and practitioners in development choices and future research.

Keywords: Data intensive systems. Big data software engineering. Systematic mapping.

LISTA DE ILUSTRAÇÕES

Figura 1 –	Processo de mapeamento sistemático.....	15
Figura 2 –	Etapas do processo de mapeamento sistemático empregado neste trabalho.....	16
Figura 3 –	Processo de seleção de estudos	38
Figura 4 –	Número de estudos por abordagem	51
Figura 5 –	Número de estudos por objetivo de estudo	54
Figura 6 –	Distribuição de estudos primários por estudo empírico	58
Figura 7 –	Distribuição de número de estudo empírico por abordagem	58
Figura 8 –	Número de estudos primários por tipo de pesquisa	59
Figura 9 –	Porcentagem de estudos por tipo de pesquisa ano a ano.....	60
Figura 10 –	Distribuição de estudos primários com estudo empírico por domínio de aplicação	61
Figura 11 –	Distribuição de estudos primários por aplicabilidade em fase do ciclo de vida	63
Figura 12 –	Número de estudos por tipo de contribuição	64
Figura 13 –	Porcentagem de estudos por tipo de contribuição ano a ano	64
Figura 14 –	Número de estudos por tipo de autor	65
Figura 15 –	Número de publicações por foro	66
Figura 16 –	Visualização do mapeamento sistemático em forma de gráfico em bolhas	68

LISTA DE TABELAS

Tabela 1 –	Trabalhos relacionados	22
Tabela 2 –	Artigos de controle selecionados	29
Tabela 3 –	String de busca	30
Tabela 4 –	String de busca genérica	31
Tabela 5 –	String de busca para cada biblioteca digital	32
Tabela 6 –	Utilização de critério de seleção na seleção dos estudos	34
Tabela 7 –	Conjunto inicial de artigos selecionados	35
Tabela 8 –	Estudos primários recuperados por tipo de <i>snowballing</i>	37
Tabela 9 –	Número de estudos primários selecionados por biblioteca digital	38
Tabela 10 –	Dados de itens extraídos de cada estudo primário	40
Tabela 11 –	Tipos de contribuição de pesquisa	46
Tabela 12 –	Número de estudos por abordagem ano a ano	54
Tabela 13 –	Número de estudos por objetivo ano a ano	55
Tabela 14 –	Distribuição de abordagens por objetivo	57
Tabela 15 –	Divisão de domínios de aplicação por abordagem	63
Tabela 16 –	Número de estudos por tipo de publicação ano a ano	68
Tabela 17 –	Número de estudos por foro de publicação	80

LISTA DE ABREVIATURAS E SIGLAS

BDSE	Big Data Software Engineering
MP	Mapeamento sistemático
SGBD	Sistema de Gerenciamento de Banco de Dados

SUMÁRIO

1	INTRODUÇÃO	12
1.1	MOTIVAÇÃO	12
1.2	OBJETIVO	13
1.3	METODOLOGIA DE PESQUISA	15
1.4	ORGANIZAÇÃO DO TRABALHO	16
2	DESENVOLVIMENTO DE SISTEMAS BIG DATA	18
2.1	INTRODUÇÃO	18
2.2	SISTEMAS BIG DATA	19
2.2.1	TERMINOLOGIA	19
2.2.2	CARACTERÍSTICAS	20
2.3	BIG DATA E ENGENHARIA DE SOFTWARE	21
2.4	BIG DATA E CONTEXTUALIZAÇÃO NESTE TRABALHO	21
2.5	TRABALHOS RELACIONADOS	22
2.6	CONSIDERAÇÕES FINAIS	25
3	MAPEAMENTO SISTEMÁTICO SOBRE DESENVOLVIMENTO DE SISTEMAS BIG DATA	26
3.1	INTRODUÇÃO	26
3.2	PROTOCOLO DE MAPEAMENTO	26
3.2.1	DEFINIÇÃO DE QUESTÕES DE PESQUISA	26
3.2.2	ESTRATÉGIA DE BUSCA	30
3.3	ESCOPO DA BUSCA	32
3.3.1	PERÍODO DE TEMPO	32
3.3.2	BIBLIOTECAS ELETRÔNICAS	32
3.4	SELEÇÃO DOS ESTUDOS	33
3.4.1	DEFINIÇÃO DOS CRITÉRIOS DE INCLUSÃO E EXCLUSÃO	33
3.4.2	PROCESSO DE SELEÇÃO	34
3.5	ESQUEMA DE CLASSIFICAÇÃO	39
3.6	CONSIDERAÇÕES FINAIS	48
4	RESULTADOS OBTIDOS	50
4.1	INTRODUÇÃO	50
4.2	RESPOSTAS ÀS QUESTÕES DE PESQUISA	50
4.3	SÍNTESE DOS RESULTADOS	69
4.4	CONSIDERAÇÕES FINAIS	70
5	CONCLUSÃO	71
5.1	CONTRIBUIÇÕES	71
5.2	LIMITAÇÕES E AMEAÇAS à VALIDADE	72
5.3	TRABALHOS FUTUROS	72
	APÊNDICE A– ESTUDOS PRIMÁRIOS SELECIONADOS.....	79
	APÊNDICE B – TABELA COM NÚMERO DE ESTUDOS POR FORO DE PUBLICAÇÃO	83
	APÊNDICE C – LISTA DE REQUISITOS DE FERRAMENTA PARA GERENCIAMENTO DE MAPEAMENTOS SISTEMÁTICOS	83
	APÊNDICE D – ENDEREÇO PARA ACESSO A PLANILHA DE CONTROLE UTILIZADA NO MAPEAMENTO	84

CAPÍTULO 1 – INTRODUÇÃO

1.1 MOTIVAÇÃO

No início dos anos 2000, com o efeito do e-commerce, na época um modelo inovador de transações e trocas comerciais realizadas pela World Wide Web, desafios na gerência de dados foram identificados pelas organizações. Tipicamente as organizações lidavam com o aumento no volume de dados com a aquisição de mais espaço de armazenamento. Esse processo se mostrava ineficiente no momento em que empresas empreendiam esforços em consolidar bases de diferentes sistemas, causando muitas vezes o problema de aumento de bases de dados redundantes ao longo da operação.

Doug Laney (2001), identificando e buscando solucionar estas lacunas, dissertou sobre iniciativas adequadas para desafios na gerência de dados de grandes corporações. Assim, ele introduziu o conceito dos “3Vs”: Volume, Velocidade e Variedade. Segundo Laney (2001), volume concerne a amplitude e profundidade de cada dado disponível em uma transação ou qualquer ponto de interação (também conhecido como ponto de venda); Velocidade é identificado como o ritmo e velocidade empregados na utilização de dados para suportar interações e na geração de dados pelas interações; por último, Variedade se baseia no grau de compatibilidade existente entre variedades de formatos de dados, fator importante para uma gerência efetiva de dados.

O conceito “3Vs” introduzido por Laney (2001) foi o predecessor do que hoje definimos como big data. De acordo com a TechAmerica Foundation (2012), o termo big data descreve grande volume de dados, que, dada a sua característica complexa e variável, requer técnicas avançadas e tecnologias que habilitem a captura, armazenamento e análise da informação. Em adição, a conferência BDSE (2016) coloca que big data é sobre o processo de extração de informações de alto valor a fim de utilizá-los de maneira inteligente para revolucionar a tomada de decisão em negócios, ciência e sociedade.

O termo big data tem ganhado interesse particular desde 2011, quando a IBM se colocou como um fornecedor de soluções big data no mercado, iniciando assim um interesse contínuo e investimento no campo por parte de institutos de pesquisa e indústria. Desde então, pesquisadores e organizações, como a SAS (2017), tem introduzido outras dimensões aos três originais propostos por Laney (2001), visando abarcar necessidades identificadas no desenvolvimento de soluções. Temos como exemplo Valor e Veracidade, mencionados por Kazman (2016). Burbank (2014) define que Valor provém da habilidade de compreender,

gerenciar e integrar dados de diferentes fontes e/ou formatos de dados com o objetivo de obter informações desconhecidas previamente; além disso, ela argumenta que Veracidade é identificado como o nível de exatidão e acurácia da informação.

De acordo com um levantamento realizado em 2016 pela Capgemini em conjunto com a Informatica, executivos chave na operacionalização de projetos big data em suas organizações apontaram que dois importantes desafios em tornar big data um efetivo recurso de negócio são a falta de expertise técnica e a dificuldade na integração de dados. Em outro levantamento realizado pela Capgemini (2015), foi verificado que a adoção de uma abordagem sistemática de implementação é um fator de sucesso nas iniciativas de big data nas organizações. Entretanto, este relatório indica que apenas 70% das organizações possuem um processo bem definido para identificar e selecionar tecnologias apropriadas.

No contexto do desenvolvimento de sistemas voltados a gerir grande massa de dados, os desafios mencionados pelas pesquisas também estão presentes e são estendidos quando comparados ao desenvolvimento tradicional de sistemas. Pesquisadores expressam argumentos correlatos ao comparar o desenvolvimento tradicional de sistemas com o desenvolvimento de sistemas big data. Por exemplo, segundo Chen *et al.* (2016b), alguns riscos são a dificuldade na seleção de tecnologias big data, a complexa integração de sistemas legados com novos sistemas e, por ser um campo novo, profissionais detém pouco ou nenhum conhecimento. Em outro estudo (Chen *et al.*, 2016a), os mesmos autores discorrem que o desenvolvimento de sistemas que lidam com dados em menor escala é tradicionalmente baseado em bancos de dados relacionais ou *data warehouses* e explicitam a importância de um processo de design de arquitetura. Além disso, Gorton e Klein (2015) argumentam que sistemas big data devem estar aptos a sustentar altas cargas de escrita, variadas cargas e tipos de requisições e alta disponibilidade. Por último, segundo Chua *et al.* (2014), sistemas big data tem como requisitos o armazenamento e gerência de conjuntos de dados massivos e heterogêneos, provendo rápida aquisição e escalabilidade. No que se refere a este trabalho, as necessidades e desafios expostos trazem a oportunidade para a identificação de quais abordagens e estratégias tem sido propostos para a construção de sistemas big data.

1.2 OBJETIVO

Baseado nos desafios expostos e a motivação levantadas na seção anterior, o objetivo deste trabalho é apresentar um mapeamento sistemático da literatura, conduzido para responder à seguinte questão de pesquisa: **“(QP1) Que abordagens da Engenharia de Software tem sido propostos para o suporte e construção de sistemas de software big**

data?”. As seguintes questões de pesquisa complementares foram derivadas da questão principal:

- (QP1a) Quais os objetivos de estudo das abordagens propostas para o suporte e construção de sistemas big data?
- (QP1b) Que avaliações empíricas têm sido utilizadas para a apresentação de propostas de abordagens?
- (QP1c) Quais são os tipos de artigos de pesquisa propondo abordagens da engenharia de software para sistemas big data?
- (QP1d) Para quais domínios de aplicação as abordagens propostas tem sido aplicados (e quantos artigos cobrem os diferentes domínios de aplicação)?
- (QP1e) A quais fases do ciclo de vida do desenvolvimento de software as abordagens propostas se aplicam?
- (QP1f) Que tipos de contribuição de pesquisa tem sido realizadas pelas abordagens propostas?
- (QP1g) Qual o grau de colaboração entre a indústria e a academia na proposta de abordagem de cada estudo?
- (QP1h) Em que foros os estudos propondo abordagens para a construção de sistemas big data estão sendo publicados?

Nesse mapeamento, o foco foi direcionado para abordagens da Engenharia de Software para a construção de sistemas big data. Respondendo essas questões no estudo, identificamos as técnicas e estratégias atualmente empregadas na construção de sistemas com processamento intensivo de dados, os domínios onde tais técnicas têm sido aplicadas, as fases do ciclo de vida priorizadas, metodologias de desenvolvimento utilizadas e boas práticas no gerenciamento de projetos dessa natureza. Estes resultados contribuem para a evolução do corpo de conhecimento da Engenharia de Software no processo de desenvolvimento de sistemas big data.

Acredita-se que os resultados do estudo apresentado neste trabalho serão benéficos tanto para pesquisadores como profissionais da indústria. Para a comunidade de pesquisadores, o mapeamento irá prover informação sobre o atual status das pesquisas em sistemas big data, bem como tópicos de pesquisa que necessitam de maior investigação. Para profissionais, este trabalho apresenta as abordagens atualmente empregadas na construção de sistemas big data, bem como as estratégias para identificação de requisitos e gerência do ciclo de vida de sistemas big data. Profissionais da indústria podem fazer uso da informação

contida neste estudo como base para definir melhores práticas no desenvolvimento de sistemas big data.

1.3 METODOLOGIA DE PESQUISA

Um mapeamento sistemático da literatura é um método útil para reunir informações classificadas em uma revisão de literatura. Permite delinear as características de uma área de estudo em particular por meio de um procedimento sistemático. É definido como um estudo secundário empregado com o objetivo de categorizar as pesquisas existentes e resultados de estudos primários na área, provendo assim um resumo visual das pesquisas relevantes para um dado problema particular (ALVES *et al.*, 2016).

Uma revisão sistemática, apesar de também ser um estudo secundário, emprega uma revisão aprofundada de estudos primários, descrevendo suas metodologias e resultados (PETERSEN *et al.*, 2008). Por outro lado, um mapeamento sistemático (MS) busca mapear os resultados da busca, provendo uma visão geral sumarizada.

Nesse trabalho decidiu-se empregar um mapeamento sistemático, ao invés de uma revisão sistemática, por ser uma área de estudo nova e uma oportunidade para identificar problemas ainda não endereçados na área. Este mapeamento envolve pesquisar a literatura a fim de determinar que tipos de estudos estão sendo empreendidos, quais abordagens têm sido empregadas no desenvolvimento de aplicações big data, seus objetivos, quais domínios de aplicação têm sido executados, as fases do ciclo de vida que se aplicam e que tipo de contribuição proveem. Além disso, é importante também identificar que bases de dados os mesmos se encontram indexados, onde se encontram publicados e que espécie de resultados eles tem alcançado. Para tal, a extração de dados se baseou nas mais importantes bases de dados da área de engenharia de software e sua análise contou com um minucioso processo de revisão dos estudos obtidos.

Um processo padrão para realizar um mapeamento sistemático no contexto da Engenharia de Software foi definido por Petersen *et al.* (2008) e serve de base para o estudo apresentado nesse trabalho. Este é aplicado com o objetivo de identificar lacunas no contexto atual de pesquisas e assim prover insumo para o planejamento de novas. Em adição, um mapeamento pode evitar esforço duplicado, uma vez que pesquisas correlatas podem abordar o mesmo tema.

A Figura 1 exhibe as fases de um mapeamento sistemático para a engenharia de software como proposto por Petersen *et al.* (2008). É importante notar que o trabalho exposto

contou com exemplos de definição de string de busca um mapeamento para a área de linhas de produto de software. Esta área, à época do trabalho de Petersen *et al.* (2008), já provia de muitos estudos primários de qualidade. Dessa forma, etapas importantes do processo, como a definição da string de busca, não necessitaram de uma etapa prévia a fim de obter uma visão geral da área ou identificar os termos e palavras-chave utilizadas no campo. Além disso, as fases propostas por Petersen *et al.* (2008) desconsideraram esforços muito importantes na busca de estudos primários, como o processo de *forward* e *backward snowballing*. Dessa forma, este estudo empreendeu uma adaptação das fases de mapeamento sistemático baseado nas características da área de estudo, isto é, uma área com estudos recentes e ausência de estudos primários relevantes por exemplo. Em outras palavras, novas fases foram introduzidas e outras adaptadas com novos esforços. O procedimento de execução desse mapeamento é exibido na Figura 2. Os detalhes de cada etapa do processo são descritos detalhadamente nas próximas seções.

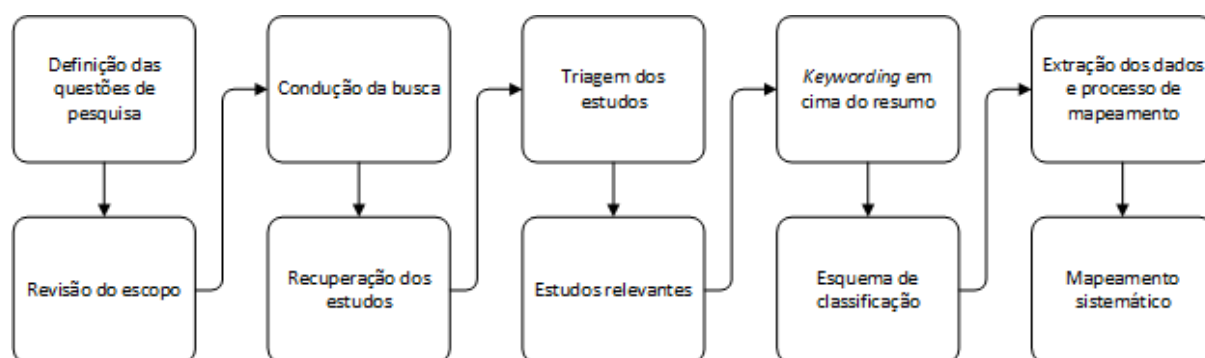


Figura 1 - Processo de mapeamento sistemático (adaptado de PETERSEN *et al.*, 2008)

1.4 ORGANIZAÇÃO DO TRABALHO

Além da introdução, esse trabalho possui outros 4 capítulos. O Capítulo 2 discute a terminologia da área de estudo em sistemas big data, a definição de sistemas big data e alguns trabalhos relacionados, aqueles que empreenderam um estudo correlato a este trabalho, onde se discorre sobre algumas diferenças. No Capítulo 3 a metodologia utilizada neste trabalho é exposta, ou seja, o método empregado para seleção dos estudos primários é explicado. Em seguida, as questões de pesquisa são apresentadas, a estratégia de busca e o processo de seleção de estudos primários. No Capítulo 4 se responde às questões de pesquisa e se discute os resultados, apresentando implicações para profissionais da indústria e pesquisadores. Por último, o Capítulo 5 apresenta as conclusões deste trabalho, ameaças à validação do estudo e trabalhos futuros.

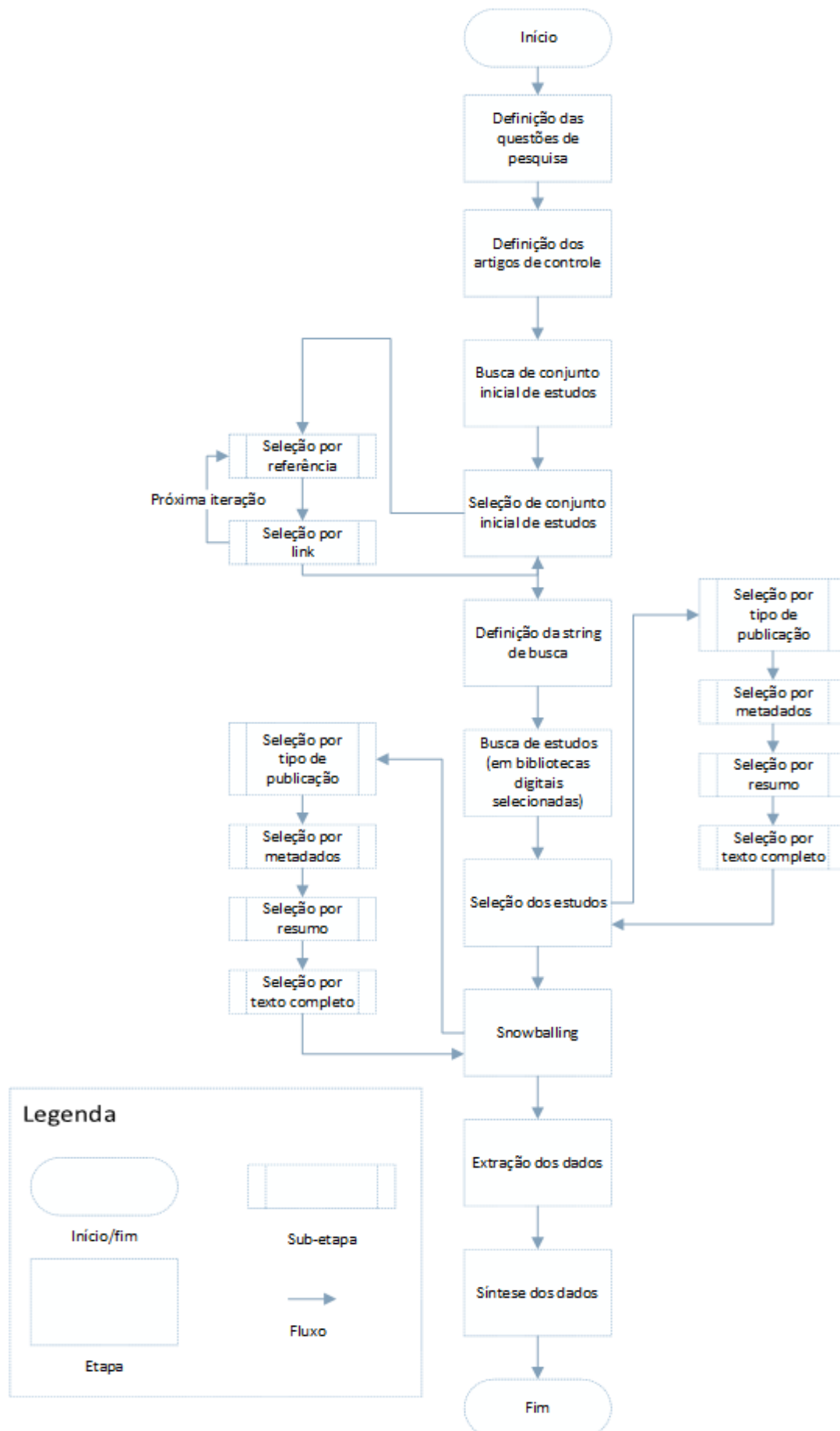


Figura 2 – Etapas do processo de mapeamento sistemático empregado neste trabalho

CAPÍTULO 2 – DESENVOLVIMENTO DE SISTEMAS BIG DATA

2.1 INTRODUÇÃO

De acordo com Carr (2003), infraestruturas de hardware e software tendem a se tornar ativos *commodity* em uma organização. O autor discorre que as organizações que se diferenciam fazem uso da TI como meio de adquirir vantagem competitiva frente a concorrentes, alinhando a estratégia de TI da organização aos objetivos de negócio. Ou seja, a TI deve agregar valor ao negócio, pois, quando relegada a suportar as operações, não incorre em vantagem competitiva. Nesse contexto, em termos comparativos, podemos colocar o desenvolvimento de sistemas de software tradicionais, aqueles onde não há uma complexidade em termos de escalabilidade, processamento intenso de dados, dados semi e não estruturados, como um processo comum na engenharia de software. Nem sempre o desenvolvimento de sistemas tradicionais foi visto como um *commodity*. Há alguns anos atrás, era possível identificar na indústria uma escassez de profissionais com competências necessárias para codificação de software. Entretanto, ao longo dos anos, é possível que a ascensão de frameworks, como Spring e Vaadin, ajudou a mudar esse panorama. Frameworks fornecem a possibilidade de rápida prototipação, retirando do desenvolvedor grande parte da responsabilidade de desenvolvimento de funções essenciais em um software, como autenticação, persistência de dados, log de erros e design da aplicação, promovem alta dependência do desenvolvedor, mas o permitem focar no núcleo da aplicação, suas regras de negócio. Entretanto, atualmente organizações tem se voltado, quando identificam necessidades organizacionais específicas como gerência da cadeia de suprimentos, para softwares de prateleira (por exemplo, Oracle Hyperion e SAP), soluções aquelas que são posicionadas no mercado como produtos. Isto ocorre por maior confiabilidade no suporte e menor probabilidade de erros, dado que já testado e implementado em diversos clientes, e menor custo em relação à aquisição de um software customizado, aqueles modelados para as características e necessidades específicas da organização. Soluções de prateleira fornecem preço final menor pois possuem maior escala de vendas.

