

Capítulo 3

Mapeamento Sistemático da Literatura

3.1 Introdução

Neste capítulo apresentamos um Mapeamento Sistemático com o objetivo de identificar estudos que propõem melhorias das funcionalidades fornecidas pelas FGRMs. A partir de um conjunto de 64 artigos realizamos a divisão em dois grupos pela pertinência do estudo com as seguintes categorias: *(i) dimensões de melhoria*, conforme proposto por Zimmermann e outros [Zimmermann et al., 2009a] e *(ii) função desempenhada* no processo de manutenção de software, a partir de um conjunto de papéis discutidos por Polo e outros [Polo et al., 1999b].

A principal contribuição deste mapeamento é uma visão abrangente do estado da arte sobre propostas de melhorias das funcionalidades das FGRMs, com foco especial na gestão das RMs. Nossa expectativa é que pesquisadores possam encontrar questões para pesquisa e que profissionais com interesses em FGRMs possam obter um material de suporte às diversas questões deste domínio. Além disso, os responsáveis pelo desenvolvimento de FGRMs podem incorporar alguns dos achados deste estudo nas funcionalidades oferecidas pelo sistema do qual é responsável.

Este capítulo está organizado conforme descrito a seguir. Na Seção 3.2 descrevemos a metodologia de adotada através das perguntas direcionadoras, dos critérios para seleção dos estudos e dos esquemas de classificação utilizados. Na Seção 3.3 apresentamos o resultado do processo de classificar os artigos que fizeram parte do mapeamento. Uma discussão dos resultados é feita na Seção 3.4. As ameaças à validade e os trabalhos relacionados são discutidas nas Seções 3.5 e 3.6, respectivamente. Um resumo do

capítulo é feito na Seção 3.7.

3.2 Metodologia de Pesquisa

Um *Mapeamento Sistemático da Literatura*, também conhecido como Estudo de Escopo (Scoping Studies), tem como objetivo fornecer uma visão geral de determinada área de pesquisa, estabelecer a existência de evidências de estudos sobre tema de interesse e fornecer uma indicação da quantidade de trabalhos na linha de pesquisa sob análise [Keele, 2007, Wohlin et al., 2012]. Nesta dissertação empregamos as diretrizes propostas por Petersen e outros [Petersen et al., 2008] em que um conjunto de questões de pesquisa é utilizado para guiar a busca e seleção dos estudos primários. Em seguida, foram construídos esquemas de classificação com base nos dados extraídos dos artigos. Por fim, foi realizada uma análise para posicionar os trabalhos em seus respectivos esquemas. A estrutura desta seção está de acordo com o processo descrito por *Petersen e outros*, de modo que cada subseção representa uma das etapas propostas pelos autores.

3.2.1 Questões de Pesquisa

O objetivo deste mapeamento sistemático é identificar novas funcionalidades e melhorias das existentes que estão sendo propostas na literatura para as FGRMs. Deste modo, foram definidas as seguintes questões de pesquisa:

- **Questão 01:** *Quais as melhorias e novas funcionalidades estão sendo propostas para as FGRM?*
- **Questão 02:** *Quais papéis envolvidos no processo de manutenção de software as melhorias das funcionalidades visam dar suporte?*

Na *Questão de Pesquisa 01* estamos interessados em entender como a literatura da área vem propondo melhorias ou apresentando novos comportamentos para as FGRMs. Estas melhorias devem estar relacionadas com os problemas relacionados à gestão das RMs, conforme discutido na Seção 2.1.4.3. O intuito é entender as técnicas e abordagens adotadas nas soluções propostas. Na *Questão de Pesquisa 02* o objetivo é descobrir como os diferentes tipos de papéis que fazem parte do processo de manutenção de software estão recebendo suporte pelos estudos da área.