Por outro lado, a indústria se aperfeiçoa e novas necessidades organizacionais surgem, seja pela competitividade em um setor, forçando a organização a se adaptar, seja por melhoria nos processos. Nesse sentido, a geração de informação tende a aumentar, pois a tomada de decisões, em uma organização madura, está intrinsecamente ligada a qualidade da informação que a mesma possui (PORTER; MILLAR, 1985). Para novas necessidades, novas soluções

precisam ser desenhadas para suportá-las. Nesse contexto, o desenvolvimento de sistemas de software big data configura um desafio particular, uma vez que técnicas e conhecimentos tradicionais de codificação não são suficientes para endereçar os problemas encontrados na área (ROSENTHAL *et al.*, 2015). O estudo de Rosenthal *et al.* (2015), ao acompanhar desenvolvedores de software com pouco ou nenhum treinamento formal no processo de análise de dados, discorre que o uso de processos e ferramentas *ad hoc*, dada a falta de expertise de profissionais, incorrem em riscos em projetos de software big data.

Este capítulo se encontra organizado da seguinte forma: a Seção 2.2 discorre sobre sistemas big data, suas características e terminologia; a Seção 2.3 define o contexto que este trabalho irá focar quanto à relação entre big data e engenharia de software; por fim, na Seção 2.4, citamos os trabalhos relacionados.

2.2 SISTEMAS BIG DATA

Essa seção pretende apresentar os conceitos básicos referente a sistemas big data e as diferenças fundamentais entre sistemas de software tradicionais e sistemas com uso intensivo de dados.

2.2.1 TERMINOLOGIA

Como o processo de desenvolvimento de sistemas big data é algo novo no campo da engenharia de software, é possível encontrar termos e conceitos similares na área de estudo. Estas comumente são fontes de informação onde não há uma revisão de conteúdo por pares, como blogs e artigos, posicionando sistemas *ultra-large-scale* como sistemas big data. Apesar do primeiro ter a possibilidade de ser caracterizado como um sistema big data, por também conter grande quantidade de dados armazenados, acessados e manipulados, a dimensão da extração de valor para tomada de decisões pode muitas vezes não estar presente. Um outro exemplo de posicionamento errado é quando na utilização do termo “*big data analytics*”, que segundo Clesti *et al.* é descrito como um conjunto de processos para coletar, organizar e analisar uma grande quantidade de dados heterogêneos. Em outras palavras, é o emprego de técnicas analíticas em conjuntos de dados distintos e extensos. Dessa forma, é evidente que “*big data analytics*” não se caracteriza como um sistema big data. Entretanto, é importante notar que um sistema big data pode suportar processos específicos deste, mas não substitui o processo de análise como um todo.

Por último, encontramos na literatura referências a sistemas big data como *data-intensive systems* (ANDERSON; SCHRAM; ALZARABAH; PALEN, 2013), que traduzido

em sentido aplicável significa sistemas com uso intensivo de dados. A palavra intensivo pode contextualizar o desenvolvimento de aplicações corporativas que fazem uso de tecnologias big data, como Hadoop/MapReduce e NoSQL (CASALE *et al.*, 2015) ou sistemas que empreendem uma coleta e análise de dados em larga escala (JENNINGS *et al.*, 2016). Este trabalho tem interesse nas duas contextualizações.

2.2.2 CARACTERÍSTICAS

Na literatura é possível encontrar diferentes definições sobre sistemas big data. Autores discorrem sobre as diferenças encontradas no desenvolvimento de sistemas tradicionais em comparação com sistemas big data, aqueles onde há um foco em alta taxa de processamento de dados.

Segundo Chen *et al.* (2016), o desenvolvimento de sistemas big data incorre em um número maior de riscos em comparação com o desenvolvimento de sistemas de software tradicionais, aqueles onde o conjunto de informações tratados pelo sistema se dá em menor escala. Dentre os riscos mencionados pelos autores, se destacam: a complexa integração entre sistemas antigos e novos, rápidas mudanças tecnológicas e a dificuldade na seleção de tecnologia big data. Em adição, Chen *et al.* (2016) argumenta que no desenvolvimento de sistemas tradicionais (os autores se referem a este como *small-data system development*), decisões sobre a arquitetura do software são relativamente triviais e, no contexto de sistemas big data, o design da arquitetura se torna crítico e impõe grandes riscos se realizado de maneira incorreta.

Em outro trabalho, Gorton e Klein (2015) discorrem que os requisitos de sistemas corporativos tradicionais são geralmente restritos em relação a crescimento de dados e escalabilidade. Dessa forma, Gorton e Klein (2015) argumentam que sistemas big data compartilham quatro requisitos fundamentais:

- Aptidão para suportar altas taxas de escrita
- Suporte a carga de requisição variáveis (adição e liberação de recursos de processamento sob demanda)
- Suporte a diversas cargas de consulta de dados (requisições que requerem rápidas respostas e requisições de longa duração em grandes coleções de dados)
- Alta disponibilidade

Em outro estudo, Madhavji *et al.* (2015) explica que sistemas big data se caracterizam por serem soluções compostas por diversos componentes, como bancos de dados, nós distribuídos, redes, middleware e camadas de tecnologias de *business intelligence*. Madhavji *et al.* (2015) adiciona que, dado esse contexto, há desafios no desenvolvimento, teste e manutenção, uma vez que a falha de um componente ao interagir com outros, resulta na degradação a qualidade operacional.

2.3 BIG DATA E ENGENHARIA DE SOFTWARE

De acordo com um dos workshops da área de engenharia de software e big data, *International Workshop on Big Data Software Engineering*, big data no contexto da engenharia de software pode ser posicionada em duas perspectivas (BIGDSE, 2016). A primeira, big data para engenharia de software, se refere a tecnologias e soluções big data para o campo da engenharia de software. Nesse contexto, a aplicação de big data *analytics* para manipulação de conjuntos massivos de dados a fim de se identificar desvios e padrões de falha em sistemas é um exemplo. Por outro lado, o termo “engenharia de software para big data” endereça os desafios na área de estudo da engenharia de software no contexto da construção de sistemas de software big data. Neste, incluem-se técnicas de levantamento de requisitos, padrões de arquitetura de software, linguagens para big data, desenvolvimento de frameworks e restrições como custo e qualidade em projetos de software big data.

Alguns autores iniciaram estudos na área de big focando em ferramentas e metodologias voltadas ao desenvolvimento de sistemas big data e sentimos a necessidade de realizar um mapeamento que pudesse dar uma visão geral sobre as abordagens sendo usadas e os principais objetivos na área.

2.4 BIG DATA E CONTEXTUALIZAÇÃO NESTE TRABALHO

Para este trabalho, é preciso de definir big data no contexto do desenvolvimento de sistemas de software. Assim, com base nas definições e conceitos encontrados no Capítulo 1, podemos identificar big data como o processo de aquisição e armazenamento de grandes conjuntos de dados com o objetivo de suportar a análise de dados e prover conhecimento para o processo de tomada de decisão.

Em seguida, como exposto na subseção anterior, este trabalho foca especificamente na dimensão engenharia de software para big data, isto é, analisando abordagens da área de estudo da engenharia de software que habilitem ou auxiliem o desenvolvimento de sistemas big data. Dessa forma, baseado nas definições de sistemas big data encontradas na seção

2.2.2, compreende-se sistemas de software big data como qualquer solução de software que empreenda a coleta, armazenamento e processamento de grande volume de dados.

Segundo Gandomi e Haider (2015), dimensões de tamanho em big data se referem a múltiplos terabytes e petabytes de dados, assim, este trabalho também faz uso dessa grandeza quando se refere a grandes volumes de dados ou processamento intensivo de dados. Portanto, big data, no contexto desse trabalho, está intrinsecamente ligado a desenvolvimento de sistemas de software.

2.5 TRABALHOS RELACIONADOS

Big data tem atraído atenção particular desde o início da década, sendo o tópico de pesquisa de diversas publicações recentes. Isto fez surgir quatro mapeamentos sistemáticos com focos diferentes, dada a vasta área de tópicos em big data. Nessa seção, vou discutir em ordem cronológica os objetivos e resultados de cada estudo.

Para a busca dos estudos secundários relacionados a este trabalho, além da técnica de *snowballing*, introduzida em detalhes na seção 3.4.2, a seguinte string de busca foi definida para busca nas bibliotecas digitais:

“systematic mapping study” AND “Big Data”

A Scopus foi a única biblioteca digital em que a string de busca foi aplicada, uma vez que outras bibliotecas falharam em retornar mapeamentos sistemáticos sobre o tópico big data. É importante mencionar também que a string de busca foi aplicada somente ao título. Os estudos retornados podem ser acessados na Tabela 1.

Tabela 1 - Trabalhos relacionados

Título	Autores	Ano	Fonte
Research on Big Data – A systematic mapping study	Akoka, J., Comyn-Wattiau, I., Laoufi, N.	2017	Computer Standards and Interfaces 54, pp. 105-115
Big data DBMS assessment: A systematic mapping study	Ortega, M.I., Genero, M., Piattini, M.	2017	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 10563 LNCS, pp. 96-110
A Systematic Mapping Study for Big Data Stream Processing Frameworks	M. Alayyoub, A. Yazici, Z. Karakaya	2015	International Conference on Digital Information Management (ICDIM), 2016
Big data in manufacturing: a	O'Donovan, P., Leahy, K., Bruton,	2015	Journal of Big Data 2(1),20

systematic mapping study	K., O’Sullivan, D.T.J.		
-----------------------------	---------------------------	--	--

O’Donovan *et al.* (2015) apresentou o primeiro estudo secundário na área. Por um mapeamento sistemático, O’Donovan *et al.* (2015) pretenderam prover uma contextualização de estudos no campo big data para a indústria manufatureira (questão de pesquisa primária: Como as tecnologias big data tem sido utilizadas na indústria manufatureira?), classificar as análises empregadas em estudos (questão de pesquisa: Que tipo de analytics tem sido empregados na área de big data para a indústria manufatureira?) e identificar as áreas de aplicação de tecnologias big data (questão de pesquisa: Que áreas da indústria manufatureira possuem tecnologias big data sendo aplicadas). De acordo com os autores, o estudo forneceu uma revisão pioneira da aplicação de tecnologias big data na indústria, bem como uma base para estudos futuros e maiores investigações na área. Em outro estudo secundário, Akoka *et al.* (2017) empreenderam um mapeamento inicial da área respondendo as seguintes questões de pesquisa:

- Quantas artigos de pesquisa foram produzidos?
- Qual é a tendência anual de publicações?
- Quais são os “*hot topics*” na pesquisa em big data? Quais são os tópicos mais investigados em big data?
- Por que a pesquisa é realizada?
- Quais são os artefatos de pesquisa mais frequentemente obtidos?
- O que a pesquisa em big data produz?
- Quais são os autores mais ativos?
- Que periódicos incluem pesquisas em big data?
- Quais são as disciplinas ativas?

Como resultados, os autores apresentaram:

- 11 artefatos produzidos por pesquisas em big data
- 5 “*hot topics*” em big data
- 11 objetivos de pesquisa em big data
- 12 domínios de aplicação onde pesquisas em big data se aplicaram
- Lista dos autores mais ativos da área

Por outro lado, em outro importante trabalho relacionado, Alayyoub *et al.* (2015) realizaram um mapeamento sistemático com foco em coletar estudos sobre frameworks de processamento de stream para big data e fornecer uma classificação e análise desses estudos. Diversos frameworks relatados no estudo, como o Spark e Storm, são implementados por sistemas big data para processamento de grande volume de dados. Os principais objetivos foram identificar as principais contribuições dos estudos, as ferramentas para ingestão de dados, as propostas de contribuição (aprimoramento de uso, aprimoramento de performance e outros), as formas de experimentação (apenas stream, apenas batch e ambos), os tipos de dados e número de nós utilizados em experimentos. Como resultado, os estudos foram classificados em 11 tipos de contribuição (Método/Técnica/Approach, Framework, Modelo, Ferramenta, Arquitetura, Plataforma, Estudo Empírico, Análise, Comparação, Visão Geral e Outros). Em adição, foram identificadas 7 ferramentas de ingestão de dados (por exemplo, Kafka, Rabbit e ZeroMQ) e 6 tipos de dados (sensor, gráfico, geoespacial, logs, conteúdo web e mídia social). Os frameworks utilizados para as classificações foram Spark, Storm, Flink, InfoSphere e S4.

Mais recentemente, Ortega *et al.* (2017) realizaram um mapeamento sistemático com o objetivo de identificar os principais métodos de avaliação de bancos de dados no contexto big data com foco em características de qualidade. Especificamente, a análise foi realizada com respeito às técnicas e medidas utilizadas para avaliar as características de qualidade de sistemas de gerenciamento de banco de dados (SGBDs). Os autores encontraram que as principais características de qualidade avaliadas nos estudos primários, baseados no padrão ISO/IEC 25010, são eficiência de performance, usabilidade, adaptabilidade e disponibilidade. Em seguida, foi identificado que os SGBDs mais avaliados no contexto big data são liderados por Cassandra e MongoDB, seguidos por Hadoop e HBase. Por último, os autores também explicitaram as principais técnicas empregadas pelos estudos ao avaliar a qualidade de SGBDs big data. As técnicas e sua respectiva porcentagem estão abaixo:

- Propostas de benchmark (53.63%)
- Yahoo! Cloud Serving Benchmark (YCSB) (31.58%)
- YCSB++ (5.26%)
- LUBM benchmark (5.26%)
- TPCx-HS (5.26%)
- BigDataBench (5.26%)

As principais notas dos autores se referem a falta de maturidade na escolha de um SGBD para big data, uma vez que a aplicação de um complexo sistema como o Hadoop não é necessário na maioria dos estudos identificados. Em adição, foi identificado que os SGBDs mais avaliados são utilizados para prover eficiência de performance.

Como pode ser notado, os estudos de O'Donovan et al. (2015) (QP3), Alayyoub *et al.* (2015) (QP1) e Akoka *et al.* (2017) (QP1c e QP2) apresentaram questões de pesquisa presentes neste mapeamento. Entretanto, dado o foco específico deste trabalho em termos de escopo, muitas questões de pesquisa não se apresentaram correlatas entre este e outros estudos relacionados.

2.6 CONSIDERAÇÕES FINAIS

Neste capítulo foram introduzidos aspectos estratégicos no que concerne o desenvolvimento de soluções de software big data para as organizações. Em seguida, foram apresentadas as características de sistemas big data e principais definições encontradas na literatura, bem como as dimensões da engenharia de software para o contexto de big data e, por conseguinte, o foco da dimensão deste trabalho, engenharia de software para big data. Em adição, este capítulo teve por intenção posicionar o desenvolvimento de sistemas big data quanto à nomenclatura utilizada na academia, discorrendo sobre as principais diferenças entre os termos empregados. Também foram apresentados os trabalhos relacionados da literatura, onde os estudos secundários correlatos a esse trabalho foram analisados quanto ao seus objetivos e resultados.

CAPÍTULO 3 – MAPEAMENTO SISTEMÁTICO SOBRE DESENVOLVIMENTO DE SISTEMAS BIG DATA

3.1 INTRODUÇÃO

Nos capítulos 1 e 2, por meio da contextualização das características de sistemas big data e principais termos encontrados na literatura, no que se refere a soluções de softwares que fazem uso intensivo de dados, bem como a revisão de estudos correlatos na área, foi possível obter uma visão geral da área e o tópico foco deste estudo.

A proposta deste trabalho é contribuir para o campo da engenharia de software, no contexto do desenvolvimento de sistemas big data. Para isso, tem por objetivo promover um mapeamento do atual estado da pesquisa em sistemas com uso intensivo de dados, focando nos aspectos que permitiram o desenvolvimento do estudo, isto é, caracterizando as técnicas e/ou estratégias empregadas, os objetivos endereçados, o nível de maturidade de cada abordagem, o grau de integração entre a indústria e a academia, os domínios de aplicação e os tipos de contribuição.

Este capítulo descreve a metodologia implementada por este trabalho. A Seção 3.2 discorre sobre como se deu a implementação do processo de mapeamento, isto é, o protocolo definido para este trabalho, incluindo as questões de pesquisa, a estratégia de busca e termos utilizados, bem como o processo e critério de seleção. Por fim, nosso esquema de classificação é apresentado, apresentando a aplicação dos critérios definidos em cada etapa do processo.

3.2 PROTOCOLO DO MAPEAMENTO

O protocolo do mapeamento sistemático aplicado a essa pesquisa, isto é, o conjunto de etapas aplicadas para este estudo, será apresentado a seguir.

3.2.1 DEFINIÇÃO DAS QUESTÕES DE PESQUISA

Para esse estudo, uma questão primária de pesquisa foi definida: **“(QP1) Que abordagens da engenharia de software tem sido propostas para o suporte e construção de sistemas big data? ”**. Baseado na questão primária, questões de pesquisa complementares foram definidas. Respondendo essas questões, teremos uma caracterização detalhada dos estudos:

QP1: Que abordagens da engenharia de software tem sido propostas para o suporte e construção de sistemas big data?

De acordo com Madhavji *et al.* (2015), no campo de estudo de big data, há um menor foco no desenvolvimento de aplicações de software em comparação com *data analytics* e desenvolvimento de infraestrutura. Em conjunto com o já citado problema nas organizações sobre a falta de processos padronizados para projetos big data (CAPGEMINI; INFORMATICA, 2016), identificar as abordagens propostas para o desenvolvimento de sistemas com uso intensivo de dados se torna particularmente importante no contexto da seleção de tecnologias apropriadas e estratégias para construção de aplicações por pesquisadores e profissionais da indústria.

Nesse contexto, eu pretendo organizar um conhecimento sobre a pesquisa existente na área de desenvolvimento de aplicações big data. Assim, essa questão visa abordar os processos, métodos, técnicas e estratégias no campo do desenvolvimento de aplicações big data. As próximas questões de pesquisa complementares foram derivadas dessa questão primária. Respondendo essas questões, teremos uma contextualização detalhada dos estudos no campo.

QP1a: Quais os objetivos de estudo das abordagens propostas para o suporte e construção de sistemas big data?

Tão importante quanto identificar as abordagens, é caracterizar o objetivo fim da abordagem da engenharia de software proposta. Uma abordagem, quando proposta, tem por objetivo solucionar um problema ou endereçar uma lacuna encontrada em um tópico de estudo. Essa questão visa, para cada proposta identificada, identificar os objetivos correntes da área de estudo. Essa associação entre abordagem e seu respectivo objetivo irá prover, na escolha de uma abordagem em particular, informações precisas de que objetivo tal proposta pode endereçar.

QP1b: Que avaliações empíricas têm sido utilizadas para a apresentação de propostas de abordagens?

Além da identificação das abordagens para a construção de sistemas big data, é importante avaliar com qual o rigor de pesquisa os mesmos têm sido propostos. A literatura prevê diferentes tipos avaliação, como *survey*, experimento e estudos de caso.

Essa questão de pesquisa tem por objetivo identificar que estudos empíricos tem sido utilizados para endereçar as abordagens para o desenvolvimento de sistemas big data na

academia. Dessa forma, é possível identificar o nível de maturidade das propostas, uma vez que experimentos são avaliados como estudos de maior impacto científico. Muitos experimentos, então área mais imatura. Muitos estudos de caso, maior nível de maturidade.

QP1c: Quais são os tipos de artigos de pesquisa propondo abordagens da engenharia de software para sistemas big data?

De acordo com a classificação de Wieringa *et al.* (2006), o objetivo dessa questão pesquisa é identificar que tipo de pesquisa as abordagens da engenharia de software propostas estão empregando. Dentre os tipos de pesquisa, temos: *Evaluation Research*, *Validation Research*, *Solution Research*, *Philosophical Paper*, *Experience Report* e *Opinion Paper*.

QP1d: Para quais domínios de aplicação as abordagens propostas tem sido aplicados (e quantos artigos cobrem os diferentes domínios de aplicação)?

Além de identificar as áreas de pesquisa, a identificação dos domínios de aplicação onde as abordagens são propostas também tem importância. Segundo Klein e Gorton (2015), domínios como linhas aéreas e *healthcare* possuem grande janela para captura, gerência e análise de dados, portanto, apresentam-se com alta viabilidade de aplicação de sistemas big data.

O objetivo desta questão é delinear os domínios de aplicação recorrentes no corpo de estudo da área, uma vez que é comum estudos, no desenvolvimento de soluções ou na proposta de metodologias que habilitem um melhor processo de desenvolvimento, utilizem um domínio de aplicação como base para aplicar a abordagem ou mesmo para resolver um problema específico de um domínio. Obter essa informação é importante para determinar a variedade de domínios sendo endereçados nos estudos atuais e quais deles possuem mais estudos, caracterizando assim maior amadurecimento no domínio.

QP1e: A quais fases do ciclo de vida do desenvolvimento de software as abordagens propostas se aplicam?

O objetivo dessa questão é identificar a que fases do ciclo de vida do desenvolvimento de software as abordagens propostas para sistemas big data têm aplicabilidade. Endereçar essa questão nos ajudará a entender, por exemplo, se há uma concentração de abordagens em dada fase (dado que pode levar a problemas contínuos na área de estudo) ou fases do ciclo de vida preteridas nos estudos.

QP1f: Que tipos de contribuição de pesquisa tem sido realizadas pelas abordagens propostas?

Baseado na questão de pesquisa de O'Donovan *et al.* (2015), esta tem por intenção identificar qual a natureza da contribuição da proposta para a área. Uma vez que as abordagens são conhecidas, é fundamental identificar os tipos de contribuição em que os esforços estão sendo empregados no desenvolvimento de aplicações big data. Madhavji *et al.* (2015) observa que desafios específicos são encontrados em áreas da engenharia de software, como engenharia de requisitos, arquitetura de software, teste e manutenção. Entretanto, a importância na identificação do tipo de contribuição se dá pela necessidade de verificação da existência de desafios em outras áreas de pesquisa, como modelagem e design de sistemas. Isto tem por objetivo enriquecer o escopo deste trabalho, contribuindo para o amadurecimento das abordagens sendo propostas na construção de sistemas big data. Pela identificação dos tipos de contribuição que as abordagens proveem, será possível identificar brechas, pontos não explorados, e, por conseguinte, as contribuições mais abordadas nas pesquisas existentes.

QP1g: Qual o grau de colaboração entre a indústria e a academia na proposta de abordagem de cada estudo?

Esta questão de pesquisa tem por objetivo delinear qual o nível de colaboração entre autores acadêmicos e autores que atuam na indústria. Como explicitado na questão de pesquisa QP1b, há uma gama de domínios onde big data se apresenta como um importante ativo para aplicação, como o de linhas aéreas e manufatura. Um alto nível de colaboração fortalece a crença de que os estudos primários identificados neste trabalho estão endereçando desafios reais das organizações.

QP1f: Em que foros os estudos propondo abordagens para a construção de sistemas big data estão sendo publicados?

No que concerne os foros de pesquisa no contexto deste trabalho, há uma conferência especialmente dedicada à big data no contexto da engenharia de software, a *International Workshop on BIG Data Software Engineering*. Também há outras focadas em big data em um aspecto mais amplo, que abarcam outras áreas da literatura, como a *International Congress on Big Data*. Estudos apresentados nestes foros estão presentes neste estudo. Entretanto, é importante pesquisar também possíveis focos de trabalhos no contexto da construção de sistemas big data em outros locais. Por fim, o objetivo dessa questão é identificar os principais foros que estão publicando os approaches e/ou estratégias para a construção de sistemas big data.

3.2.2 ESTRATÉGIA DE BUSCA

A estratégia de busca definida para um mapeamento sistemático influencia diretamente no esforço empregado ao recuperar os estudos, bem como na completude dos estudos primários. Dessa forma, um método de busca adaptado para o contexto deste trabalho foi elaborado para realizar a busca de estudos relevantes.

Primeiro, os seguintes artigos de controle, exibidos na Tabela 2, foram selecionados para prover insumo na busca de estudos primários:

Tabela 2 - Artigos de controle selecionados

Título	Autores	Ano	Fonte
Strategic Prototyping for Developing Big Data Systems	Hong-Mei Chen, Rick Kazman, Serge Haziyeu	2016	IEEE Software
Distribution, Data, Deployment Software Architecture Convergence in Big Data Systems	Ian Gorton, John Klein	2015	IEEE Software
Software Engineering for Big Data Systems	Ayse Basar Bener, Ian Gorton, Audris Mockus	2016	IEEE Software

Artigos de controle são artigos identificados previamente que poderiam ser incluídos e que, portanto, a string de busca deveria retornar. Ou seja, a string de busca deve ser ajustada para retorná-los. Neste momento do processo de mapeamento, os critérios de inclusão e exclusão ainda não estavam definidos. Estes, em um mapeamento sistemático costumam ser definidos no processo de seleção de estudos e neste trabalho não é diferente. Dessa forma, durante o processo de seleção, algum dos artigos de controle podem vir a ser excluídos por não se adequar aos critérios de inclusão.

Em seguida, com base nos artigos de controle, um conjunto inicial de estudos foram selecionados por meio de uma busca manual em foros de interesse (por exemplo, periódicos e conferências), uma vez que há foros específicos para este tópico de estudo, como o *International Workshop on BIG Data Software Engineering*. Em seguida, com uma visão geral do escopo da área, onde se identificam os principais termos usados e estudos importantes, outros estudos primários foram selecionados a partir de um conjunto de bibliotecas digitais pré-selecionadas. Os detalhes do método de busca definido podem ser encontrados nas seções a seguir.

A princípio, as seguintes palavras-chave foram escolhidas para executar a busca nas bibliotecas digitais:

População:

- *Big data*;
- *Software, system*;

Intervenção:

- *Practice, technique, method, process;*
- *Requirement, design, development, testing.*

A string de busca derivada das palavras-chave está presente na Tabela 3.

Tabela 3 - String de busca

((("big Data") AND ("Software" OR "System"))
AND
("Practice" OR "Technique" OR "Method" OR "Process")
AND
("Requirement" OR "Design" OR "Development" OR "Testing")

A definição da string de busca se baseou no objetivo de cobrir os aspectos mais relevantes no desenvolvimento de sistemas de software. Dessa forma, fases do ciclo de vida e palavras que remetem a possíveis approaches/estratégias são incorporadas. Entretanto, baseada na biblioteca digital Scopus (escolhida por indexar um conjunto significativa de estudos de outras bibliotecas digitais), a busca retornou um inesperado conjunto de resultados (18,947). Foi observado que a maioria não atendia aos objetivos de nosso estudo, muitas vezes apresentando resultados fora do escopo pretendido e/ou fora do domínio de desenvolvimento de software. Essa característica nos resultados promoveria um alto grau de esforço para excluir os trabalhos fora do critério de exclusão. Prosseguimos então com a tática de limitar a estratégia de busca para apenas Título e Abstract. Os resultados ainda apresentavam um número de resultados que inviabilizava a classificação dos estudos em tempo hábil para este trabalho (1,278). Em adição, o problema de resultados fora do domínio de desenvolvimento de software persistiu. Desta forma, afim de obter um número factível, porém sólido, conjunto de resultados, uma estratégia de busca mais geral foi escolhida baseada nas seguintes palavras-chave:

População:

- *Big data;*
- *Software engineering.*

Para estas palavras-chave, a string de busca exibida na Tabela 4 foi definida. A mesma foi aplicada ao título e ao resumo de cada estudo. A fim de evitar o problema de um grande conjunto de resultados com estudos em domínios diferentes de desenvolvimento de software, *full text search* não foi aplicada, isto é, o motor de busca da biblioteca digital não percorreu o texto completo de cada estudo buscado para identificação da string de busca submetida. Esta apresentou o retorno de resultados consistentes com a área da engenharia de software (57).

Tabela 4 – String de busca genérica

(("big Data") AND ("Software engineering"))

3.3 ESCOPO DA BUSCA

Nesta subseção são identificados aspectos importantes sobre as características da busca de estudos primários realizada, como as bibliotecas digitais onde a string de busca foi submetida, bem como o período de tempo.

3.3.1 PERÍODO DE TEMPO

Como mencionado por Chen *et al.* (2015), pesquisas sobre big data se encontram em sua fase inicial. Dessa forma, a restrição por limite de ano não é recomendada. É observado que os primeiros estudos mencionando sistemas com uso intensivo de dados datam de 2011 (SCHRAM; ANDERSON, 2011). A busca nas bibliotecas digitais foi realizada em Janeiro de 2017, portanto, este estudo contempla estudos primários realizados entre 2011 e 2016.

3.3.2 BIBLIOTECAS ELETRÔNICAS

As fontes de dados foram definidas com base no nível de relevância de revistas e conferências da área da Engenharia de Software. Uma vez que o tema big data é um tópico de estudo recentemente abordado pela literatura, nenhuma restrição de período de tempo foi aplicada. Nós selecionamos publicações recuperadas das bibliotecas digitais e motores de busca da web apresentadas na Tabela 1. As fontes de dados foram escolhidas com base em Chen *et al.* (2010), que recomenda estas para a área da engenharia de software, uma vez que habilitam o acesso a influentes jornais e conferências na área.

As bibliotecas eletrônicas foram escolhidas em conjunto com meus orientadores, visando abarcar o maior número de fontes de estudos de qualidade possível. Como cada biblioteca digital possui um motor de busca diferente (aplicando assim diferentes regras de formatação), a string de busca precisava ser convertida para a string de busca nativa da biblioteca digital. A Tabela 5 exhibe as bibliotecas digitais onde estudos foram buscados durante esse estudo, seguido pela respectiva string de busca utilizada e o número de estudos retornados pela aplicação da string de busca.

Tabela 5 – String de busca para cada biblioteca digital

Biblioteca digital	String de busca	Número de resultados
Scopus	((("big Data") AND ("Software engineering"))	57
IEEE Xplore	((("Document Title":big data AND "Document Title":software engineering) OR (p_Abstract:big data AND "Abstract":software engineering))	106
ACM	(acmdlTitle:(+"big data") AND acmdlTitle:(+"software engineering")) OR (recordAbstract:(+"big data") AND recordAbstract:(+"software engineering"))	54
ScienceDirect	(TITLE("big data") and TITLE("software engineering")) OR (ABSTRACT("big data") and ABSTRACT("software engineering"))	7
EI Compendex	((("big data") WN TI) AND (("software engineering") WN TI)) OR (((("big data") WN AB) AND (("software engineering") WN AB))	54
Web of science	(TI=("big data") AND TI=("software engineering")) OR (TS=("big data") AND TS=("software engineering"))	163

Após a submissão da string de busca e recuperação dos estudos, foi possível observar o retorno de muitos artigos que apareceram em mais de uma biblioteca digital. De um total de 441 artigos obtidos pela busca, após a remoção de artigos duplicados, restaram 305 estudos primários.

3.4 SELEÇÃO DOS ESTUDOS

Nesta Subseção, o processo de seleção dos estudos é explicitado. Primeiro, os critérios de inclusão e exclusão são definidos. Para a inclusão de apenas estudos relevantes, a definição dos critérios é um importante instrumento para a qualidade de um mapeamento. Em seguida, o processo de seleção dos estudos primários é delineado, incluindo a aplicação dos critérios de inclusão e exclusão, onde cada etapa é detalhada.

3.4.1 DEFINIÇÃO DOS CRITÉRIOS DE INCLUSÃO E EXCLUSÃO

De acordo com Petersen *et al.* (2008), apenas estudos que são relevantes para responder as questões de pesquisa devem ser incluídos. Portanto, essa pesquisa excluiu os estudos que apresentam conceitos e propostas fora do escopo das questões de pesquisa ou fora do domínio da engenharia de software. Mesmo com a mudança para uma string de busca genérica, ainda assim houve o retorno de um conjunto alto de estudos que não são relevantes para esse trabalho. Dessa forma, uma filtragem precisou ser realizada a fim de excluir os estudos irrelevantes. Para isto, foi definido um conjunto de critérios.

Critério de inclusão:

I1: A publicação está relacionada a desenvolvimento de software com uso intensivo de dados. Estudos que tratam de big data, porém fora do contexto do desenvolvimento de software não devem ser incluídos.

I2: Estudos publicados que descrevem approaches ou estratégias para construção de sistemas big data. Se o estudo apenas menciona uma técnica e não provê uma explicação detalhada sobre sua aplicabilidade e contexto, então não deve ser incluído.

I3: Publicações revisados por pares (*Peer reviewed*). Estudos publicados em periódicos, conferências e workshops. Um estudo não revisado por pares é considerado impróprio e não incluído.

Critério de exclusão:

E1: Estudos publicados na forma de *abstracts*, tutoriais, apresentações Power Point, *White Papers*, reporte de workshop ou conferência. Estes não proveem o rigor metodológico requerido para uma pesquisa científica. Dessa forma, falham em não corresponder ao que se é buscado nas questões de pesquisa e assim são excluídos desta pesquisa.

E2: Publicações onde foco em engenharia de software não é identificado

E3: Publicações duplicadas

E4: Publicações onde a língua não é o inglês

3.4.2 PROCESSO DE SELEÇÃO

A identificação e filtragem dos artigos foi dividida em 7 etapas, como mostrado na Tabela 6, utilizando uma combinação de buscas em bibliotecas digitais e *forward* e *backward snowballing*. Cada etapa é detalhada a seguir.

Tabela 6 - Utilização de critério de seleção na seleção dos estudos

Etapa de seleção	Critério utilizado
1ª etapa – aplicação de critérios em artigos de controle	I1, E2
2ª etapa – definição de conjunto inicial	I1, E2
3ª etapa – seleção por string de busca	Não se aplica
4ª etapa – seleção por tipo de publicação	E1, E3, E4
5ª etapa – seleção por metadados	I1, I3
6ª etapa – seleção pelo resumo	I1, I2, E2
7ª etapa – seleção por leitura do texto completo	I1, I2, E2

A primeira etapa revisa os artigos de controle com o objetivo de definir se os mesmos atendem aos critérios de inclusão deste trabalho. Por mais que os artigos de controle devam ser retornados pela string de busca, isso não significa necessariamente que os mesmos atendam ao rigor dos critérios de inclusão deste trabalho. Dessa forma, ao revisar os 3 artigos de controle, o estudo de Gorton *et al.* (2016) foi removido por não realizar a proposta de nenhuma abordagem para construção de sistemas de software big data, e sim fornecer apenas um panorama geral da área.

A segunda etapa consistiu na identificação de um conjunto inicial de estudos por meio de buscas exploratórias na base de dados Scopus. Para tal, o ponto de partida foram estudos conhecidos previamente, cunhados na literatura como artigos de controle (ver Tabela 1).

Artigos de controle tem por objetivo auxiliar na definição da string de busca, provendo insumo para o ajuste da mesma uma vez que a busca deve retornar os artigos de controle (KITCHENHAM, 2004). Assim, por meio de referências e links para publicações citadas, 10 publicações foram encontradas e estudadas para obter uma visão geral da área; desafios encontrados; classificações utilizadas; termos e palavras-chave importantes. As publicações encontradas foram divididas por tópico de estudo, com a finalidade de auxiliar a definição de categorias de pesquisa nesse estudo. O único critério de seleção utilizado nesta etapa foi o posicionamento do tópico de estudo, que deve estar especificamente relacionado a construção de sistemas big data. A divisão pode ser acessada na Tabela 7.

A terceira etapa consistiu na definição da string de busca e posterior busca dos estudos utilizando a mesma. A string de busca foi definida com base no conjunto inicial de estudos identificados. A busca dos estudos foi realizada em cada uma das bibliotecas digitais previamente selecionadas para esse estudo e encontradas na Tabela 5.

Na quarta etapa, após obtenção dos resultados da busca, a primeira filtragem é aplicada, baseada nos critérios E1, E3, E4. Os critérios de exclusão são aplicados, removendo todos os abstracts, apresentações PowerPoint, reportes técnicos, estudos duplicados e

publicações que não passaram por revisões aos pares. Este objetivou evitar redundância nas contribuições de pesquisa e estabelecer um nível de qualidade nas publicações incluídas.

Tabela 7 - Conjunto inicial de artigos selecionados

Tópico de estudo	Artigo
Metodologia de desenvolvimento	Big Data System Development An Embedded Case Study with a Global Outsourcing Firm (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015)
	Engineering Big Data Solutions (MOCKUS, 2014)
	On the Model Design of Integrated Intelligent Big Data Analytics Systems (CHEN; LI; WANG, 2015)
	Toward Scalable Systems for Big Data Analytics: A Technology Tutorial (CHUA <i>et al.</i> , 2014)
Modelagem	Modeling and Management of Big Data Challenges and opportunities (GIL; SONG, 2016)
Performance	Performance evaluation of SQL and MongoDB databases for big e-commerce data (ABOUTORABI; REZAPOUR; MORADI; GHADIRI, 2015)
Requisitos	Understanding Quality Requirements in the Context of Big Data Systems (NOORWALI; ARRUDA; MADHAVJI, 2016)
	Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems (PÄÄKKÖNEN; PAKKALA, 2015)
	PAUSE: A Privacy Architecture for Heterogeneous Big Data Environments (BODORIK, 2015)
Infraestrutura	Real-Time Big Data the JUNIPER Approach (AUDSLEY; CHAN; GRAY; WELLINGS, 2014)

Na quinta etapa um segundo filtro é aplicado. A filtragem foi executada pela verificação de metadados como título, palavras-chave e nome do foro, utilizando os critérios I1, I2 e E2. Quando a decisão não era possível, o estudo procedia para a próxima etapa, onde o mesmo era identificado pelo seu resumo. Após essa etapa 46 estudos avançaram para a próxima etapa de filtragem.

Na sexta etapa, a filtragem foi realizada pela leitura do resumo de publicações selecionadas na quarta etapa e aplicação dos critérios I1, I2, E2. Se a inclusão de um estudo suscitasse dúvida, o mesmo era levado para discussão com o orientador. Quando a decisão não era possível, o estudo procedia para a próxima etapa, onde o mesmo era lido de maneira completa. Após essa etapa 43 estudos avançaram para a próxima etapa de filtragem.

Na sétima etapa, o último filtro foi aplicado. Desta vez, os artigos selecionados foram lidos em sua completude. Ao fim desta etapa, mais 12 artigos foram excluídos. Por exemplo, o artigo *Developing sustainable software solutions for bioinformatics by the “Butterfly” paradigm* (AHMED; ZEESHAN; DANDEKAR, 2014) reportou uma abordagem para lidar com desafios típicos em um ambiente acadêmico, como contratos de curto período de tempo e documentação inconsistente. Foi retornado pela submissão da string de busca por mencionar em seu resumo que novos desafios seriam big data. Entretanto, o foco do experimento não atendia ao critério de inclusão, isto é, falhou ao propor uma abordagem para lidar especificamente com projetos de software com uso intensivo de dados.

Snowballing

Após o processo de seleção dos estudos primários, a técnica de *snowballing* foi aplicada com o objetivo de evitar a não inclusão de potenciais estudos relevantes na pesquisa. Os estudos obtidos pela aplicação dessa técnica, subdivididos pelo tipo de *snowballing* empregado, são verificados na Tabela 8. *Snowballing* é um processo iterativo de seleção de estudos onde as referências dos estudos primários selecionados pelo processo de seleção são verificadas a fim de se selecionar outros estudos que possam compor a pesquisa (BUDGEN *et al.*, 2008) (WOHLIN, 2014). Neste estudo, duas formas de se implementar a técnica de *snowballing* foram utilizadas: *backward* e *forward snowballing*. A primeira se refere ao processo explicitado acima, onde as referências dos estudos são consultadas para se obter novos estudos. Por outro lado, *forward snowballing* se baseia na busca de estudos que citam os estudos primários selecionados. Para tal, utilizou-se a seção “*referenced by*” das bibliotecas digitais de cada estudo selecionado para se obter o conjunto de estudos a analisar.

Para ambos os tipos de *snowballing*, a aplicação dos critérios seguiu o mesmo padrão encontrado na fase de seleção de estudos: baseado em metadado, no resumo e leitura do texto completo. Ao fim, os artigos selecionados pelo processo de *snowballing* foram combinados com os artigos inicialmente selecionados. Dessa forma, acredita-se que a busca pelas publicações é suficientemente extensiva e o conjunto de publicações incluídas proveem um retrato adequado de pesquisas sobre o desenvolvimento de sistemas big data. A Tabela 8 descreve, para cada estudo encontrado pela técnica de *snowballing*, o tipo de *snowballing* empregado e o estudo de origem, isto é, o artigo citado ou qual artigo o cita.

Resultado final da seleção de estudos

O número final dos artigos selecionados após aplicação do processo de seleção em cada biblioteca digital é mostrado na Tabela 9, ordenados pela ordem de busca empreendida.

Tabela 8 – Estudos primários recuperados por tipo de snowballing

Tipo de snowballing	Estudo	Cita/é citado por
Forward	A tool for verification of big-data applications (BERSANI; ERASCU, 2016)	DICE: Quality-Driven Development of Data-Intensive Cloud Applications (CASALE <i>et al.</i> , 2015)
Forward	Agile Big Data Analytics for Web-Based Systems: An Architecture-Centric Approach (CHEN; KAZMAN; HAZIYEV, 2016a)	Big Data System Development: An Embedded Case Study with a Global Outsourcing Firm (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015)
Forward	Application-Specific Evaluation of No SQL Databases (DONOHOE; ERNST; KLEIN; MATSER; PHAM, 2015)	Distribution, Data, Deployment Software Architecture Convergence in Big Data Systems (GORTON; KLEIN, 2015)
Backward	Architectural Implications of Social Media Analytics in Support of Crisis Informatics Research (SCHRAM; ANDERSON, 2013)	Embrace the Challenges: Software Engineering in a Big Data World (ANDERSON, 2015)
Backward	Design and Implementation of a Data Analytics Infrastructure in Support of Crisis Informatics Research (NIER Track) (SCHRAM; ANDERSON, 2013)	Embrace the Challenges: Software Engineering in a Big Data World (ANDERSON, 2015)
Backward	Design Assistant for NoSQL Technology Selection (GORTON; KLEIN, 2015)	A reference architecture for big data systems in the national security domain (BLOCKOW; BUGLAK; COOPER; KLEIN; WUTTKE, 2016)
Backward	Design challenges/solutions for environments supporting the analysis of social media data in crisis informatics research (SCHRAM; ANDERSON, 2015)	Embrace the Challenges: Software Engineering in a Big Data World (ANDERSON, 2015)
Backward	Early Experience with Model-driven Development of MapReduce based Big Data Application (BELLARYKAR; KULKARNI; RAJBHOJ, 2014)	Towards a model-driven design tool for big data architectures (GUERRIERO; TAJFAR; TAMBURRI; NITTO, 2016)
Backward	MySQL to NoSQL Data Modeling Challenges in Supporting Scalability (SCHRAM; ANDERSON, 2012)	Embrace the Challenges: Software Engineering in a Big Data World (ANDERSON, 2015)

Alguns estudos que atenderam os critérios de inclusão foram recuperados em mais de uma biblioteca digital, dessa forma, foi computado apenas a primeira biblioteca digital que o mesmo apareceu (segundo a ordem de busca da biblioteca digital). Claramente, a maioria dos artigos selecionados tem como origem as bibliotecas digitais da ACM e Web of science. Estes estudos se juntam aos 9 estudos selecionados pelo processo de *snowballing* (ver Tabela 7), 2 artigos de controle e 10 artigos compondo o conjunto inicial, totalizando 52 estudos primários identificados por esse estudo.

Tabela 9 - Número de estudos primários selecionados por biblioteca digital

Biblioteca digital	Número de estudos primários
Scopus	1
IEEE	7
ACM	10
ScienceDirect	1
EI Compendex	3
Web of science	9

Como complemento as informações explicitadas nesta Seção, a Figura 3 exibe a seleção dos resultados em cada etapa da seleção dos estudos primários, a partir do retorno de resultados das bibliotecas digitais. Houve um total de 441 estudos retornados das buscas nas bibliotecas digitais, onde apenas 305 seguiram para o processo de validação uma vez que as bibliotecas digitais retornaram os mesmos estudos em alguns casos. Em seguida, a seleção por metadado forneceu o maior corte de estudos, onde apenas 46 seguiram para as etapas seguintes. Dessa forma, após as etapas de seleção pelo resumo e seleção por leitura do texto completo, o processo de *snowballing* foi aplicado. Ao final, 31 estudos retornados das bibliotecas digitais se somaram aos 9 estudos obtidos por *forward* e *backward snowballing*.

3.5 ESQUEMA DE CLASSIFICAÇÃO

Os atributos no esquema de classificação foram estruturados em 9 categorias com o fim de permitir uma análise aprofundada da informação contida nos artigos: abordagem da engenharia de software utilizada, objetivo do estudo, tipo do estudo, tipo da pesquisa, fase do ciclo de vida em que o approach e/ou estratégia é aplicada, biblioteca eletrônica, tipo do autor e domínio de aplicação. Cada categoria está relacionada a uma questão de pesquisa (como definido na seção 3.2.1). Essa referência pode ser acessada na Tabela 10.