3.2.2 Pesquisa da Literatura

Para encontrar o conjunto de estudos mais relevantes, bem como eliminar aqueles que não permitam responder as questões de pesquisas, adotamos os seguintes critérios para inclusão ou exclusão de artigos no mapeamento:

- Critérios de Inclusão
 - Artigos publicados em conferências e periódicos (journals)
 - Estudos publicados a partir de 2010¹
 - Artigos escritos em língua inglesa
 - Artigos disponíveis com texto completo
- Critérios de Exclusão
 - Livros e literatura cinza (gray literature)
 - Artigos que não possuem relação com FGRM
 - Estudos duplicados, neste caso foi considerada a versão mais completa do trabalho

Os estudos primários foram coletados mediante a aplicação de sentenças de buscas nas seguintes bibliotecas digitais: *IEEE Explore*, *ACM Digital Library*, *Scopus*, e *Inspec/Compendex*. No estudo descrito por Dyba e outros [Dybå et al., 2007] verifica-se que o uso de apenas algumas bibliotecas apresenta um resultado semelhante que uma configuração quase exaustiva de base de dados. Neste sentido, tomando como base o trabalho de Dyba e outros, o conjunto de bibliotecas digitais utilizadas neste mapeamento nos permite encontrar estudos primários em quantidade satisfatória. As sentenças de buscas foram produzidas com base na metodologia PICO (Population, Intervention, Comparison and Outcomes) que é sugerida por Kitchenham e Charters [Keele, 2007] para ajudar pesquisadores na formulação de termos tomando como ponto de partida as questões de pesquisa.

Após uma busca automatizada nas base de dados chegamos a um total de 286 artigos. A Tabela 3.1 exibe o total de estudos recuperados por biblioteca digital. Os trabalhos coletados foram avaliados, através da ferramenta *JabRef*², em busca de possíveis duplicados. Esta etapa resultou na exclusão de 81 artigos, desta forma chegamos

¹Foram considerados neste estudo artigos publicados até maio/2016, data de realização da pesquisa nas base de dados.

²<https://www.jabref.org/>

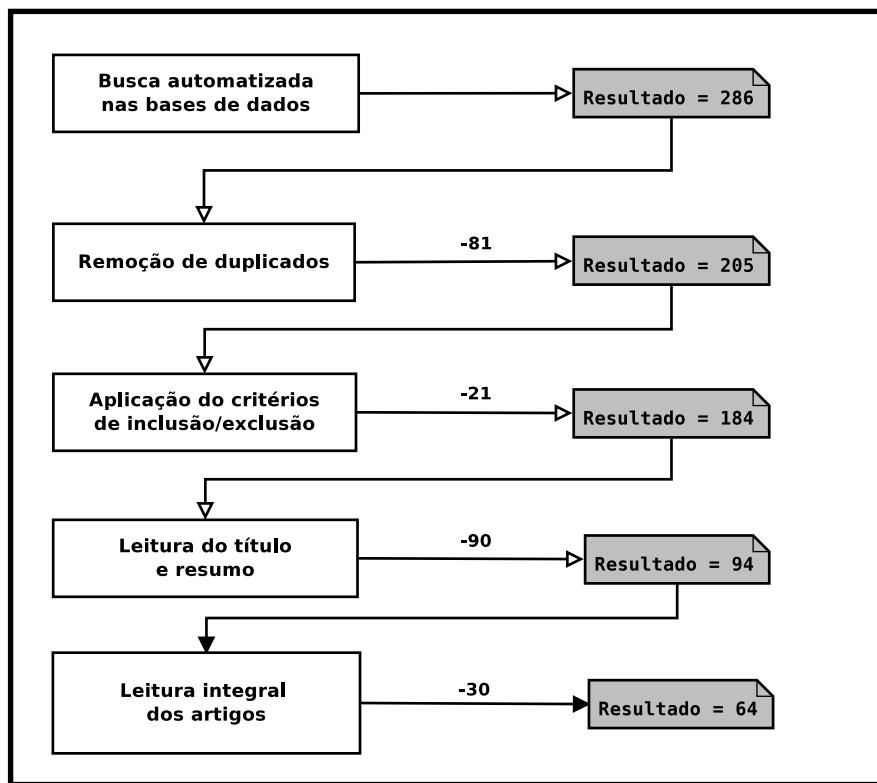


Figura 3.1: Número de artigos incluídos durante o processo de seleção dos estudos. Baseado em [Petersen et al., 2015]