As categorias são descritas a seguir:

Abordagem da Engenharia de Software

Essa categoria inclui informação sobre técnicas e estratégias para construção de

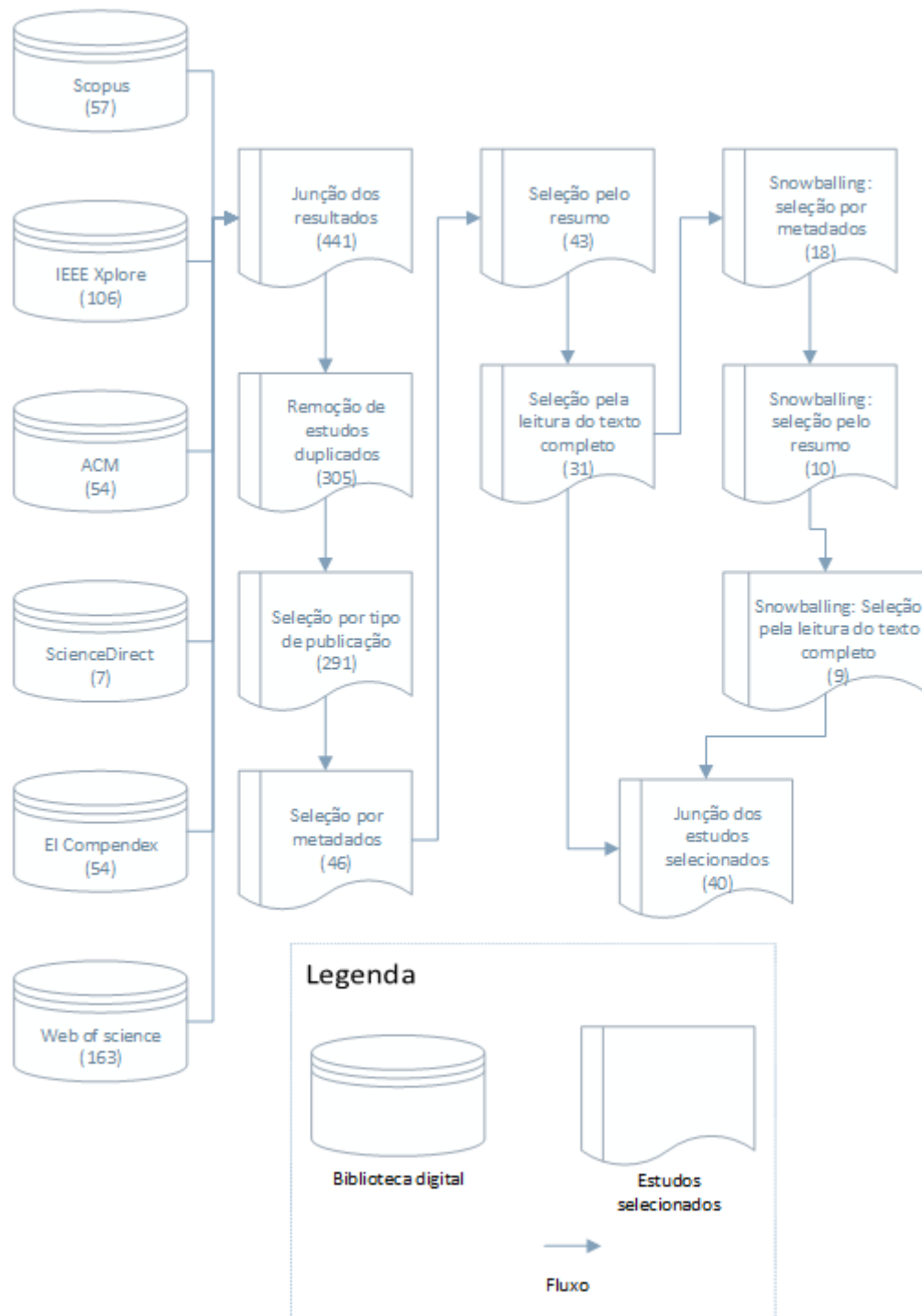


Figura 3 - Processo de seleção de estudos

sistemas big data. Pela natureza da área de estudo, ainda nova e com estudos ainda em processo de consolidação da maturidade, buscou-se nessa categoria identificar o meio pelo qual o estudo primário apresentado propôs ou se chegou a solução de um problema. Em outras palavras, para ser considerada uma técnica, approach, ou estratégia, o critério adotado é que tal estudo deve propor uma solução que suporte todo o ciclo (ou uma fase do ciclo) de desenvolvimento de software. Dessa forma, processos e tecnologias empregados, modelos de

desenvolvimento e propostas de arquitetura e modelagem são tópicos descritos nesta categoria.

Tabela 10 – Dados de itens extraídos de cada estudo primário

#	Nome do item	Descrição	Questão de Pesquisa
1	Ano	O ano de publicação do estudo	Nenhuma
2	Biblioteca digital	Biblioteca digital onde o estudo foi encontrado	Nenhuma
3	Abordagem	Técnica ou estratégia aplicada	QP1
4	Objetivo	Objetivo do estudo, que promoveu a aplicação da abordagem	QP1a
5	Estudo empírico	<i>Experiment, case study</i> ou <i>survey</i>	QP1b
6	Tipo de pesquisa	<i>Evaluation research, Proposal of solution, Validation research, Philosophical paper, Opinion Paper e Experience Paper</i>	QP1c
7	Domínio de aplicação	Domínio de aplicação ao qual o estudo se aplicou	QP1d
8	Fase do ciclo de vida da ES	Levantamento de requisitos, modelagem/design, implementação e teste	QP1e
9	Tipo de contribuição	Tipo de contribuição que o estudo forneceu	QP1f
10	Tipo do autor	Indústria, academia ou ambos	QP1g
11	Foro	O nome do foro de publicação do estudo	QP1h
12	Tipo de publicação	Conferência e workshops, periódicos e relatórios técnicos	PQ1h

As técnicas e estratégias são usualmente identificadas nos estudos por meio das seguintes formas:

- Direta: a proposta é apresentada de forma clara no resumo e/ou introdução do trabalho por meio de frases como “*we introduce a software architecture framework*” (NITTO *et al.*, 2016), “*to develop and validate a new [development] method*” (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015) e “*our solution was to create a [tool]*” (GORTON; KLEIN, 2015);
- Indireta: identificada em frases como *the development of a tool to support data scalability* (abordagem se refere a um desenvolvimento de sistema) ou *process to address requirements retrieval for data intensive systems* (abordagem se refere a proposta de metodologia de desenvolvimento).

Foram listadas todas as estratégias e técnicas descritas na literatura com o objetivo de gerenciar cada tipo empregada na construção de sistemas big data. Além das estratégias e técnicas propostas e suas definições, foi investigado também o tipo de estudo empírico empregado nas mesmas. Ao fim deste processo, a informação sobre cada estratégia e/ou

técnica foi consolidada. As técnicas e estratégias, sob um ponto de vista técnico, podem divergir em aplicabilidade de domínio e característica. Entretanto, ao apresentar a mesma natureza de proposta, podem ser categorizadas em uma mesma técnica ou estratégia.

Objetivo do estudo

Nessa categoria, a informação sobre o objetivo primário do estudo é representada. A técnica e/ou estratégia empregada no estudo inexistem sem um objetivo. Isto é, uma estratégia proposta deve ter a finalidade de cobrir uma lacuna, solucionar um problema. Ao lado das técnicas e estratégias, a identificação dos objetivos dos estudos se tornam particularmente importantes para mapear as lacunas que os recentes estudos querem abarcar, que tipos de problemas estão sendo endereçados e quais são os desafios (técnicos e de negócio) na construção de aplicações big data.

Para a identificação dos diferentes objetivos nos estudos primários, foi observada a apresentação da técnica, que, naturalmente estão relacionados no texto a introdução do objetivo do estudo. Para ser considerado um objetivo de estudo, foi adotado o critério de que o objetivo deve ser a força motriz, a origem, o motivo do desenvolvimento da técnica, estratégia proposta. A terminologia utilizada para classificação foi extraída diretamente dos estudos primários.

Dada a natureza de aplicações big data, envolvendo mais entidades e dados, sugerindo uma engenharia de requisitos distinta no que concerne o gerenciamento de dados e modelagem do design (CHEN *et al.*, 2015) e dados não estruturados onde se requer análise em tempo real (CHUA *et al.*, 2014), os objetivos esperados são:

- Suportar o processamento intenso de dados
- Endereçar requisitos voltados para aplicações big data
- Oferecer diretrizes para design de sistemas big data
- Prover oportunidades de pesquisa
- Fornecer ferramenta de apoio ao desenvolvimento

Tipo de estudo empírico

A adoção de práticas na engenharia de software tem como recomendação considerar a presença de evidência científica como fator de análise da eficácia do método proposto (DYBÅ; KITCHENHAM; JØRGENSEN, 2005). Com a alavancagem da adoção contínua do uso de grandes massas de dados por organizações e centros de pesquisa, é crucial que, ao propor uma estratégia e/ou técnica para construção de sistemas com uso intensivo de dados,

para que esta possa ter seu nível de viabilidade medido, o estudo deve apresentar riscos associados, contexto de aplicabilidade (fase do ciclo de vida, modelo de prototipação, arquitetura aplicável, natureza da persistência de dados), limitações e custos. Dessa forma, profissionais da indústria e pesquisadores que levem em consideração a base em evidências, visando suportar a tomada de decisões nas escolhas de tecnologia, podem encontrar na abordagem proposta insumo adequado para adoção de novas práticas e/ou ferramentas.

Nesse contexto, há diferentes tipos de estudos empíricos que podem ser aplicados para avaliar a efetividade de uma abordagem proposta. Cada tipo de estudo promove um benefício e sua aplicação segue um objetivo específico. Assim, nesta categoria explicito as formas de se coletar informações sobre estudos empíricos na literatura. Para classificar os estudos com base na natureza de avaliação, foram utilizadas as seguintes taxonomias, com base no trabalho de Höst e Runeson (2009):

- *Case study*: processo de pesquisa de natureza flexível onde, dado um ou mais objetivos específicos, busca-se a coleta de dados atendendo a um padrão. Desta coleta de dados, empreende-se a coleta de evidências e posterior análise (essas duas etapas podem coexistir, não sendo exclusivas entre si) para reporte.
- *Survey*: De acordo com Robson (2002), *survey* se refere a um conjunto de coeso (padronizado) de informações coletadas de uma população específica ou amostra. Apesar de comum, essa coleta não necessariamente se dá pela aplicação de questionários ou entrevistas. Os resultados são então analisados e servem de insumo para conclusões descritivas ou exploratórias (WOHLIN et al., 2012)
- *Experiment*: um estudo controlado, pesquisa que mede os efeitos da manipulação de um ou mais variáveis independentes (conhecidas como fatores).

Assim como um *case study*, divide-se em etapas, não sendo um processo rigidamente iterativo (retorno a etapas para refinamento antes de continuar com o experimento são válidos). Um experimento é dividido nas seguintes etapas:

- *Scoping*: Nesta primeira etapa, o escopo, objetivo e metas do experimento são definidos
- *Planejamento*: etapa responsável pelo design do experimento, onde ameaças são avaliadas e a instrumentação definida
- *Operação*: etapa onde as medições são coletadas
- *Análise e interpretação*: medições da etapa anterior são analisadas e avaliadas
- *Apresentação*: etapa onde os resultados são apresentados

Tipo de pesquisa

Baseado na proposta de classificações de estudos por Wieringa *et al.* (2006), este trabalho inclui a categoria tipo de pesquisa com o objetivo de identificar as classes de cada estudo primário e, dessa forma, suportar o processo de decisão do critério de avaliação. As classes de estudo propostas originalmente são direcionadas à engenharia de requisitos. Desta forma, neste estudo flexibilizamos a classes para que as mesmas estejam de acordo com os objetivos deste estudo, isto é, endereçar técnicas e/ou estratégias para construção de sistemas big data.

As classes de estudo utilizadas nesta categoria são:

- *Evaluation research*: relacionado a investigação ou implementação de técnicas e/ou estratégias na prática. Os critérios de avaliação para essa classe são definidos se as propriedades lógicas do problema estão estabelecidas e se a afirmação do estudo está validada.
- *Proposal of solution*: compreende a proposta de uma solução e sua relevância, entretanto sem uma validação criteriosa (apesar de poder oferecer uma prova de conceito como exemplo). Critérios de avaliação para essa classe de estudo são definidos em dois aspectos: se o problema a ser solucionado pela técnica proposta é devidamente explicado e o quão disruptiva é a técnica ou a aplicação da técnica para uma natureza de problema.
- *Validation research*: pesquisa que investiga as propriedades de uma proposta de solução ainda não implementada na prática (WIERINGA *et al.*, 2006). Isto é, uma abordagem proposta é investigada, mas sua implementação em cenário real ainda não foi realizada.
- *Philosophical paper*: estudos que desenham uma ideia não antes endereçada, um novo framework conceitual, segundo Wiering *et al.* (2006). O critério de avaliação é o quão original é a ideia.
- *Opinion paper*: contêm a opinião do autor sobre a qualidade de um determinado tema e/ ou forma de se empreender um esforço sobre esse tema. Os critérios de avaliação são formados por quão a posição emitida é coesa e quão discussão a mesma pode provocar no ambiente inserido.
- *Experience paper*: expressa a experiência do autor em um ou mais projetos contendo suas lições aprendidas. A experiência reportada pode ter como origem a indústria ou academia, onde as ferramentas (técnica e/ou estratégia) utilizadas são colocadas em

prática. O critério de avaliação se pelo nível de relevância para profissionais da indústria e academia.

Domínio de aplicação

Os domínios de aplicação têm fundamental importância na construção de um software. Segundo Evans (2004), podem influenciar em constantes como risco e qualidade dependendo da complexidade relacionada ao design da solução bem como na arquitetura da aplicação (uma vez que se caracterizam pela aplicação de frameworks, bibliotecas, classes de negócio). As necessidades complexas de um software, dado o seu domínio, é objeto de interesse há alguns anos na Engenharia de Software. Evans (2004) retratou em seu livro, *Domain-Driven Design*, uma abordagem para construção de software que tem como núcleo o domínio da aplicação. Segundo Zimmerman *et al.* (2013), aplicações big data abrangem domínios como:

- (i) intelligent mobility systems and services; (ii) intelligent energy support systems; (iii) smart personal healthcare systems and services; (iv) intelligent transportation and logistics services; (v) smart environmental systems and services; (vi) intelligent systems and software engineering; (vii) intelligent engineering and manufacturing.

Entretanto, a classificação de estudos segundo seu domínio de aplicação se torna particularmente importante para endereçar o foco de estudos de aplicações big data.

Esta categoria representa os domínios de aplicação que o desenvolvimento de sistemas big data podem abarcar. Todos os domínios mencionados na literatura foram listados com o objetivo de identificar domínios aplicáveis e domínios ainda não endereçados pelos estudos. Por último, apenas domínios de aplicação onde há o emprego de um estudo empírico (isto é, *case study*, *survey* ou *experiment*) são considerados nessa categoria, ou seja, apenas mencionar ou citar a aplicabilidade em um dado domínio não se aplica, uma vez que não há validade formal da proposta de estudo.

Fase do ciclo de vida do desenvolvimento de software

A Engenharia de Software definiu ao longo dos anos, por meio de experiências adquiridas no projeto e prototipação de software, diferentes metodologias. Cada uma delas foi sendo proposta de acordo com a mudança do contexto e característica da construção de software, com o objetivo de atuar nas constantes base de um projeto, qualidade, custo e tempo. O modelo cascata, com fases bem definidas e como característica a inflexibilidade existente entre fases (não sendo permitindo retornar a uma fase anterior), não prevendo mudança no escopo do projeto após as fases iniciais, falhava ao evoluir o software de acordo

com novas necessidades levantadas por *stakeholders* ao longo do desenvolvimento. Por fim, o manifesto ágil, visando abarcar essa lacuna existente, e baseado na experiência adquirida de desenvolvedores na indústria ao longo dos anos, definiu um conjunto de diretrizes cunhado como manifesto ágil, que prevê principalmente a alta interação com *stakeholders*, em especial os especialistas do domínio e usuários diretamente ligados ao uso do software. Tendo como base o manifesto, diversos outros modelos foram propostos sempre com o objetivo de endereçar o compartilhamento de informações ao longo do projeto, independentemente de fase do ciclo de vida, como o SCRUM.

As fases do ciclo de vida de um software, independentemente da abordagem (ágil ou não ágil), preveem fases correlatas, como levantamento de requisitos e construção. Desta forma, com o objetivo de cobrir um conjunto de etapas presentes majoritariamente nas metodologias de desenvolvimento, foram definidas quatro fases, com suas respectivas características, para o contexto da categorização deste trabalho. Isto tem por objetivo prover uma linguagem comum (nível uniforme) independente de modelo de ciclo de vida. As fases propostas podem ser acessadas abaixo:

- Levantamento de requisitos: Onde se empreende a coleta de informações sobre o problema a ser solucionado e a determinação dos requisitos do software;
- Modelagem: Fase em que se desenha o sistema em alto nível, sem um aprofundamento técnico, detalhado. Por exemplo, diagramas técnicos como os definidos pela UML podem ser desenhados nesta etapa;
- Design: Fase que sucede a modelagem, o desenho do sistema tem como característica o baixo nível, isto é, com detalhamento técnico aprimorado. Uma prototipação como prova de conceito pode ser desenvolvida nesta etapa;
- Construção/Implementação: Compreende o desenvolvimento de artefatos de software, codificação e configurações de suporte para o funcionamento do software, como a definição do banco de dados;
- Teste: Fase responsável por verificar se o software atende os requisitos funcionais e não funcionais. Esta pode ser executada por uma série de processos e abarcar diferentes objetivos, como validação de performance e testes funcionais, aqueles que devem atender a casos de uso;
- *Deploy*: Compreende as atividades necessárias para colocar um software disponível para utilização por usuários;

- Manutenção: Fase responsável por suportar a operação do software, envolvendo esforços de manutenção corretiva e evolutiva.

Nesse contexto, esta categoria tem a intenção de identificar a que fases do ciclo de vida as abordagens propostas têm aplicabilidade.

Tipo de contribuição

Baseado na classificação de O'Donovan *et al.* (2015), este trabalho empreendeu uma classificação de tipos de contribuição pelo emprego do método *keywording* em conjunto com a identificação da abordagem e objetivo do estudo. Isto é, as descrições de cada dimensão do trabalho de O'Donovan *et al.* (2015) são reutilizadas, dessa forma, os estudos primários são classificados segundo as classificações propostas e a identificação de palavras-chave. Isto tem por objetivo alavancar uma classificação comum nos mapeamentos e estudos primários desta área de estudo, promovendo assim maior maturidade, e prover maiores informações sobre cada estudo primário. As palavras-chave foram identificadas não apenas pelo resumo do estudo, mas também pela conclusão. Os tipos de contribuição, podem ser visualizados na Tabela 11 abaixo.

Tabela 11 - Tipos de contribuição de pesquisa

Classificação	Descrição
Arquitetura	Estudo que descreve uma visão teórica e/ou de implementação de escolhas arquiteturais de um software
Framework	Estudo que descreve a proposta e/ou desenvolvimento de bibliotecas de software para resolução de um problema
Teoria	Estudo que provê diretrizes para solucionar determinado problema
Metodologia	Estudo que apresenta abordagens e/ou métodos para a resolução de um problema
Modelo	Estudo que propõe um modelo matemático para resolução de um problema
Plataforma	Estudo que provê um sistema para suporte a execução de aplicações
Processo	Estudo que apresenta processos para a resolução de um problema
Ferramenta	Estudo que visa o desenvolvimento de utilitário de software para endereçar um problema

Tipo do autor

A Engenharia de Software é conhecida por ser uma área onde muitos estudos apresentam uma grande interseção entre a academia e a indústria. Um exemplo são as pesquisas sobre débito técnico, termo que descreve o efeito de artefatos de software (uma biblioteca, uso de framework, código) imaturos introduzidos na manutenção de software com vista a prover produtividade com uma rápida entrega, benefício a curto prazo, entretanto, tendo como possíveis malefícios a adoção de um design pobre, inadequado para mudanças futuras.

Nesta categoria, a informação sobre o tipo do autor que empreendeu o estudo é representada. Basicamente, seus atributos são: academia, indústria e academia/indústria. Estudos podem ser realizados em conjunto por pesquisadores e profissionais da indústria ou realizados num ambiente individual sem a parceira.

Nesse contexto, o objetivo dessa categoria é avaliar o grau de interseção entre as pesquisas atualmente em curso e as necessidades da indústria para a pesquisa em construção de software big data e assim identificar se os estudos identificados contemplam desafios reais das organizações.

Foro de publicação

O objetivo desta categoria é representar a informação sobre os foros de publicação dos estudos primário identificados neste trabalho. Cada estudo pode ser publicado em uma conferência ou periódico, entretanto, é importante identificar quais são os foros mais proeminentes na área e se há estudos científicos que são apresentados de outra maneira, como em relatórios técnicos.

Metadados

Além das categorias explicitadas acima, para suportar a análises sobre tendências de publicação e maturidade da área de estudo, os seguintes dados também foram extraídos dos estudos primários: ano de publicação do estudo e biblioteca digital. Estes são conhecidos como metadados dos estudos e suas dimensões suportam a análise dos resultados de pesquisa.

3.6 CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentado o protocolo utilizado para realizar o mapeamento sistemático sobre desenvolvimento de sistemas big data, com a string de busca definida e as

técnicas empregadas para obtenção dos estudos primários, como *snowballing*. Em adição, foram apresentadas as categorias de pesquisa uma a uma e sua contextualização, isto é, sua definição e, quando possível, suas categorizações (estudo empírico, tipo de pesquisa, fase do ciclo de vida e tipo de contribuição). A descrição detalhada dos resultados do mapeamento será comentada no próximo capítulo.

CAPÍTULO 4 – RESULTADOS OBTIDOS

4.1 INTRODUÇÃO

Neste capítulo, baseado nas questões de pesquisa explicitadas no Capítulo 2 e nas categorias de pesquisa endereçadas no Capítulo 3, vamos responder a cada questão levantada por esse estudo, caracterizando cada categoria extraída e se apoiando em gráficos e tabelas para melhor compreensão dos estudos primários na área. A Seção 4.2 apresenta, para cada categoria extraída dos estudos primários (ver Tabela 10), os resultados de pesquisa, apontando tendências históricas, quando necessário. Em seguida, a Seção 4.3 fornece uma síntese dos resultados, expondo alguns pontos positivos e negativos encontrados nos resultados. Por fim, a Seção 4.4 expõe as considerações finais do capítulo.

4.2 RESPOSTAS ÀS QUESTÕES DE PESQUISA

Esta seção apresenta a análise dos dados extraídos dos estudos primários selecionados. Primeiramente é apresentado a análise dos metadados relacionados às publicações. Em seguida, são apresentadas as subseções de abordagens da ES para construção de sistemas big data, seus objetivos, domínios de aplicação, tipos de contribuição, fases do ciclo de vida, estudos empíricos identificados e tipos de pesquisa. Para a extração dados, os estudos primários selecionados foram lidos em sua completude. Uma planilha foi utilizada para a coleta e análise dos metadados. Esta pode ser acessada através de um link apresentado no Apêndice D.

Coding

Para duas categorias deste trabalho, abordagem e objetivo, foi aplicado um processo conhecido como *coding*. Este processo foi aplicado para identificação da categoria no estudo primário e, dessa forma, sua descrição era associada a uma *tag* que representasse bem seu conteúdo. A técnica de *coding* é muito utilizada em estudos qualitativos envolvendo *grounded theory*, em que não se sabe as categorias a priori. No contexto desse trabalho, a técnica de coding se apresentou como adequada pela pouca maturidade da área e pela necessidade de identificar classes de categorias.

Segundo Fitzgerald *et al.* (2016), o objetivo de *grounded theory* é gerar teoria em lugar de testar ou validar uma teoria existente. Em outras palavras, por meio de informações extraídas, busca construir uma teoria. Para tal, se apoia em textos não estruturados, textos estruturados, diagramas, imagens e dados quantitativos (GLASER, 1992). No contexto deste

trabalho, textos estruturados, muitas vezes baseados na parte de resumo e conclusão do trabalho, foram utilizados. Em adição, apenas um componente específico de *grounded theory* foi aplicado a esse estudo: *coding*.

Coding pode ser exemplificado quando o pesquisador infere categorias teóricas de um conjunto de dados pela rotulagem de incidentes e suas propriedades (FITZGERALD *et al.*, 2016). No nosso caso, a cada conjunto de palavras identificada que fortemente representava o conteúdo do trabalho, um rótulo era marcado no texto. Ao final, foi possível identificar tendências e estas se tornaram insumos para as algumas categorias a seguir.

Abordagem da Engenharia de Software (PQ1)

QP1: Que abordagens da Engenharia de Software tem sido propostas para o suporte e construção de sistemas big data?