a 205 estudos ao final do processo. Finalmente os trabalhos foram analisados com base na leitura do título e resumo. Nos casos em que o título e resumo não eram capazes de caracterizar o trabalho foi realizada uma leitura completa do texto. O processo descrito resultou em **64** estudos. A Figura 3.1 resume a seleção dos estudos primários através da exibição do número de artigos em cada etapa.

Base de Dados	Total
ACM Digital Library	109
IEEE Explore	100
Inspec/Compendex	22
Scopus	55

Tabela 3.1: Número de Estudos Recuperados por Base de Dados

3.2.3 Esquemas de Classificação

O mapeamento foi conduzido utilizando dois esquemas de classificação. O primeiro organiza os artigos pela pertinência com a dimensão de melhoria que a funcionalidade

proposta pertence. As dimensões de melhorias foram baseadas no trabalho de Zimmermann e outros cujo objetivo é aperfeiçoar as funcionalidades das FGRMs de maneira integral [Zimmermann et al., 2009a]. A segunda classificação distribui os estudos pelo relacionamento com o suporte dado à determinado papel no processo de manutenção de software. Entendemos que estes dois esquemas nos fornecem uma visão de como as melhorias das funcionalidades vêm sendo propostas tanto do ponto de vista de quem desenvolve quanto das diferentes partes interessadas envolvidas nos projetos de software. As próximas subseções discutem em maior detalhe cada esquema.

3.2.3.1 Classificação por Dimensão de Melhoria

Em um estudo sobre o aperfeiçoamento das FGRMs [Zimmermann et al., 2009a], os autores argumentam que ter informações completas nos relatos de falhas (Requisição de Mudança), tão logo quanto possível, ajuda os desenvolvedores a resolver com mais rapidez o problema. Neste mesmo estudo, eles discutem como melhorar as funcionalidades oferecidas pelas FGRMs de forma integral, ou seja, que atenda aos diversos contextos em que este tipo software está integrado. As dimensões de melhorias que eles propuseram estão descritas a seguir e são exibidas na Figura 1.3.

Foco na Informação Estas melhorias focam diretamente na informação fornecida pelo reportador da RM. Com ajuda da FGRM, o responsável por descrever uma falha, por exemplo, poderia ser motivado a coletar mais informações sobre o problema. O sistema poderia verificar a validade e consistência do que foi repassado pelo Reportador (detalhes sobre este papel pode ser encontrado na Subseção 2.1.3).

Foco no Processo Melhorias com foco no processo visam dar suporte às atividades de administração focadas na solução das RMs. Por exemplo, a triagem de RM, poderia ser automatizada visando acelerar o processo. Um outro exemplo de melhoria poderia ocorrer no aumento do entendimento do progresso realizado em cada RM ou mesmo fornecer ao usuário afetado por uma falha a estimativa do tempo necessário para atendimento (estimativa de esforço).

Foco no Usuário Nesta dimensão estão incluídos tanto os usuários que relatam as RMs (Reportadores) quanto os desenvolvedores responsáveis por solucioná-las. Os reportadores podem ser orientados de qual informação fornecer e como coletá-la. Os desenvolvedores também podem se beneficiar de um treinamento sobre qual informação esperar e como esta informação pode ser utilizada para solucionar uma RM.