Para a definição das abordagens, que são entendidas como técnicas e estratégias para a construção de sistemas big data (exposto na seção de metadados dos estudos), a técnica de *coding* foi aplicada. Assim, as abordagens encontradas neste estudo são apresentadas ordenadas por frequência e com sua respectiva definição:

- **Desenvolvimento de sistema:** Se refere a sistemas de software construídos a fim de atender um objetivo específico, que varia entre cobrir uma lacuna na área de estudo ou domínio de aplicação e aprimorar processos comuns no processo de desenvolvimento de software, como o design e *deploy* de aplicações;
- **Proposta de metodologia de desenvolvimento:** Se refere a introdução de mudanças ou adaptações em uma metodologia de desenvolvimento pré-existente ou a proposta de um modelo de ciclo de vida com o objetivo de atender características de projetos de sistemas big data, como dados não estruturados, análise em tempo real, e foco na captura, armazenamento, processamento e análise de dados (CHUA *et al.*, 2014);
- **Proposta de arquitetura de software:** Se refere a propostas de definição de arquitetura ou metodologia empregada para definição da arquitetura voltadas a atender requisitos de aplicação big data. Restrições que se buscam atender são aquisição de dados, processamento de dados em larga escala e integração entre diferentes modelos de dados e suas tecnologias;
- **Proposta de método para design de arquitetura de sistemas:** Se refere a propostas de soluções relacionadas ao design da arquitetura do software;
- **Proposta de método para design de sistemas:** Se refere a propostas de soluções relacionadas ao design da aplicação;

- **Proposta de método para modelagem de sistemas:** Se refere a propostas de solução relacionadas à modelagem de dados da aplicação;
- **Compartilhamento de experiência:** Se refere ao reporte sobre a experiência adquirida no contexto da construção de sistemas com uso intensivo de dados. Decisões tomadas ao longo do projeto, como arquiteturas, de tecnologia e de modelo de dados são esperadas nessa abordagem;
- **Proposta de abordagem para engenharia de requisitos:** Se refere a estratégias desenhadas para endereçar requisitos em aplicações big data. Espera-se nessa abordagem a introdução de modelos metodológicas para tratar características de sistemas big data, como o volume de dados e as decisões de design voltadas a atendê-los (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015);
- **Proposta de extensão de linguagem de modelagem:** Se refere a adição de recursos visuais a uma linguagem de modelagem com o objetivo de aumentar o nível de detalhes a serem capturados na notação. A notação UML, utilizada para modelagem e design de sistemas se inclui como um exemplo de linguagem de modelagem;
- **Proposta de modelo de análise de performance:** Se refere a modelos matemáticos e/ou estatísticos propostos para identificar medidas de performance em aplicações big data;
- **Maapeamento de problemas:** Se refere à pesquisa de problemas latentes na indústria de software ou academia, relacionada à gerência de grande volume de dados em aplicações big data ou gerados pelos mesmos, como logs e dados estatísticos;
- **Desenvolvimento de ferramenta de apoio à IDE:** Se refere a plugins desenvolvidos e integrados à IDEs que tem por objetivo suportar o desenvolvimento de artefatos de software em um projeto big data;
- **Monitoramento de time de desenvolvimento:** Se refere ao acompanhamento de desenvolvedores atuando em conjunto em projetos de sistemas big data. O objetivo nessa abordagem é colher informações sobre as práticas adotadas;
- **Proposta de método para migração de dados:** Se refere à elaboração de técnicas e diretrizes arquiteturas para habilitar a migração de dados entre fontes de dados de diferente natureza (paradigma relacional e não-relacional, por exemplo) no contexto de uma aplicação big data.

A Figura 4 ilustra a distribuição de estudos por abordagem. É possível observar que a maioria dos estudos possuem como abordagem *Desenvolvimento de software*, indicando uma

tendência em desenvolver soluções de software visando endereçar um problema específico. Em seguida, *Propostas de metodologias de desenvolvimento* também se apresentam como foco de abordagem. Além disso, há uma alta concentração de estudos na abordagem *Proposta de Arquitetura de Software* e *Proposta de método para design de arquitetura de sistemas*. Isso pode se dar ao fato de aplicações com uso intensivo de dados necessitarem de arranjos arquiteturais não convencionais, fomentando assim a pesquisa nessa abordagem. Por último, é importante salientar que abordagens relacionadas a aspectos não puramente técnicos na construção de sistemas big data, como *Monitoramento de time de desenvolvimento*, *Mapeamento de problemas na indústria* e *Compartilhamento de experiência*, tem sido preteridas. Como um campo novo na área da engenharia de software, é comum abordagens de natureza técnica prevalecerem neste início.

Como complemento, a Tabela 12 exibe o número de estudos por abordagens identificadas neste mapeamento distribuídas por ano. É observado que, entre os anos 2015 e 2016, os estudos se concentraram na abordagem de “*Desenvolvimento de Sistema*” para solucionar problemas da área de estudo.

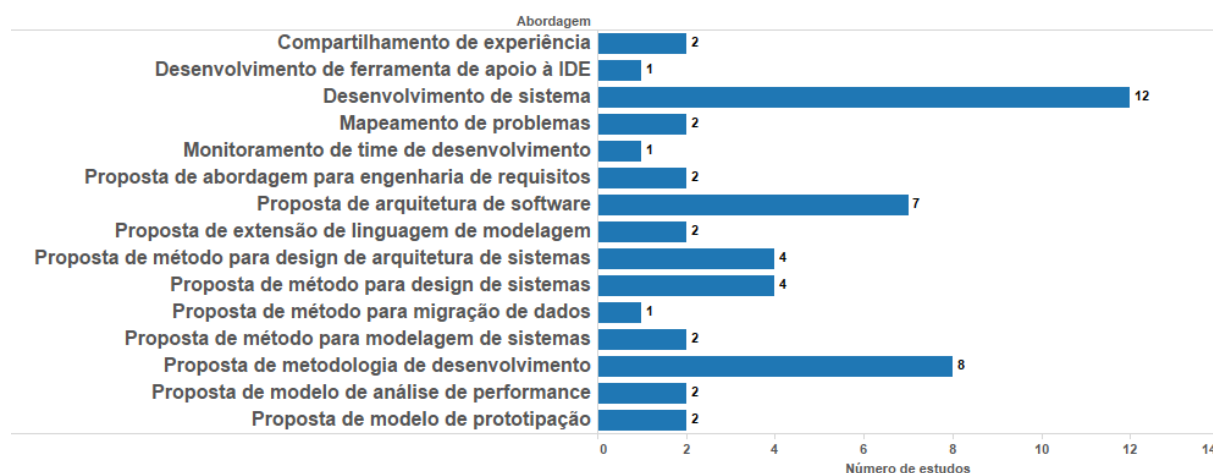


Figura 4 – Número de estudos por abordagem

Ademais, abordagens relacionadas a metodologias de desenvolvimento para aplicações big data, como adaptações e propostas, se mostram constantes desde 2014, tendo a última apresentado 3 estudos em 2016. Isso é um indicativo de que pesquisadores tem se atentado a identificar barreiras e oportunidades de melhoria nas metodologias de desenvolvimento existentes, visando abarcar características de aplicações big data.

Objetivo do estudo (QP1c)

RQ1c: Quais os objetivos de estudo das abordagens propostas para o suporte e construção de sistemas big data?

Nessa categoria, da mesma forma que na categoria *Abordagem da Engenharia de Software*, a técnica de *coding* foi aplicada. Os objetivos de estudo das abordagens encontradas neste mapeamento também são apresentados ordenados por frequência.

- **Prover uma metodologia para o desenvolvimento:** Se refere ao objetivo de endereçar efetivamente especificidades encontradas no ciclo de vida do desenvolvimento de sistemas big data, como o foco nos dados e suas características (volume, complexidade e processamento) pela aplicação;
- **Prover uma solução para o design:** Se refere ao objetivo de solucionar uma lacuna ou problema existente no design de aplicações big data, endereçando por exemplo, a estrutura da aplicação, modelo de dados adotado e tecnologias;
- **Prover uma solução para o processamento de grandes conjuntos de dados:** Se refere a abordagens propostas voltadas a endereçar o processamento de um largo volume de dados em aplicações big data e requisitos de performance associados;
- **Evoluir o corpo de conhecimento da arquitetura de software:** Se refere ao objetivo de estruturar, introduzir conhecimento na área da Engenharia de Software obtido por meio de pesquisa, experiência ou experimentação;
- **Prover uma solução para o design da arquitetura:** Se refere ao objetivo de suportar e/ou prover informações adequadas para a especificação e design da arquitetura de aplicações big data, endereçando por exemplo o modelo apropriado dado os requisitos da solução;
- **Prover uma solução para suporte à execução e/ou construção de aplicações:** Se refere a soluções propostas voltadas a suportar a construção de aplicações big data, seja provendo bibliotecas de software, seja oferecendo uma plataforma de execução;
- **Prover uma solução para endereçar requisitos:** Se refere ao objetivo de atender características chave na engenharia de requisitos para aplicações big data, como o foco em custo e atributos de qualidade (por exemplo, escalabilidade e disponibilidade);
- **Identificar lacunas e/ou oportunidades:** Se refere ao objetivo de identificar no campo de estudo lacunas em projetos de aplicação big data como falta de adaptabilidade da metodologia de desenvolvimento escolhida. Em adição, concerne também em argumentar sobre oportunidades de melhoria no processo de construção de software big data, como falta de ferramentas de suporte ao desenvolvimento;

Tabela 12 – Número de estudos por abordagem ano a ano

Abordagem	2011	2012	2013	2014	2015	2016	Total
Compartilhamento de experiência		1			1		2
Desenvolvimento de ferramenta de apoio à IDE					1		1
Desenvolvimento de sistema			1	3	3	5	12
Mapeamento de problemas						2	1
Monitoramento de time de desenvolvimento					1		1
Proposta de abordagem para engenharia de requisitos				1		1	2
Proposta de arquitetura de software	1		1	1	2	2	7
Proposta de extensão de linguagem de modelagem			1			1	2
Proposta de método para design de arquitetura de sistemas			1		3		4
Proposta de método para design de sistemas				1	3		4
Proposta de método para migração de dados						1	1
Proposta de método para modelagem de sistemas				1		1	2
Proposta de metodologia de desenvolvimento				3	1	4	8
Proposta de modelo de análise de performance				1	1		2
Proposta de modelo de prototipação					1	1	2
Total	1	1	4	11	17	18	52

- **Prover uma solução para a modelagem:** Se refere ao objetivo de solucionar uma lacuna ou problema existente na modelagem de aplicações big data, endereçando por exemplo, o processo da modelagem de classes da aplicação;
- **Prover solução para logging de dados:** Se refere ao objetivo de estruturar uma solução para realizar o armazenamento de informações sobre performance e erros de execução de uma aplicação com uso intensivo de dados;
- **Prover uma solução para o deployment:** Se refere ao objetivo de fornecer os artefatos necessários para dar suporte ao processo de deploy de uma aplicação big data;

A Figura 5 exibe o número de estudos por objetivo de estudo. Dentre os estudos primários selecionados, é possível identificar uma alta concentração de estudos nos seguintes objetivos: *Prover uma solução para o design*, *Prover uma metodologia para o desenvolvimento* e *Prover uma solução para o processamento de grandes conjuntos de dados*. Dessa forma, os

resultados evidenciam novamente a tendência na busca por melhores processos no desenvolvimento de sistemas big data.

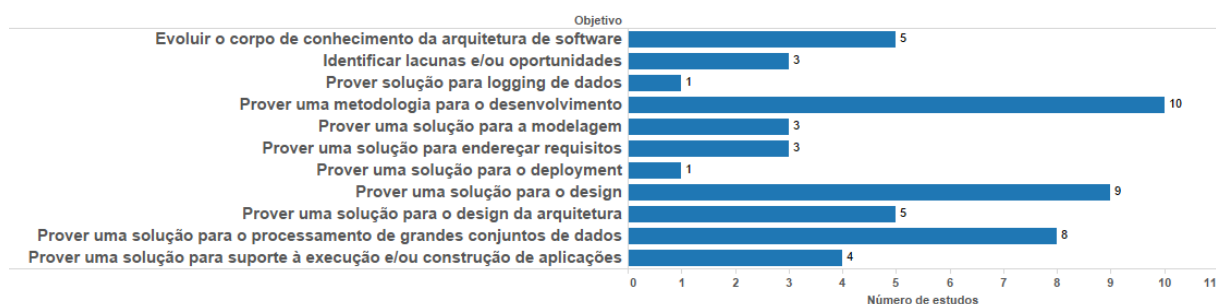


Figura 5 – Número de estudos por objetivo de estudo

A distribuição de objetivos por ano de publicação pode ser acessada na Tabela 13. É importante notar que entre 2011 e 2014, menos da metade dos objetivos mapeados neste trabalho haviam sido endereçados até então, o que explicita a fase inicial desta área de estudo. Entretanto, os objetivos foram se expandindo com o tempo, indicando que novas necessidades foram surgindo e lacunas sendo identificadas, como metodologias para desenvolvimento e design de aplicações big data.

Tabela 13 - Número de estudos por objetivo ano a ano

Objetivo	2011	2012	2013	2014	2015	2016	Total
Prover uma solução para o processamento de grandes conjuntos de dados	1		1	2	2	2	8
Prover uma solução para endereçar requisitos			1	1		1	3
Prover uma solução para o design da arquitetura			1		2	2	5
Prover uma solução para o design				1	5	3	9
Prover uma metodologia para o desenvolvimento				3	2	5	10
Evoluir o corpo de conhecimento da arquitetura de software		1	1	1	1	1	5
Identificar lacunas e/ou oportunidades					2	1	3
Prover uma solução para a modelagem				1	1	1	3
Prover uma solução para o deployment				1			1
Prover solução para logging de dados						1	1
Prover uma solução para suporte à execução e/ou construção de aplicações				1	2	1	4
Total	1	1	4	11	17	18	52

Além disso, é possível identificar objetivos que, apesar de representarem uma pequena parcela da amostra de resultados, se mostram constantes ano a ano, como *Prover uma solução para o processamento de grandes conjuntos de dados* e *Evoluir o corpo de conhecimento da arquitetura de software*.

A Tabela 14 apresenta o número de estudos dividido por cada objetivo de estudo e suas respectivas abordagens mapeadas. Em conjunto com as informações providas pelas Tabelas 12 e 13, é possível observar que, à medida que novas abordagens são propostas, mais objetivos de estudo são identificados. A identificação do objetivo de estudo se torna particularmente importante no caso de abordagens alto nível, isto é, aquelas que a classificação não caracteriza as especificidades da abordagem, suas características técnicas, como *Proposta de Arquitetura de Software* e *Desenvolvimento de Sistema*. Entretanto, uma vez que o objetivo do estudo é delineado, é possível determinar de forma clara o que a solução arquitetural ou o software desenvolvido visa endereçar. Para mapear os objetivos ligados a cada abordagem, foi utilizada informação extraída de cada estudo relacionando as categorias. Ou seja, os objetivos identificados são explicitamente mencionados nos estudos, como por exemplo: “*we highlight the lack of developer support tools for data-intensive systems, the importance of multidisciplinary teams*” (ANDERSON, 2015). Nesse contexto, o autor explicita que há uma lacuna no que concerne ferramentas de apoio ao desenvolvimento e argumenta sobre a necessidade de equipes multidisciplinares, isto é, profissionais com diferentes conhecimentos atuando em conjunto. Dessa forma, o objetivo do trabalho de Anderson (2015) foi classificado como *Identificar lacunas e/ou oportunidades*. Além disso, o que Anderson (2015) expõe vai de encontro ao explicitado por Rosenthal *et al.* (2015), onde desenvolvedores tecnicamente habilitados na escrita de código, se mostram ineficientes ao construir soluções de software *analytics*, uma vez que o conceito e o processo de *data analytics* não são conhecidos. Além disso, é possível observar que alguns objetivos, como *Evoluir o corpo de conhecimento da arquitetura de software* e *Prover uma solução para o design*, possuem um considerável número de abordagens diferentes. Por outro lado, há abordagens que endereçam poucos objetivos diferentes, como *Proposta de metodologia de desenvolvimento* e *Proposta de método para migração de dados*. Em adição, há abordagens que aparecem em apenas um objetivo, como *Mapeamento de problemas na indústria* (objetivo *Prover solução para logging de dados*), *Monitoramento de time de desenvolvimento* (objetivo *Identificar lacunas e/ou oportunidades*) e *Desenvolvimento de ferramenta de apoio à IDE* (objetivo *Prover solução para o design*). Em seguida, é possível identificar também que as abordagens *Proposta de arquitetura de software* e *Proposta de método para design de*

Tabela 14 – Distribuição de abordagens por objetivo

Objetivo	Abordagem	Contagem de estudos
Evoluir o corpo de conhecimento da arquitetura de software	Compartilhamento de experiência	1
	Proposta de abordagem para engenharia de requisitos	1
	Proposta de método para design de arquitetura de sistemas	2
	Proposta de modelo de análise de performance	1
Identificar lacunas e/ou oportunidades	Compartilhamento de experiência	1
	Mapeamento de problemas	1
	Monitoramento de time de desenvolvimento	1
Prover solução para logging de dados	Mapeamento de problemas	1
Prover uma metodologia para o desenvolvimento	Proposta de metodologia de desenvolvimento	8
	Proposta de modelo de prototipação	2
Prover uma solução para a modelagem	Proposta de método para modelagem de sistemas	2
	Proposta de modelo de análise de performance	1
Prover uma solução para endereçar requisitos	Proposta de abordagem para engenharia de requisitos	1
	Proposta de arquitetura de software	1
	Proposta de extensão de linguagem de modelagem	1
Prover uma solução para o deployment	Desenvolvimento de sistema	1
Prover uma solução para o design	Desenvolvimento de ferramenta de apoio à IDE	1
	Desenvolvimento de sistema	2
	Proposta de extensão de linguagem de modelagem	1
	Proposta de método para design de arquitetura de sistemas	1
	Proposta de método para design de sistemas	4
Prover uma solução para o design da arquitetura	Desenvolvimento de sistema	1
	Proposta de arquitetura de software	2
	Proposta de método para design de arquitetura de sistemas	1
	Proposta de método para migração de dados	1
Prover uma solução para o processamento de grandes conjuntos de dados	Desenvolvimento de sistema	5
	Proposta de arquitetura de software	3
Prover uma solução para suporte à execução e/ou construção de aplicações	Desenvolvimento de sistema	3
	Proposta de arquitetura de software	1

sistemas possuem muitos objetivos correlatos, indicando um grau alto de relação entre as abordagens; é relevante citar que na construção de sistemas a definição da arquitetura da solução antecede a fase de design, o que pode explicar a relação identificada entre as propostas.

Em adição, um dos pontos notáveis desta análise é que alguns objetivos se apresentam em quase sua totalidade em abordagens específicas. Por exemplo, *Prover uma solução para o processamento de grandes conjuntos de dados* é citado majoritariamente como objetivo na abordagem *Desenvolvimento de sistema*. Dessa informação é possível extrair duas conjunturas. A primeira se refere a tendência de, quando no objetivo de obter soluções voltadas a lidar com volume massivo de dados, têm-se optado pelo desenvolvimento de soluções customizadas de software. Em seguida, isso também pode revelar uma tendência em buscar abordagens pré-definidas para resolver problemas específicos. É também possível identificar que algumas categorias em *Abordagem* e *Objetivo* estão relacionadas por conta de características descritivas correlatas, como por exemplo a abordagem *Proposta de Metodologia de Desenvolvimento* e o objetivo *Prover uma metodologia para suporte ao desenvolvimento de aplicações big data*. De fato, alguns objetivos podem estar ligados naturalmente à uma abordagem, mas isso não é verdade em todos os casos. Por exemplo, a abordagem *Proposta de método para design de sistemas* também possui estudos com o objetivo *Prover uma metodologia para o desenvolvimento*, o que explicita um dos dados mais importantes identificados por esse trabalho, isto é, a caracterização dos estudos na área.

Estudo Empírico (PQ1e)

RQ1e: Que avaliações empíricas têm sido utilizadas para a apresentação de propostas de abordagens?

A fim de classificar os tipos de avaliação realizadas, foram consideradas as propostas de estratégias empíricas mencionadas por Wohlin *et al.* (2000): *Case study*, *experiment* e *survey*. A Figura 6 apresenta o número de pesquisas divididos por estudo empírico empregado. É observado que quase metade dos estudos não se utilizou de um estudo empírico para validade da abordagem introduzida. O método mais utilizado foi *case study*, com 20 estudos. Para o método *experiment*, entretanto, apenas 7 estudos o aplicaram. Em adição, é importante notar também que parte significativa dos estudos, 21 de um total de 50, não implementou nenhum método empírico, uma vez que apresentam a abordagem e sua respectiva proposta, mas não apresentam resultados conclusivos de sua validade. Em conclusão, essa conjuntura (alto número de estudos de caso e estudos sem estudo empírico)

caracteriza que muitas propostas na área de construção de sistemas big data ainda necessitam de mais experimentos, afim de que seus resultados sejam validados e sua utilidade medida.

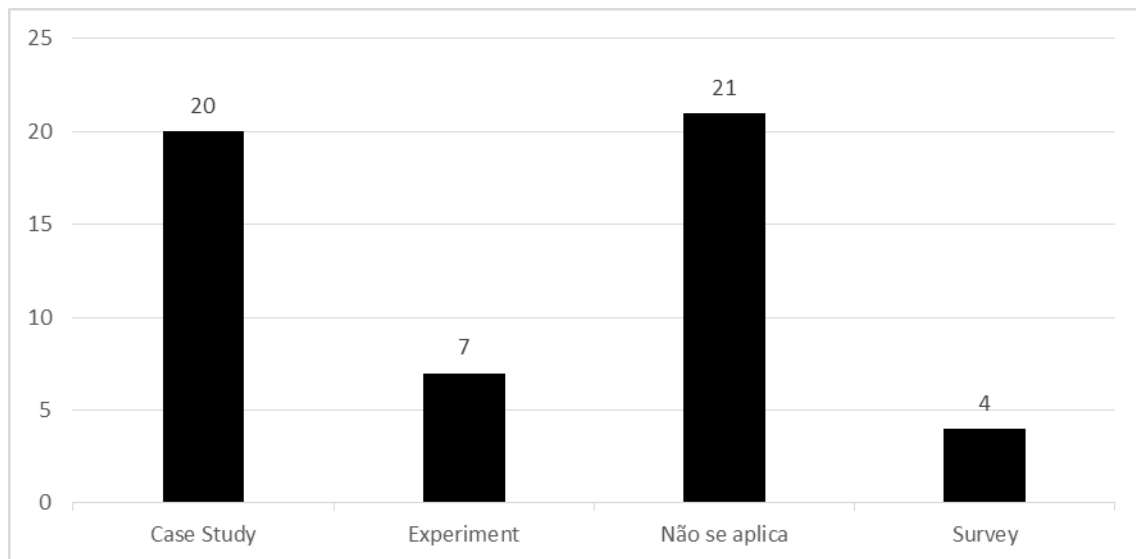


Figura 6 - Distribuição de estudos primários por estudo empírico

Como informação secundária, na Figura 7 é apresentado o número de estudos empíricos por abordagem. Ao todo, 15 abordagens foram caracterizadas neste estudo, entretanto em apenas poucas delas foi empreendido uma avaliação criteriosa da aplicabilidade, ou seja, uma avaliação empírica e posterior evidência da validade. Os resultados exprimem que *case study* e *experiment* são os estudos empíricos mais utilizados entre as abordagens propostas.

A abordagem *Desenvolvimento de sistema*, apesar de se apresentar com o maior número de estudos empíricos, também se mostra como a abordagem com maior número de trabalhos que não empregam um estudo empírico (5). Em seguida, a abordagem *Proposta de Metodologia de desenvolvimento* aparece com o segundo maior número de estudos empíricos. Pela natureza da abordagem, isto é, foco no processo do software, é natural a observação de um elevado número de estudos de caso.

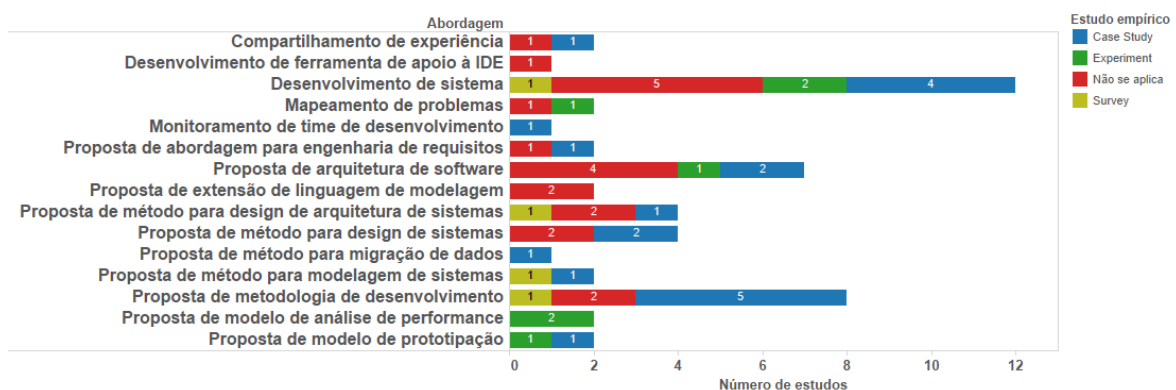


Figura 7 - Distribuição de número de estudo empírico por abordagem

Tipo de pesquisa (PQ1h)

RQ1h: Quais são os tipos de pesquisa dos estudos primários?

A Figura 8 apresenta a divisão de cada tipo de pesquisa por número de estudos primários selecionados. Com 20 estudos, a maioria das pesquisas conduzidas possuem como tipo de pesquisa *Proposal of solution*, onde, dado um determinado problema, apresenta uma amostra de evidência da solução. Em seguida, *Philosophical paper*, tipo de pesquisa onde o objetivo primário não é validar os conceitos apresentados, aparece como o segundo tipo de pesquisa mais empregado, com um total de 14 estudos. Com 9 estudos cada, aparecem *Experience paper*, tipo de pesquisa onde o autor exprime sua experiência em um projeto de pesquisa, e *Evaluation research*, tipo de pesquisa onde se busca a validação da técnica ou estratégia proposta.

Um dos tipos de pesquisa menos comum é *Evaluation research*. Novamente, assim como identificado nos dados extraídos da categorização de estudo empírico, é possível observar uma imaturidade na área. Isso se configura por *Evaluation research*, um tipo de pesquisa avançado, uma vez que a proposta da pesquisa possui validade empírica, possuir um número de estudos menor que *Proposal of solution*, um tipo de pesquisa que muitas vezes não empreende uma validação, e *Philosophical Paper*, onde o foco do estudo se dá em expressar um conceito.

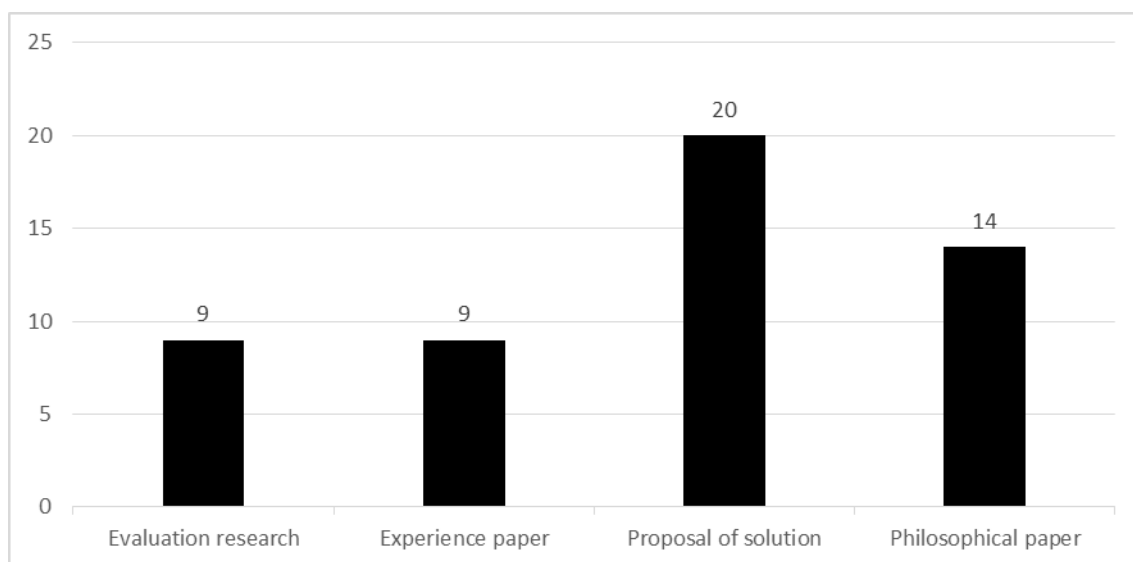


Figura 8 – Número de estudos primários por tipo de pesquisa

Para um maior entendimento dessa classificação, a Figura 9 ilustra a porcentagem de estudos para cada tipo de pesquisa ano a ano. Os primeiros estudos da área se concentraram em *Experience Paper* em 2011 e 2012, até que *Proposal of solution* começou a manter a

liderança a partir de 2013, entretanto, apresenta tendência de menor proporção em relação aos demais tipos desde então.

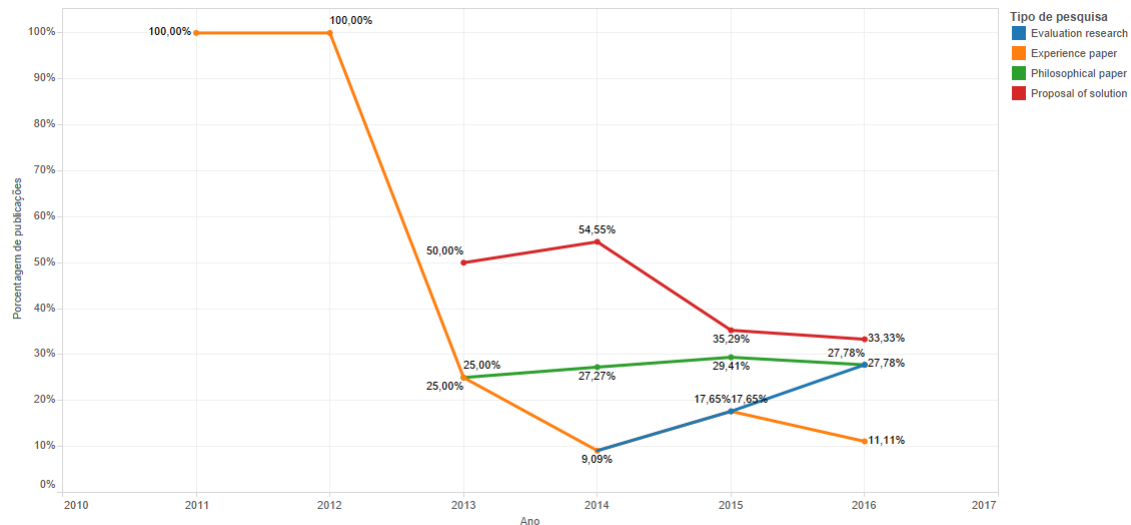


Figura 9 – Porcentagem de estudos por tipo de pesquisa ano a ano

Em seguida, trabalhos com o tipo de estudo *Philosophical paper* tem se mostrado estáveis, abarcando pouco mais de ¼ das pesquisas na área. Outra observação importante é que estudos com o tipo de pesquisa *Evaluation research* apenas se iniciam no ano de 2014 e seu número baixo se configura insuficiente para uma caracterização de maturidade da área, uma vez que comparado com *Proposal of solution*, se apresenta em número inferior. Entretanto, é possível identificar que estudos com tipo de pesquisa *Evaluation research* tem apresentado uma taxa de crescimento considerável desde seus primeiros estudos identificados e, se esta tendência se confirmar, em um curto período de tempo uma teremos uma área com considerável número de estudos amadurecidos.

Domínio de Aplicação (QP1b)

QP1b: Para quais domínios de aplicação as abordagens propostas são aplicadas?

Abordagens podem também ser classificadas pelo domínio de aplicação que se propõem a solucionar um problema. A Figura 10 exibe a distribuição de estudos primários que empregam um estudo empírico por domínio de aplicação. Os domínios foram selecionados em abordagens que empreenderam um estudo empírico, ou seja, estudos que apenas citam um dado domínio como aplicável na construção de sistemas big data não estão incluídos. Isto tem por objetivo endereçar os domínios que efetivamente possuem estudos primários com rigor experimental aplicado. É possível verificar que os domínios em que mais abordagens foram propostas são E-Commerce, Segurança de Rede e Marketing Social. Uma

possível explicação para essa conjuntura é a natureza desses domínios: largo conjunto de dados, alta geração de dados e processamento de dados em larga escala.

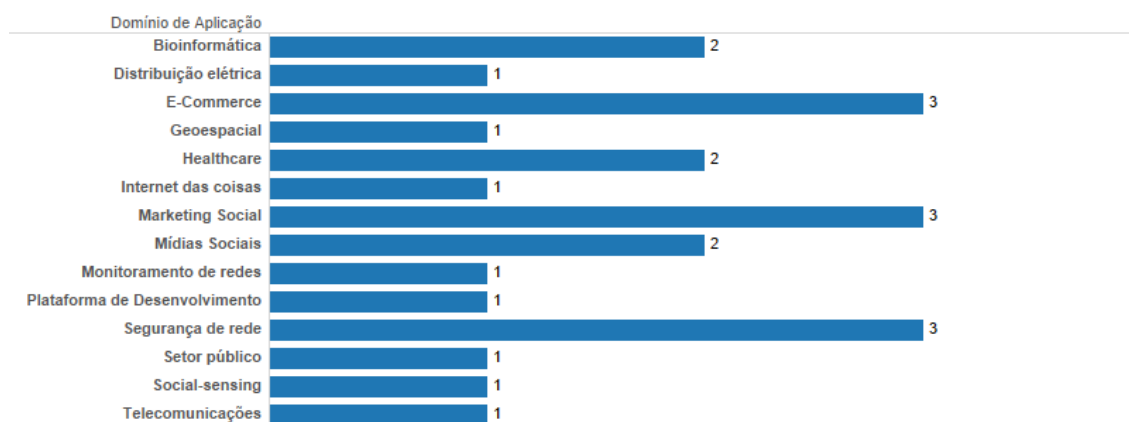


Figura 10 - Distribuição de estudos primários com estudo empírico por domínio de aplicação

Em seguida, os domínios Bioinformática, *Healthcare*, *Mídia Social* e *Telecomunicações* também se apresentam como um foco de estudo, porém em menor escala. É importante mencionar também que quatro estudos (CHEN; KAZMAN; HAZIYEV, 2016a) (CHEN; KAZMAN; HAZIYEV, 2016b) (CHEN; KAZMAN; HAZIYEV, 2016c) (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015) que empreenderam estudo empírico são correlatos, isto é, utilizaram dos mesmos estudos de caso para coleta de informações. Isto se dá especialmente por serem provenientes dos mesmos autores e os mesmos utilizaram dos mesmos estudos de caso, porém cada estudo se introduz em um contexto diferente. Nesse contexto, para evitar a repetição na contagem de domínios de aplicação, apenas os domínios do estudo *Big Data System Development: Na Embedded Case Study with a Global Outsourcing Firm* (CHEN; KAZMAN; HAZIYEV, 2015) foi considerado nos resultados.

Como um complemento para os resultados exibidos na Figura 10, a Tabela 15 apresenta as abordagens presentes em cada domínio de aplicação. É verificado que as abordagens mais aplicadas, *Desenvolvimento de Sistema*, *Método para Design de Sistemas* e *Proposta de Metodologia de Desenvolvimento*, estão presentes nos domínios de aplicação com maior número de estudos. Por outro lado, a abordagem *Proposta de Arquitetura de Software*, por mais que seja uma das abordagens mais citadas, é objeto de aplicação em apenas 2 domínios de aplicação distintos. Isso se dá por, de um total de 7 estudos, apenas 3 apresentarem a aplicação de um estudo empírico (KRAEMER; SENNER, 2015) (NITTO *et al.*, 2016) nesta abordagem. Isso indica um baixo nível de maturidade para a mesma. Por fim, a abordagem com maior número de domínios diferentes é *Proposta de metodologia de*

desenvolvimento, o que se relaciona com o fato de ser a segunda abordagem com maior número de estudos empíricos (ver Figura 4).

Tabela 15 – Divisão de domínios de aplicação por abordagem

Abordagem	Domínio de Aplicação	Contagem de estudos
Compartilhamento de experiência	Mídias Sociais	1
Desenvolvimento de sistema	Bioinformática	2
	Internet das coisas	1
Monitoramento de time de desenvolvimento	Segurança de rede	1
Proposta de abordagem para engenharia de requisitos	Setor público	1
Proposta de arquitetura de software	Geoespacial	1
	Social-sensing	1
Proposta de método para design de arquitetura de sistemas	Mídias Sociais	1
Proposta de método para design de sistemas	E-Commerce	1
Proposta de método para modelagem de sistemas	E-Commerce	1
Proposta de metodologia de desenvolvimento	Distribuição elétrica	1
	E-Commerce	1
	Healthcare	1
	Marketing Social	3
	Monitoramento de redes	1
	Plataforma de Desenvolvimento	1
	Segurança de rede	2
	Telecomunicações	1
	Healthcare	1
Proposta de modelo de prototipação	Healthcare	1

Fase do Ciclo de Desenvolvimento de Software (QP1d)

RQ1d: A quais fases do ciclo de vida do desenvolvimento de software as abordagens propostas se aplicam?

Para a definição de que fase do ciclo de vida do desenvolvimento de software o estudo primário se aplica, as abordagens propostas e seus respectivos objetivos de estudo foram identificados; isto é, as fases são explicitadas no texto ou inferidas pelos autores. Por exemplo, “*the first attempt to systematically combine architecture design with data modeling approaches to address big data system development challenges*” (CHEN; KAZMAN; HAZIYEV; HRYTSAY, 2015), identifica as fases de design e modelagem de uma aplicação big data.

A Figura 11 apresenta o número de estudos por fase do ciclo de vida identificados. É possível observar que a fase de design tem vantagem significativa em comparação com os demais, seguido pela fase de Modelagem. Novamente, uma possível explicação para o fato se dá pela grande ênfase dos estudos em propostas e métodos de design para arquitetura de software.

Um dado importante a se notar é a maior presença da fase *Levantamento de Requisitos* em relação à *Implementação* e *Manutenção*. Isso demonstra um esforço na identificação de requisitos para aplicações big data, onde a criticidade na identificação de tipos de dados processados pelo sistema é maior em comparação com o desenvolvimento de sistemas tradicional.

Para esta categoria, foi identificado que, para alguns estudos, a abordagem proposta não se aplica a uma fase específica do ciclo de vida, mas sim a todo o ciclo de vida, dessa forma, foram excluídos dos resultados. As abordagens que se enquadram nesse contexto são: *Desenvolvimento de Sistema*, *Proposta de Metodologia de Desenvolvimento*, *Monitoramento de Time de Desenvolvimento* e *Compartilhamento de Experiência*. Estes são identificados na Figura 10 como “Não se aplica”.

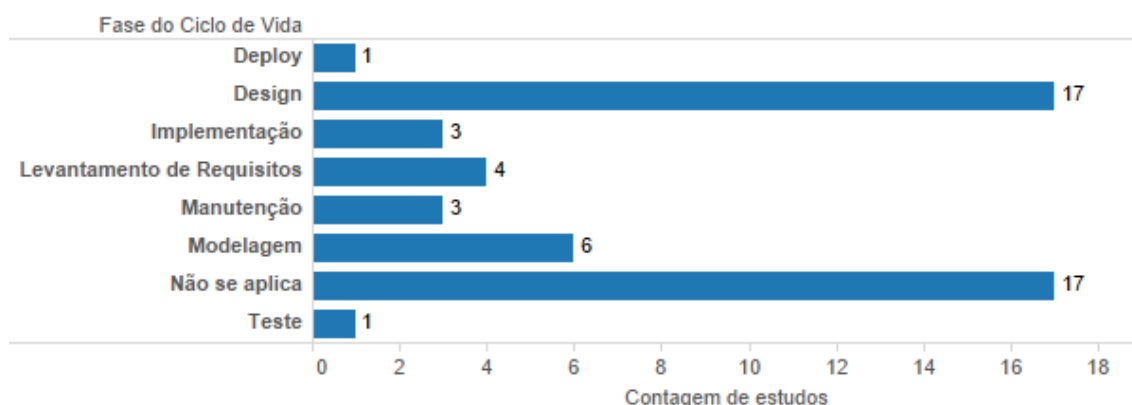


Figura 11 – Distribuição de estudos primários por aplicabilidade em fase do ciclo de vida

Tipo de Contribuição (PQ1a)

QP1a. Que tipos de contribuição de pesquisa tem sido realizadas pelas abordagens propostas?

A figura 12 apresenta os tipos de contribuição observados, decompondo as contribuições por publicação em conferências e workshops, periódico e relatório técnico. É possível observar a larga concentração de estudos que possuem como tipo de contribuição *Metodologia*, que se caracterizam por definir um conjunto sistemático de métodos para resolução de um problema. Esse resultado é explicado pelo conjunto de estudos que possuem como abordagem uma proposta de metodologia, design e modelagem de sistemas big data. Em seguida, *Ferramenta*, com 8 estudos, se apresenta como o segundo tipo de contribuição mais comum. Esse número é caracterizado pelo alto número de abordagens *Desenvolvimento de Sistema*, onde de um total de 12 estudos, 7 deles se apresentam como tipo de contribuição *Ferramenta*. Em adição, *Arquitetura*, também com 8 estudos, é impulsionado pelos estudos com abordagem *Proposta de Arquitetura de Software*. É evidente que, na maioria dos tipos de contribuição identificados, excluindo *Metodologia*, a diferença entre estudos presentes em conferências e workshops em comparação com estudos publicados em periódicos é significativa.

Para a identificação da tendência de estudos nesta categoria, a Figura 13 ilustra a porcentagem de estudos por tipo de contribuição ano a ano. É possível observar que os tipos

de contribuição se concentravam em *Processo*, *Ferramenta*, *Teoria* e *Arquitetura* até 2013. Entretanto, a partir de 2014, se inicia a hegemonia da contribuição de pesquisa *Metodologia*. Ainda em 2014, os primeiros estudos com contribuição *Plataforma* e *Framework* são observados. Por fim, no último ano da coleta de estudos primários, a distribuição de estudos por tipo de contribuição se mostra bastante heterogênea, indicando um grau de diversidade.

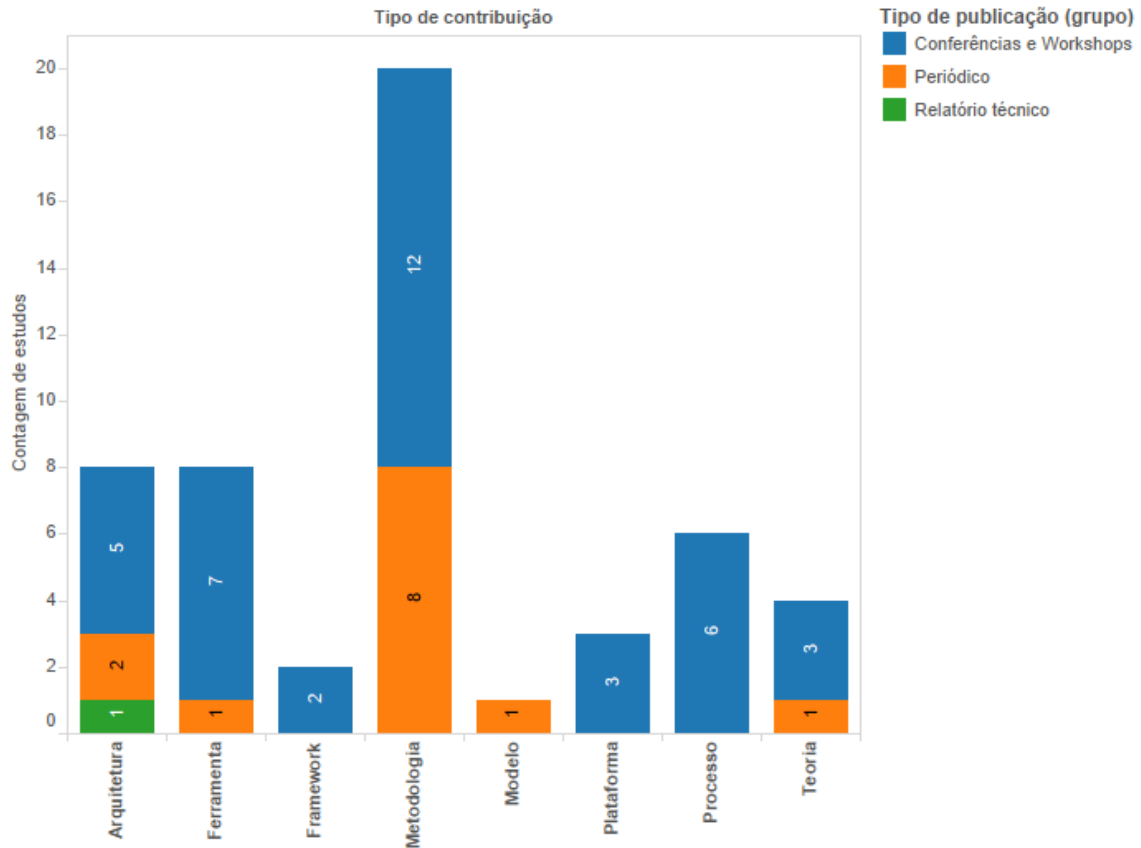


Figura 12 – Número de estudos por tipo de contribuição

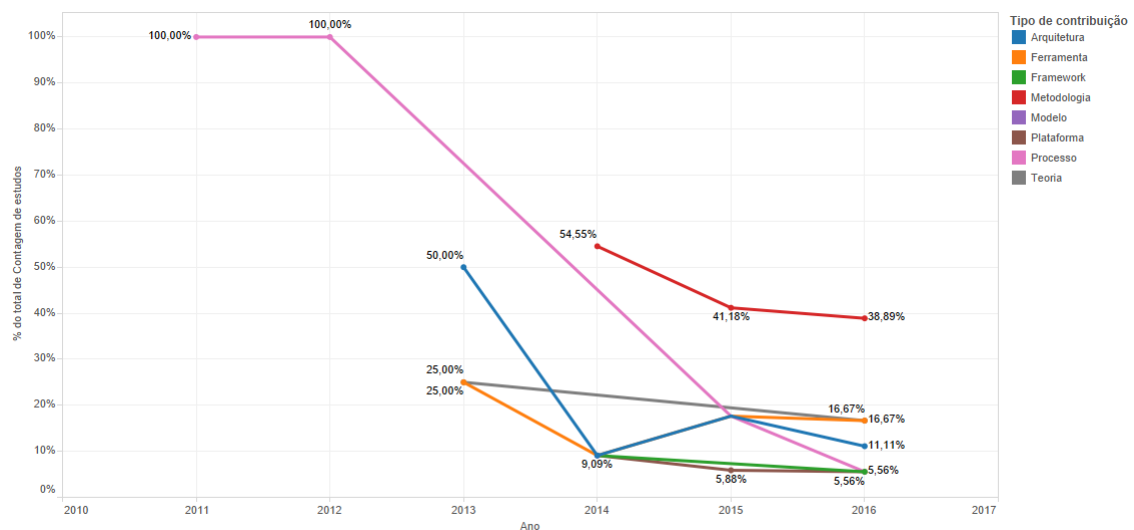


Figura 13 - Porcentagem de estudos por tipo de contribuição ano a ano

Tipo do Autor (QP1g)

QP1g: Qual o grau de colaboração entre a indústria e a academia na proposta de abordagem de cada estudo?