Foco na Ferramenta As melhorias centradas na ferramenta são discutidas com vistas às funcionalidades das FGRMs. Elas podem reduzir a complexidade da coleta e fornecimento das informações necessárias para solucionar a RM. Por exemplo, as FGRMs poderiam ser configuradas para automaticamente identificar a cadeia de registros de ativação de funções (stack trace) e adicioná-la ao erro reportado. A ferramenta poderia ainda simplificar o processo de reprodução do erro mediante a captura automatizada de tela (screenshots).

Para a classificação dos estudos foi realizado um processo baseado no trabalho de Petersen e outros [Petersen et al., 2008], que é composto por duas etapas:

- I análise das palavras-chaves e conceitos que identificam as contribuições do estudo por meio da análise do título e resumo.
- II combinações das palavras-chaves para construir um conjunto de categorias para classificação dos artigos.

Os autores recomendam que nos casos em que o resumo e o título do estudo não sejam capazes de caracterizá-lo, as seções de introdução e conclusão também devem ser analisadas. Para as bases de dados onde era informado mais de um conjunto de palavras-chaves para um mesmo artigo, utilizamos aquelas que foram informadas pelos autores. Mediante a aplicação do processo descrito foi construído o esquema de classificação apresentado na Figura 3.2.

3.2.3.2 Classificação por Suporte ao Papel da Manutenção de Software

Neste esquema de classificação estamos interessado em avaliar como os estudos dão suporte aos papéis apresentados na Subseção 2.1.3. A construção da classificação utiliza a mesma metodologia descrita na seção anterior.

3.3 Resultados

Nesta Seção apresentamos para cada item de classificação os estudos relacionados. Iniciamos com uma análise da frequência de publicação sobre o tema do mapeamento e posteriormente apresentamos os resultados para cada uma das dimensões de melhoria. Seguimos com a análise dos estudos pelo papel ao qual a funcionalidade proposta visa dar suporte.

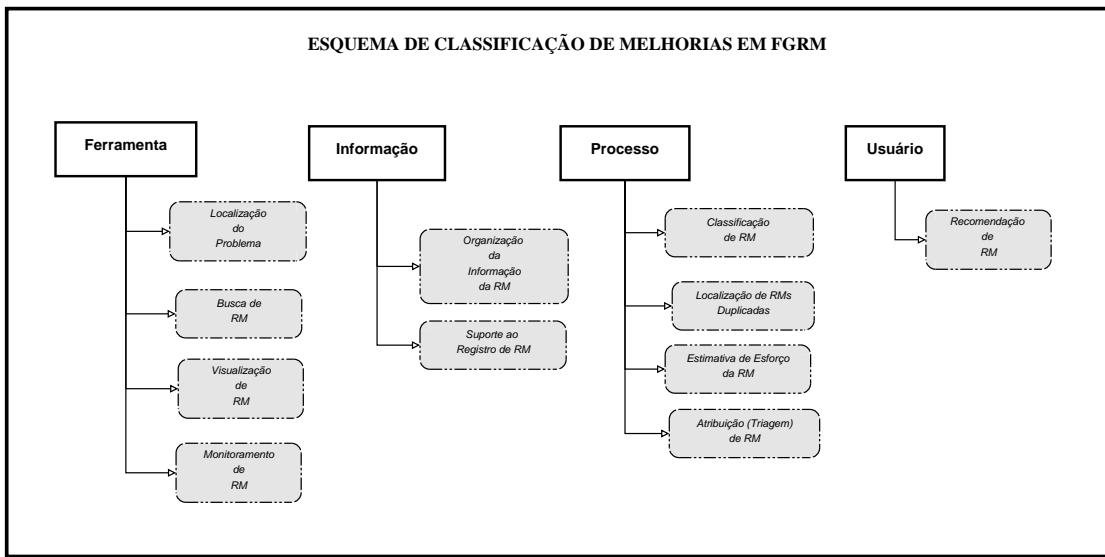


Figura 3.2: Esquema de classificação das melhorias propostas na literatura. Os retângulos representam as dimensões de melhorias e os polígonos de cantos arredondados representam tópicos de problemas da gestão das RMs.