Nesta categoria busca-se representar o grau de contribuição entre academia e indústria nos estudos primários identificados. Big data é uma área com alta aplicabilidade em problemas na indústria, abarcando diversos domínios de aplicação, como exposto nesta seção. A Figura 14 mostra o número de estudos por tipo de autor. É possível identificar que majoritariamente os estudos possuem apenas autores da academia, com 42 estudos (80,77%), enquanto que a parceria academia/indústria ocorre apenas em 7 estudos (13,46%). Por último, estudos com autores somente da indústria se referem a apenas 3 estudos, correspondendo a 5,77% do total.

A ausência de autores da indústria na maioria dos estudos pode caracterizar dois contextos: a resolução de problemas muito específicos da academia, a resolução de problemas não aplicáveis às necessidades da indústria ou a aplicação de métodos ou processos não adequados à indústria. Entretanto, essa hipótese pode ter sua probabilidade reduzida quando no emprego de um estudo empírico pelo autor, pois infere-se a revisão por pares para a entrada em uma conferência ou periódico. Dessa forma, a Figura 14 também distribui o número de estudos por estudo empírico empregado. Assim, é possível observar que quando o estudo só possui autores da academia, mais da metade (23/41) emprega um estudo empírico.



Figura 14 - Número de estudos por tipo de autor

Foro de publicação (PQ1f)

PQ1f: Em que foros os estudos propondo abordagens para a construção de sistemas big data estão sendo publicados?

A Figura 15 apresenta os foros de publicação com seu respectivo número de estudos. Uma parte significativa dos estudos se concentram no International Workshop on Software Engineering. Outra pequena concentração de estudos se dá no IEEE Software, International Conference on System Sciences e International Workshop on Quality-Aware DevOps. Entretanto, é importante notar que publicações no contexto de sistemas big data estão surgindo em outros foros como International Workshop on Modeling in Software Engineering, International Workshop on Real-Time and Distributed Computing in Emerging Applications, International Conference on Software Engineering,

International Workshop on the Future of Software Architecture Design Assistants e International Conference on Data and Software Engineering. Houve um total de 33 diferentes foros com um ou mais estudos cada. A tabela com todos os foros e seus respectivos número de estudos podem ser acessados no Apêndice. A categoria “Outros” representa os foros que possuem apenas um estudo primário neste trabalho. A lista completa de foros identificados por este estudo se encontra no Apêndice B.

Em seguida, a Tabela 16 exibe a distribuição dos estudos por tipo de publicação. É verificado que 72% (36 estudos) dos estudos foram publicados em conferências e workshops, 26% (13 estudos) em periódicos e 2% (1 estudo) se refere a um relatório técnico. A maioria dos estudos publicados em conferências e workshops, em conjunto com apenas pouco mais de $\frac{1}{4}$ do total de estudos publicados em periódicos, indica que a área ainda está em processo de amadurecimento.

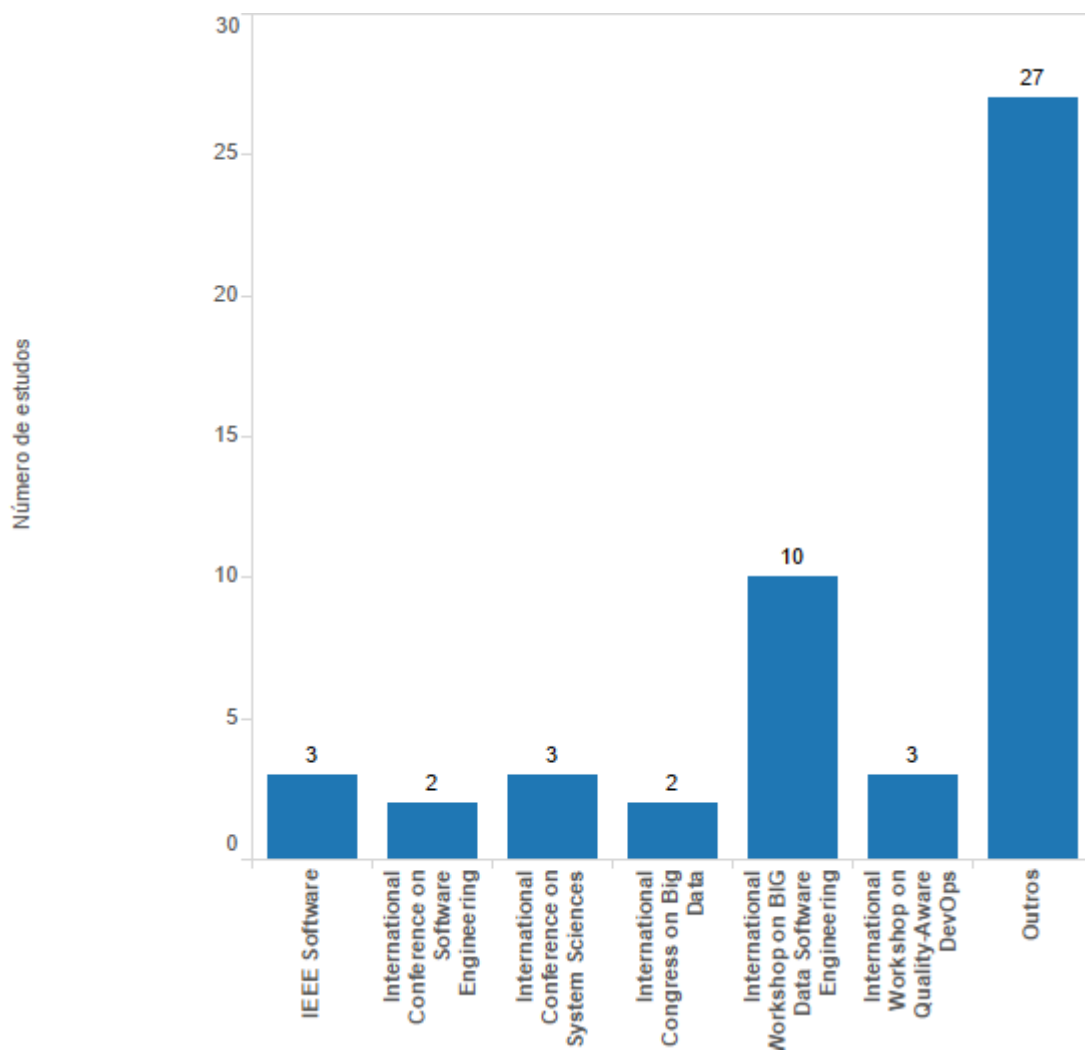


Figura 15 – Número de publicações por foro

Em uma análise ano a ano, é possível identificar que o número de estudos publicados em conferências e workshops cresce a cada ano. A mesma característica é identificada para periódicos, porém em menor escala, visto que os primeiros estudos se iniciam em 2014, com um aumento de 2 estudos no ano seguinte. Por último, é importante notar o número crescente de estudos desde 2011, ano do primeiro estudo que atende os critérios de inclusão. Nesse contexto, podemos identificar no futuro uma tendência de maior número de estudos em periódicos.

Tabela 16 – Número de estudos por tipo de publicação ano a ano

Tipo de publicação	2011	2012	2013	2014	2015	2016	Total
Conferência e workshop	1	1	3	8	11	12	36
Periódico				3	5	5	13
Relatório técnico			1				1

4.3 SÍNTESE DOS RESULTADOS

Nesta seção é apresentada uma síntese dos resultados. O objetivo deste mapeamento sistemático é identificar as principais abordagens da engenharia de software para a construção de sistemas big data. Para isso, foram analisados 305 estudos com o objetivo de identificar os objetivos de estudo, os tipos de contribuição e as abordagens no contexto de sistemas com uso intensivo de dados.

Com o objetivo de consolidar os resultados obtidos por esse mapeamento, a Figura 16 detalha a relação existente entre o número de estudos analisados por abordagem, tipos de contribuição e tipos de pesquisa. É possível observar que a maioria dos estudos empreendem como abordagem *Desenvolvimento de sistema*, tendo a maioria se posicionando como contribuição em *Ferramenta*, ou seja, visam a construção de uma solução de software para endereçar um problema. É também possível detectar outro foco de estudos com abordagem *Proposta de arquitetura de software*, sendo quase a totalidade do foco na contribuição de Arquitetura, onde há uma descrição teórica e/ou de implementação no contexto de decisões de arquitetura em um software. Em seguida, a abordagem *Metodologia de Desenvolvimento* se apresenta como um foco de estudos dentre outras abordagens que possuem grande parte de seus estudos com o tipo de contribuição *Metodologia*, como *Método para design de sistemas*.

Por outro lado, é importante notar que as abordagens com maior foco têm a maior parte de seus estudos empregando o tipo de pesquisa *Proposal of solution*, onde não é empregado uma validação rigorosa.

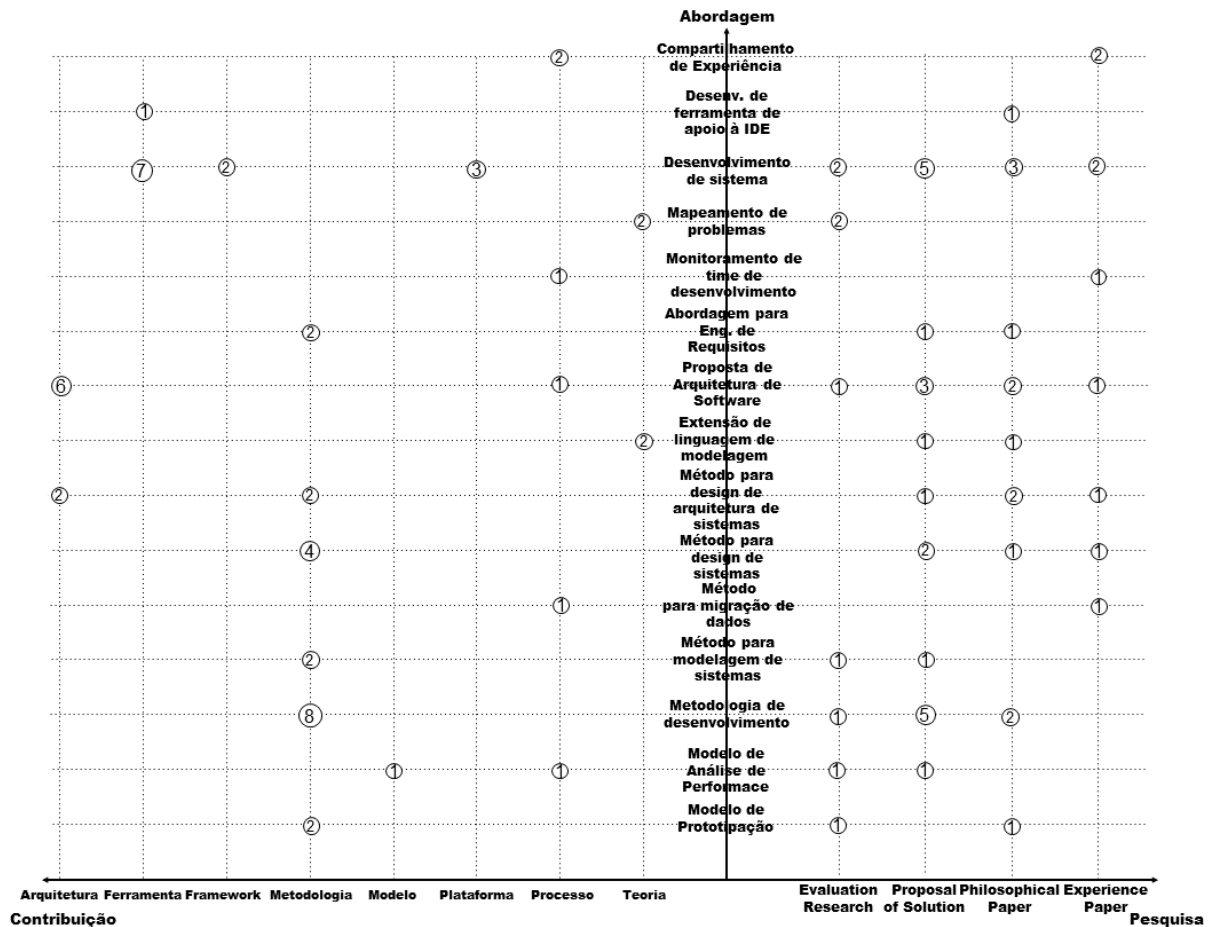


Figura 16 - Visualização do mapeamento sistemático em forma de gráfico em bolhas

4.4 CONSIDERAÇÕES FINAIS

Como pode ser visto nas seções anteriores, houve um grande conjunto de informações extraídas e estas servem de base para diversas conclusões. Foram definidas classificações para as abordagens e os objetivos na construção de sistemas com uso intensivo de dados. Além disso, foi identificada a evolução dos estudos ao longo do tempo, os tipos de contribuição, o grau de interseção entre a indústria e a academia, os domínios de aplicação onde um estudo empírico foi realizado e os métodos de pesquisa empregados. Dessa forma, o mapeamento sistemático apresentado apontou um contínuo interesse na área de desenvolvimento de sistemas big data e a tendência é termos mais estudos ao longo do tempo, contribuindo assim para maior maturidade da área.

O capítulo 5 discorre sobre os principais resultados do mapeamento e as contribuições deste trabalho.

CAPÍTULO 5 – CONCLUSÃO

5.1 CONTRIBUIÇÕES

A principal contribuição deste trabalho é o mapeamento sistemático empreendido para a área de desenvolvimento de sistemas big data. Este teve por objetivo contribuir para o amadurecimento dos estudos na área, promover a aplicação de estudos empíricos, identificar janelas para aperfeiçoamento de estudos e permitir que pesquisadores e profissionais utilizem este trabalho como um corpo de conhecimento sobre a área.

Como contribuição secundária deste trabalho temos o protocolo proposto para um mapeamento sistemático na área da engenharia de software, uma estratégia híbrida de busca em bibliotecas digitais, que dado o grau de maturidade da área, envolveu a seleção de artigos de controle, a busca manual por um conjunto inicial de estudos que atendiam ao critérios de inclusão, a definição da string de busca com base nos estudos identificados anteriormente, a submissão da string de busca nas bibliotecas digitais e recuperação de estudos e aplicação da técnica de *snowballing*, após aplicação dos critérios de inclusão. Além disso, uma técnica de *grounded theory*, cunhada como *coding*, foi experimentada para identificação de categorias de estudo. Essa técnica se mostrou eficiente em uma área com poucos estudos secundários e ainda imatura, permitindo que se determinasse classificações a partir de teoria.

Com base no mapeamento foi possível observar que os maiores desafios estão na área da arquitetura de software, onde a construção de uma infraestrutura adequada, envolvendo a modelagem de dados, o design da aplicação, integrações entre diferentes tecnologias de banco de dados e aplicações, muitas vezes se colocam como barreiras para o desenvolvimento de uma solução adequada ao objetivo do estudo. Dessa forma, há um significativo número de estudos onde o foco é construir uma teoria para a definição, prototipação e implementação de soluções arquiteturais de software, e outros onde o foco é desenhar uma metodologia para definição da arquitetura de software da solução. Além disso, há um grande foco de estudos na fase do ciclo de vida design. Isso demonstra um alto interesse de pesquisadores nessa fase específica, alavancada pelo alto número de estudos focados em solucionar problemas relacionados à definição e implementação da arquitetura de software em sistemas com uso intensivo de dados. Mas como a fase de design não apenas se furta ao design da arquitetura de uma solução de software, estudos também endereçam o design do modelo de dados definido.

Outro dado importante extraído das categorias de pesquisa diz respeito aos objetivos de pesquisa. Há um considerável número de estudos que tem por objetivo *Prover uma*

metodologia para o desenvolvimento de sistemas. Podemos observar com isso que há um entendimento, tanto por parte da indústria, como da academia, que as metodologias tradicionais não abarcam as características necessárias para o desenvolvimento de sistemas big data.

Além disso, como um sinal de imaturidade, muitos autores argumentam em seus estudos que mais experimentação é necessária e que no futuro ela deve ser empregada. Isso coloca em cheque a validade das abordagens propostas. Em alguns casos, houve a implementação de uma prova de conceito, entretanto, sem empreender um estudo empírico. A área ainda está em sua infância, entretanto demonstra boas perspectivas futuras, uma vez que há o crescimento de estudos com o tipo de pesquisa *Evaluation research*. Como qualquer nova área, é normal a identificação de índices que demonstrem imaturidade.

5.2 LIMITAÇÕES E AMEAÇAS À VALIDADE

O fato de apenas um pesquisador ter empreendido o processo de mapeamento sistemático na totalidade de suas fases pode incorrer em erros e lacunas nas categorizações e informações obtidas dos estudos primários. É verificado que os mapeamentos sistemáticos são compostos sempre por mais de um pesquisador. Desta forma, a atuação em fases importantes do mapeamento, como a classificação dos estudos, as informações são coletadas por um conjunto de pesquisadores e debatidas com o objetivo de prover maior qualidade na informação. Por mais que o orientador defina objetivos do estudo e auxilie na tomada de decisões no trabalho monográfico, o problema exposto acima não é minimizado.

Uma limitação é o fato de as buscas nas bibliotecas digitais terem sido rodadas em Janeiro de 2017. Alguns estudos mais recentes podem ter ficado de fora das classificações. Entretanto, o impacto desse exposto não é maior que a primeira mencionada. Portanto, como endereçado na Seção 3.3.1, consideramos no escopo apenas estudos publicados até o fim de 2016.

5.3 TRABALHOS FUTUROS

Para este trabalho, é possível delinear dois principais trabalhos futuros: a atualização dos estudos primários selecionados com publicações realizadas no ano de 2017 e o desenvolvimento de uma infraestrutura para suportar o processo de realização de um mapeamento sistemático.

Sobre a atualização dos estudos, uma nova rodada de buscas nas bibliotecas digitais definidas na Seção 3.3.3.2 deve ser realizada, excluindo os estudos previamente excluídos e

selecionados. Assim, um novo processo de revisão dos estudos não identificados na primeira rodada de buscas deve ser empreendido, empregando os critérios de inclusão e exclusão. É importante notar que nessa segunda rodada de busca de estudos primários, a seleção de artigos de controle e de um conjunto inicial de artigos para auxiliar no processo de definição da string de busca não é mais necessário, visto que neste estudo já foi concebido um corpo de conhecimento sobre a área de estudo, isto é, os principais autores e estudos de referência já foram identificados, bem como as palavras-chave e principais termos. Após a aplicação do critério de inclusão, seria necessário realizar a atualização dos resultados de estudo categoria a categoria, e identificar se as novas publicações incluídas entre os estudos primários auxiliam para maior maturidade da área de estudo.

Por outro lado, ao longo deste trabalho, detectou-se que muitos esforços realizados em um mapeamento sistemático poderiam ser extintos ou minimizados. Para esta dimensão, há três vertentes: a automatização do processo de recuperação dos estudos de bibliotecas digitais por meio de programas que implementam bibliotecas de automação de tarefas, como soluções de robotização (*bot*); o desenvolvimento de uma ferramenta de software para gerenciamento do mapeamento, permitindo o controle de diferentes versões do mapeamento, visto que o mesmo pode ser repetido ao longo dos anos, e o controle dos estudos primários que se adequam ao critério de inclusão; por último, a aplicação de técnicas de *text-mining* ou aprendizado de máquina para apoiar (semi automatização) o processo de seleção de possíveis estudos que atendam ao critério de inclusão.

Para a vertente de automatização do processo de recuperação dos estudos diretamente das bibliotecas digitais, é possível empreender o desenvolvimento de uma solução de robotização (por exemplo, por meio de ferramentas de mercado como o UIPath) que recupere os estudos com base em um conjunto de regras, como a definição de uma string de busca para cada biblioteca buscada e também considerar uma *blacklist*, estudos que não precisam ser recuperados pois já passaram pelo processo de seleção anteriormente.

Em seguida, uma ferramenta de software para gerenciamento de um mapeamento pode prover ao pesquisador um controle sobre os estudos buscados, permitindo gerenciar estudos primários que atendam ao critério de inclusão e as revisões de diferentes pesquisadores envolvidos no mapeamento, bem como prover o cadastro de todas as categorias e metadado relacionados à recuperação dos estudos. Uma lista detalhada de requisitos esperados pela ferramenta pode ser visualizada no Apêndice C.

Por último, seria importante investigar a aplicabilidade de técnicas de *text-mining* ou aprendizado de máquina para apoiar o processo de seleção de estudos primários que se adequem aos critérios de inclusão pré-estabelecidos.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABOUTORABI, S. H.; REZAPOUR, M.; MORADI, M; GHADIRI, N. **Performance evaluation of SQL and MongoDB databases for big e-commerce data.** International Symposium on Computer Science and Software Engineering (CSSE), IEEE, ago. 2015.
- AHMED, Z.; ZEESHAN, S.; DANDEKAR, T. **Developing sustainable software solutions for bioinformatics by the “Butterfly” paradigm.** F1000 Research, 3, out. 2014.
- ALVES, N. S. R.; MENDES, T. S.; MENDONÇA, M. G.; SPÍNOLA, R. O.; SHULL, F.; SEAMAN, C. **Identification and management of technical debt: A systematic mapping Study.** Information and Software Technology, ScienceDirect, v. 70, p. 100-121, 2016.
- ANDERSON, J.; SODEN, R.; ANDERSON, KENNETH M. A.; KOGAN, M.; PALEN, L. **EPIC-OSM: A Software Framework for OpenStreetMap Data Analytics.** 49th Hawaii International Conference on System Sciences (HICSS), IEEE, jan. 2016.
- ANDERSON, K. M.; SCHRAM, A. **Design and Implementation of a Data Analytics Infrastructure in Support of Crisis Informatics Research (NIER Track).** 33rd International Conference on Software Engineering (ICSE). IEEE, out. 2011.
- ANDERSON, K.; SCHRAM, A.; ALZABARAH, A.; PALEN, L. **Architectural Implications of Social Media Analytics in Support of Crisis Informatics Research.** IEEE Bulletin of the Technical Committee on Data Engineering, v. 36, p. 13-20, set. 2013.
- AUDSLEY, N. C.; CHAN, Y.; GRAY, I.; WELLINGS, A. J. **Real-Time Big Data the JUNIPER Approach.** 3rd IEEE International Workshop on Real-time and distributed computing in emerging applications. Universidad Carlos III de Madrid, nov. 2014.
- BIGDSE. **Proceedings of the 2nd International Workshop on BIG Data Software Engineering.** Disponível em: <<https://sse.uni-due.de/bigdse16/>>. Acesso em 29/10/2017
- BODORIK, J.; JUTLA, D. N. **PAUSE: A Privacy Architecture for Heterogeneous Big Data Environments.** IEEE International Conference on Big Data (Big Data). IEEE, dez. 2015.
- BURBANK, D. **The 5 V’s of Big Data.** Disponível em: <<https://www.elearning.com/resources/blog/the-5-v%E2%80%99s-of-big-data.html>>. Acesso em 29/10/2017.
- CAPGEMINI. **Cracking the Data Conundrum: How Successful Companies Make Big Data Operational.** 2015.
- CAPGEMINI; INFORMATICA. **The Big Data Payoff: Turning Big Data into Business Value.** 2016.
- CARR, N, G. **IT Doesn’t Matter.** Harvard Business Review, mai. 2003.

- CASALE, G.; ARDAGNA, D.; ARTAC, M.; BARBIER, F.; NITTO, E. D.; HENRY, A.; IUHASZ, G.; JOUBERT, C.; MERSEGUER, J.; MUNTEANU, V. I.; PÉREZ, J. F.; PETCU, D.; ROSSI, M.; SHERIDAN, C.; SPAIS, I.; VLADUIC, D. **DICE: Quality-Driven Development of Data-Intensive Cloud Applications**. Proceedings of the Seventh International Workshop on Modeling in Software Engineering, IEEE Press, p. 78-83, mai. 2015.
- CHEN, H.; KAZMAN, R.; HAZIYEV, O. **Big Data System Development: An Embedded Case Study with a Global Outsourcing Firm**. IEEE/ACM 1st International Workshop on Big Data Software Engineering (BIGDSE), IEEE, mai. 2015.
- CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Agile Big Data Analytics Development: An Architecture-Centric Approach**. 49th Hawaii International Conference on System Sciences. IEEE, mar. 2016c.
- CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Agile Big Data Analytics for Web-Based Systems: An Architecture-Centric Approach**. IEEE Transactions on Big Data, IEEE, vol. 2, 3, p. 234-248, mai. 2016a.
- CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Strategic Prototyping for Developing Big Data Systems**. IEEE Software, IEEE, 33, 2, p. 36-43, fev. 2016b.
- CHEN, K.; LI X.; WANG, H. **On the model design of integrated intelligent big data analytics systems**. Industrial Management & Data Systems, v. 115, 9, p. 1666-1682, 2015.
- DYBA, T.; KITCHENHAM, B.A.; JORGENSEN, M. **Evidence-based Software Engineering for Practitioners**. IEEE Software, IEEE, v. 22, 1, p. 58-65, jan. 2014.
- EVANS, E. **Domain-Driven Design: Tackling Complexity in the Heart of Software**. Addison-Wesley, 2004.
- GANDOMI, A; HAIDER, M. **Beyond the hype: Big data concepts, methods, and analytics**. International Journal of Information Management, v. 35, p. 137-144, 2015.
- GIL, D.; SONG, I. **Modeling and Management of Big Data Challenges and opportunities**. Future Generation Computer Systems, ScienceDirect v. 63, p. 96-99, out. 2016.
- GLASER, B. G. **Basics of Grounded Theory Analysis: Emergence vs Forcing**. Sociology Press, 1992.
- GORTON, I.; KLEIN, J. **Distribution, Data, Deployment Software Architecture Convergence in Big Data Systems**. IEEE Software, IEEE, 32, 3, jun. 2015.
- HU, H.; WEN, Y.; CHUA, T.; LI, X. **Toward Scalable Systems for Big Data Analytics: A Technology Tutorial**. IEEE Access, IEEE, v. 2, p. 652-687, jun. 2014.
- KAZMAN, R. **Prototyping for developing big data systems**. Disponível em <https://insights.sei.cmu.edu/sei_blog/2016/07/prototyping-for-developing-big-data-systems.html>. Acesso em 29/10/2017.

KITCHENHAM, B. **Procedures for Performing Systematic Reviews**. Keele, UK, Keele University, 33, p. 1-26, ago. 2004.

KRAEMER, M.; SENNER, I. **A modular software architecture for processing of big geospatial data in the cloud**. Computers & Graphics, v. 49, p. 69-81, jun. 2015.

LANEY, D. **3D Data Management: Controlling Data Volume, Velocity, and Variety**. Meta Group, 2001.

MADHAVJI, N. H.; MIRANSKY, A.; KONTOGIANNIS, K. **Big Picture of Big Data Software Engineering With example research challenges**. IEEE/ACM 1st International Workshop on Big Data Software Engineering (BIGDSE), IEEE, mai. 2015.

MOCKUS, A. **Engineering Big Data Solutions**. Proceedings of the on Future of Software Engineering. ACM, p. 85-99, mai. 2014.

NITTO, E.; JAMSHIDI, P.; GUERRIERO, M.; SPAIS, I.; TAMBURRI, D. A. **A Software Architecture Framework for Quality-aware DevOps**. Proceedings of the 2nd International Workshop on Quality-Aware DevOps. p. 12-17, jul. 2015.

NOORWALI, I.; ARRUDA, D.; MADHAVJI, N. H. **Understanding Quality Requirements in the Context of Big Data Systems**. Proceedings of the 2nd International Workshop on BIG Data Software Engineering, p. 76-79, mai. 2016.

PÄÄKKÖNEN, P.; PAKKALA, D. **Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems**. Big Data Research, v. 2, 4, p. 166-186, dez. 2015.

PETERSEN, K.; FELDT, R.; MUJTABA, S. MATTSSON, M. **Systematic Mapping Studies in Software Engineering**. Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering, BCS Learning & Development Ltd., p. 68-77, jun. 2008.

PORTER, M.; MILLAR, V. **How Information Gives You Competitive Advantage**. Harvard Business Review, jul. 1985.

ROBINSON, C. **Real World Research A resource for Social Scientists and Practitioner-Researchers**, 3. ed. Wiley, 2002.

ROSENTHAL, S.; MCMILLAN, S.; Matthew, E. G. **Developer Toolchains for Large-Scale Analytics: Two Case Studies**. IEEE International Conference on Big Data (Big Data). IEEE, dez. 2015.

RUNESON, P.; HÖST, M. **Guidelines for conducting and reporting case study research in software engineering**. Empirical Software Engineering, v. 14, 2, 131p. 2009.

SAS. **Big Data What it is and why it matters**. Disponível em: <https://www.sas.com/en_us/insights/big-data/what-is-big-data.html>. Acesso em 29/10/2017.

STOL, K.J.; RALPH, P.; FITZGERALD, B., **Grounded theory in software engineering research: a critical review and guidelines**. IEEE/ACM 38th International Conference on Software Engineering (ICSE). IEEE, p. 120-131, mai. 2016.

TECHAMERICA FOUNDATION'S FEDERAL BIG DATA COMMISSION. **Demystifying big data: A practical guide to transforming the business of Government**. Disponível em: <<http://www.techamerica.org/Docs/fileManager.cfm?f=techamericabigdatareport-final.pdf>>. Acesso em 24/12/2017.

VILLARI, M.; CELESTI, A.; FAZIO, M.; PULIAFITO, A. **AllJoyn Lambda: an Architecture for the Management of Smart Environments in IoT**. International Conference on Smart Computing, IEEE, nov. 2014.

WOHLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M.C.; REGNELL, B.; WESSLÉN, A. **Experimentation in Software Engineering**, Springer Publishing Company, 2012.

APÊNDICE A – ESTUDOS PRIMÁRIOS SELECIONADOS

ABOUTORABI, S. H.; REZAPOUR, M.; MORADI, M; GHADIRI, N. **Performance evaluation of SQL and MongoDB databases for big e-commerce data.** International Symposium on Computer Science and Software Engineering (CSSE), IEEE, ago. 2015.

ANDERSON, J.; SODEN, R.; ANDERSON, K.; KOGAN, M.; PALEN, L. **EPIC-OSM: A Software Framework for OpenStreetMap Data Analytics.** 49th Hawaii International Conference on System Sciences (HICSS), IEEE, jan. 2016.

ANDERSON, K. M. **Embrace the Challenges: Software Engineering in a Big Data World.** International Workshop on Big Data Software Engineering (BIGDSE), IEEE, jul. 2015.

ANDERSON, K. M.; SCHRAM, A. **Design and Implementation of a Data Analytics Infrastructure in Support of Crisis Informatics Research (NIER Track).** 33rd International Conference on Software Engineering (ICSE). IEEE, out. 2011.

ANDERSON, K.; SCHRAM, A.; ALZABARAH, A.; PALEN, L. **Architectural Implications of Social Media Analytics in Support of Crisis Informatics Research.** IEEE Bulletin of the Technical Committee on Data Engineering, v. 36, p. 13-20, set. 2013.

Applications in the Cloud. IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW), IEEE, set. 2013.

AUDSLEY, N. C.; CHAN, Y.; GRAY, I.; WELLINGS, A. J. **Real-Time Big Data the JUNIPER Approach.** 3rd IEEE International Workshop on Real-time and distributed computing in emerging applications. Universidad Carlos III de Madrid, nov. 2014.

BAZARGANI, S., BRINKLEY, J., TABRIZI, N. **Implementing conceptual search capability in a cloud-based feed aggregator.** International Conference on Innovative Computing Technology (INTECH), IEEE, ago. 2013.

BERSANI, F.; ERASCU, M. **A tool for verification of big-data applications.** Proceedings of the 2nd International Workshop on Quality-Aware DevOps, ACM, p. 44-45, jul. 2016.

BODORIK, J.; JUTLA, D. N. **PAUSE: A Privacy Architecture for Heterogeneous Big Data Environments.** IEEE International Conference on Big Data (Big Data). IEEE, dez. 2015.

BREWER, W.; SCOTT, W.; SANFORD, J. **An Integrated Cloud Platform for Rapid Interface Generation, Job Scheduling, Monitoring, Plotting, and Case Management of Scientific Applications.** International Conference on Cloud Computing Research and Innovation (ICCCRI), IEEE, out. 2015.

CASALE, G.; ARDAGNA, D.; ARTAC, M.; BARBIER, F.; NITTO, E. D.; HENRY, A.; IUHASZ, G.; JOUBERT, C.; MERSEGUER, J.; MUNTEANU, V. I.; PÉREZ, J. F.; PETCU, D.; ROSSI, M.; SHERIDAN, C.; SPAIS, I.; VLADUIC, D. **DICE: Quality-Driven Development of Data-Intensive Cloud Applications.** Proceedings of the Seventh

International Workshop on Modeling in Software Engineering, IEEE Press, p. 78-83, mai. 2015.

CECCHINEL, C.; JIMENEZ, M.; MOSSER, S.; RIVEILL, M. **An Architecture to Support the Collection of Big Data in the Internet of Things.** World Congress On Services (SERVICES), IEEE, jul. 2014.

CERQUEUS, T.; ALMEIDA, E. C. D.; SCHERZINGER, S. **Safely Managing Data Variety in Big Data Software Development.** IEEE/ACM International Workshop on Big Data Software Engineering, IEEE, mai. 2015.

CHEN, H.; KAZMAN, R.; GARBAJOSA, J.; GONZALEZ, E. **Toward Big Data Value Engineering for Innovation.** International Workshop on BIG Data Software Engineering, IEEE, mai. 2016.

CHEN, H.; KAZMAN, R.; HAZIYEV, O. **Big Data System Development: An Embedded Case Study with a Global Outsourcing Firm.** IEEE/ACM 1st International Workshop on Big Data Software Engineering (BIGDSE), IEEE, mai. 2015.

CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Agile Big Data Analytics Development: An Architecture-Centric Approach.** 49th Hawaii International Conference on System Sciences. IEEE, mar. 2016.

CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Agile Big Data Analytics for Web-Based Systems: An Architecture-Centric Approach.** IEEE Transactions on Big Data, IEEE, v. 2, 3, p. 234-248, mai. 2016.

CHEN, H.; KAZMAN, R.; HAZIYEV, S. **Strategic Prototyping for Developing Big Data Systems.** IEEE Software, IEEE, v. 33, 2, p. 36-43, fev. 2016.

CHEN, K.; LI X.; WANG, H. **On the model design of integrated intelligent big data analytics systems.** Industrial Management & Data Systems, v. 115, 9, p. 1666-1682, 2015.

CHEN, S.; BRONEVETSKY, G.; PENG, L.; LI, B.; FU, X. **Soft error resilience in Big Data kernels through modular analysis.** The Journal of Supercomputing, v. 72, 4, p. 1570-1596, mar. 2016.

ERIDAPUTRA, H.; HENDRADJAYA, B.; SUNINDYO, W. D. **Modeling the requirements for big data application using goal oriented approach.** International Conference on Data and Software Engineering (ICODSE), IEEE, nov. 2014.

GIL, D., SONG, I. **Modeling and Management of Big Data Challenges and opportunities.** Future Generation Computer Systems, v. 63, p. 96–99, out. 2016.

GÓMEZ, A.; MERSEGUER, J.; NITTO, E. D.; TAMBURRI, D. A. **Towards a UML Profile for Data Intensive Applications.** Proceedings of the 2nd International Workshop on Quality-Aware DevOps, ACM, p. 18-23, set. 2016.

GORTON, I.; KLEIN, J. **Distribution, Data, Deployment Software Architecture Convergence in Big Data Systems.** IEEE Software, IEEE, v. 32, 3, jun. 2015.

GUERRIERO M., TAJFAR S., TAMBURRI D.A., NITTO E. D. **Towards a model-driven design tool for big data architectures.** IEEE/ACM International Workshop on BIG Data Software Engineering, IEEE, 2016.

HE Z.L., XIAO X.H., HE Y.H. **A software design model based on big data.** Applied Mechanics and Materials, 644-650, p. 2821-2825, set. 2014.

HU, H.; WEN, Y.; CHUA, T.; Li, X. **Toward Scalable Systems for Big Data Analytics: A Technology Tutorial.** IEEE Access, IEEE, v. 2, p. 652-687, jun. 2014.

JUTLA, D. N.; BODORIK, P.; ALI, S. **Engineering Privacy for Big Data Apps with the Unified Modeling Language.** IEEE International Congress on Big Data (BigData Congress), IEEE, jul. 2013.

KLEIN J.; BUGLAK R.; BLOCKOW D.; WUTTKE T.; COOPER B. **A reference architecture for big data systems in the national security domain.** Proceedings of the 2nd International Workshop on BIG Data Software Engineering, ACM, p. 51-57, mai, 2016.

KLEIN, J.; GORTON, I. **Design Assistant for NoSQL Technology Selection.** International Workshop on the Future of Software Architecture Design Assistants, IEEE, mai. 2015.

KLEIN, J.; GORTON, I.; ERNST, N.; DONOHOE, P.; PHAM, K.; MATSER, C. **Application-Specific Evaluation of No SQL Databases.** IEEE International Congress on Big Data, jul. 2015.

KRAEMER, M.; SENNER, I. **A modular software architecture for processing of big geospatial data in the cloud.** Computers & Graphics, v. 49, p. 69-81, jun. 2015.

LI, C.; HUANG, L.; CHEN, L. **Breeze graph grammar: a graph grammar approach for modeling the software architecture of big data-oriented software systems.** Software: Practice and Experience, v. 45, 8, p. 1023-1050, mai. 2015.

MIRAKHORLI, M.; CHEN, H.; KAZMAN, R. **Mining Big Data for Detecting, Extracting and Recommending Architectural Design Concepts.** International Workshop on Big Data Software Engineering (BIGDSE), IEEE, mai. 2015.

MIRANSKY, A.; HAMOU-LHADJ, A.; CIALINI, E.; LARSSON, A. **Operational-Log Analysis for Big Data Systems Challenges and Solutions.** IEEE Software, v. 33, no. , p. 52-59, mar.-abr. 2016.

MOCKUS, A. **Engineering Big Data Solutions.** Proceedings of the on Future of Software Engineering. ACM, p. 85-99, mai. 2014.

NASEER A.; ALKAZEMI B.Y.; WARAICH E.U. **A big data approach for proactive healthcare monitoring of chronic patients.** International Conference on Ubiquitous and Future Networks (ICUFN), IEEE, ago. 2016.

NITTO, E.; JAMSHIDI, P.; GUERRIERO, M.; SPAIS, I.; TAMBURRI, D. A. **A Software Architecture Framework for Quality-aware DevOps.** Proceedings of the 2nd International Workshop on Quality-Aware DevOps. p. 12-17, jul. 2015.

NOORWALI, I.; ARRUDA, D.; MADHAVJI, N. H. **Understanding Quality Requirements in the Context of Big Data Systems.** Proceedings of the 2nd International Workshop on BIG Data Software Engineering, p. 76-79, mai. 2016.

PÄÄKKÖNEN, P.; PAKKALA, D. **Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems.** Big Data Research, v. 2, 4, p. 166-186, dez. 2015.

RAJBHOJ, A.; KULKARNI, V.; BELLARYKAR, N. **Early Experience with Model-driven Development of MapReduce based Big Data Application.** Asia-Pacific Software Engineering Conference (APSEC), IEEE, dez. 2014.

RINGLSTETTER, A.; SCHERZINGER, S.; BISSYANDÉ, T. **Data Model Evolution using Object-NoSQL Mappers: Folklore or State-of-the-Art?** IEEE/ACM 2nd International Workshop on BIG Data Software Engineering, IEEE, mai, 2016.

ROSENTHAL, S.; MCMILLAN, S.; Matthew, E. G. **Developer Toolchains for Large-Scale Analytics: Two Case Studies.** IEEE International Conference on Big Data (Big Data). IEEE, dez. 2015.

SCAVUZZO, M.; TAMBURRI, D. A.; NITTO, E. D. **Providing Big Data Applications with Fault-tolerant Data Migration Across Heterogeneous NoSQL Databases.** IEEE/ACM International Workshop on BIG Data Software Engineering, IEEE, mai. 2016.

SCHRAM, A.; ANDERSON, K. M. **Design challenges/solutions for environments supporting the analysis of social media data in crisis informatics research.** International Conference on System Sciences (HICSS), IEEE, jan. 2015.

SCHRAM, A.; ANDERSON, K. M. **MySQL to NoSQL Data Modeling Challenges in Supporting Scalability.** Conference on Systems, Programming, Languages and Applications: Software for Humanity, ACM, 2012.

SILVA, M. A. A. D.; SADOVYKH, A.; BAGNATO, A.; CHEPTSOV, A.; ADAM, L. **JUNIPER: Towards Modeling Approach Enabling Efficient Platform for Heterogeneous Big Data Analysis.** Software Engineering Conference, ACM, 2014.

VILLALPANDO, L. E. B.; APRIL, A. **Performance analysis model for big data applications in cloud computing.** Journal of Cloud Computing: Advances, Systems and Applications, v. 3, 9, dez. 2014.

VILLARI, M.; CELESTI, A.; FAZIO, M.; PULIAFITO, A. **AllJoyn Lambda: an Architecture for the Management of Smart Environments in IoT.** International Conference on Smart Computing, IEEE, nov. 2014.

YIM K.S. **Norming to performing: Failure analysis and deployment automation of big data software developed by highly iterative models.** IEEE International Symposium on Software Reliability Engineering, IEEE, nov. 2014.

ZHANG, Y.; XU, F.; FRISE, E.; WU, S.; YU, B.; XU, W. **DataLab: A Version Data Management and Analytics System**. IEEE/ACM 2nd International Workshop on BIG Data Software Engineering, IEEE, mai. 2016.

ZIMMERMANN, A.; PRETZ, M.; ZIMMERMANN, G.; FIRESMITH, D. G.; PETROV, I.; EL-SHEIKH, E. **Towards Service-oriented Enterprise Architectures for Big Data**. International Enterprise Distributed Object Computing Conference Workshops, IEEE, 2013.

APÊNDICE B – TABELA COM NÚMERO DE ESTUDOS POR FORO DE PUBLICAÇÃO

Tabela 17 - Número de estudos por foro de publicação

Foro de publicação	Contagem de estudos
Applied Mechanics and Materials	1
Asia-Pacific Software Engineering Conference	1
Big Data Research	1
Bulletin of the IEEE Computer Society Technical Committee on Data Engineering	1
Computers & Graphics	1
Conference on Systems, Programming, Languages and Applications: Software for Humanity	1
IEEE Access	1
IEEE Software	3
Industrial Management & Data Systems	1
International Conference on Big Data	2
International Conference on Cloud Computing Research and Innovation	1
International Conference on Data and Software Engineering	1
International Conference on Innovative Computing Technology	1
International Conference on Smart Computing	1
International Conference on Software Engineering	2
International Conference on System Sciences	3
International Conference on Ubiquitous and Future Networks	1
International Congress on Big Data	2
International Enterprise Distributed Object Computing Conference Workshops	1
International Symposium on Computer Science and Software Engineering	1
International Symposium on Software Reliability Engineering	1
International Workshop on BIG Data Software Engineering	11
International Workshop on Modeling in Software Engineering	1
International Workshop on Quality-Aware DevOps	3
International Workshop on the Future of Software Architecture Design Assistants	1
International Workshop on Real-Time and Distributed Computing in Emerging Applications	1
Journal of Cloud Computing: Advances, Systems and Applications	1
Journal of Future Generation Computer Systems	1
Software Engineering Conference	1
Software: Practice and Experience	1
The Journal of Supercomputing	1
Transactions on Big Data	1
World Congress On Services	1

APÊNDICE C – LISTA DE REQUISITOS DE FERRAMENTA PARA GERENCIAMENTO DE MAPEAMENTOS SISTEMÁTICOS

Requisitos funcionais:

- 1 - A ferramenta deve permitir o cadastro e gerenciamento de mapeamentos, estudos primários, autores, revisões e revisores.
- 2 - A ferramenta deve permitir, para cada mapeamento, o gerenciamento de estudo primários que possam ser incluídos no mapeamento. Isso se dará por meio de revisões de estudo primários por revisores do mapeamento (os pesquisadores conduzindo a pesquisa ou mesmo pesquisadores convidados).
- 3 - A ferramenta deve permitir a busca e visualização de mapeamentos, estudo primários, autores, revisões e revisores.
- 4 - A ferramenta deve permitir o cadastro de dimensões para a classificação dos resultados. Mapeamentos possuem diferentes objetivos e natureza de pesquisa, desta forma, é inviável determinar dimensões de pesquisa padronizadas. Algumas dimensões estarão de fato presentes em quase todos os mapeamentos, como tipo do estudo e estudo empírico, mas outros, como linguagens de programação utilizadas e fase do ciclo de vida de desenvolvimento, podem não estar presentes.
- 5 - A ferramenta deve possuir a funcionalidade de construção de indicadores customizados, baseados nas dimensões determinadas no mapeamento. Os indicadores serão base para os gráficos que serão apresentados no mapeamento.
- 6 - Qualquer usuário pode realizar o cadastro na ferramenta. Entretanto, um usuário administrador deverá conceder as permissões devidas ao mesmo.

Níveis de permissão:

- Administrador (controle total)
- Gerente do mapeamento (Pode alterar informações do mapeamento e seus estudo primários)
- Revisor de estudo primário (Pode revisar um estudo primário para um devido mapeamento)

Entretanto, qualquer usuário poderá buscar e visualizar informações de outros mapeamentos e estudo primários, mesmo sem permissão de edição do(s) mesmo(s).

Requisitos não funcionais

- 1 - A transição de telas não deve ultrapassar 1 segundo de processamento.
- 2 - A ferramenta deve ser responsiva de acordo com o dispositivo onde se dá o acesso.

APÊNDICE D – ENDEREÇO PARA ACESSO A PLANILHA DE CONTROLE UTILIZADA NO MAPEAMENTO

<https://github.com/rnlaigner/tcc/blob/master/Lista%20consolidada%20de%20artigos%20-%20Applying%20general%20Technique%20and%20Objective.xlsx>