3.3.1 Frequência das Publicações

A Tabela 3.2 exibe o número de estudos primários identificados entre os anos de 2010 e 2016, período de referência utilizado no mapeamento. Dentro dos estudos escolhidos, verificamos que em 2010 foram publicados cinco estudos sobre o assunto [Sun et al., 2010, Gegick et al., 2010, Song et al., 2010a, Nagwani & Verma, 2010a, Zimmermann et al., 2010]. Posteriormente constatamos um acréscimo no número de estudos no qual o maior aumento pode ser observado entre os anos de 2012-2014.

Ano	Frequência
2010	5
2011	8
2012	14
2013	12
2014	13
2015	9
2016	3

Tabela 3.2: Número de estudos primários por ano de publicação.

3.3.2 Classificação por Dimensões de Melhoria

Nesta seção apresentamos estudos classificados como pertinentes com a dimensão denominada “Melhoria de funcionalidades de FGRM”. A classificação está estruturada de modo que no primeiro nível temos uma das quatro dimensões de melhorias discutidas na Subseção 3.2.3.1. O segundo nível é composto por tópicos que são os problemas da gestão das RMs o qual a melhoria está relacionada. Uma discussão mais detalhada sobre este problemas pode ser encontrado na Subseção 2.1.4.3. A Tabela 3.3 exibe a distribuição dos estudos pela dimensão de melhoria e o seu respectivo tópico.

Tabela 3.3: Lista de artigos de acordo com o esquema de classificação

Dimensão de Melhoria	Tópico	Estudos	Total
Processo	Localização de RMs Duplicadas	[Alipour et al., 2013, Banerjee et al., 2012, Hindle et al., 2016, Koopaei & Hamou-Lhadj, 2015] [White et al., 2015b, Prifti et al., 2011, Song et al., 2010b, Sun et al., 2010, Sun et al., 2011] [Sun et al., 2011, Thung et al., 2014a, Tian et al., 2012a, Tomašev et al., 2013, Lerch & Mezini, 2013]	13
Processo	Atribuição (Triagem) de RM	[Banitaan & Alenezi, 2013, Hosseini et al., 2012, Hu et al., 2014, Naguib et al., 2013] [Nagwani & Verma, 2012, Shokripour et al., 2012, Tian et al., 2015, Valdivia Garcia & Shihab, 2014] [Wu et al., 2011, Xuan et al., 2012, Zanetti et al., 2013, Zhang et al., 2014]	12
Processo	Classificação de RM	[Behl et al., 2014, Chawla & Singh, 2015, Gegick et al., 2010, Izquierdo et al., 2015] [Kochhar et al., 2014, Nagwani et al., 2013, Netto et al., 2010] [Somasundaram & Murphy, 2012, Tian et al., 2013, Zhang & Lee, 2011]	10
Ferramenta	Localização do Problema	[Bangcharoensap et al., 2012, Corley et al., 2011, Nguyen et al., 2012] [Thung et al., 2014c, Wong et al., 2014] [Romo & Capiluppi, 2015, Thung et al., 2013]	7
Informação	Suporte ao Registro da RM	[Bettenburg et al., 2008a, Correa et al., 2013, Moran et al., 2015, Moran, 2015] [Tu & Zhang, 2014, White et al., 2015a, Kaiser & Passonneau, 2011]	7
Processo	Estima de Esforço da RM	[Bhattacharya & Neamtiu, 2011, Nagwani & Verma, 2010b, Thung et al., 2012b] [Vijayakumar & Bhuvaneswari, 2014, Xia et al., 2015]	5
Ferramenta	Visualização de RM	[Dal Sasso & Lanza, 2013, Dal Sasso & Lanza, 2014, Hora et al., 2012, Takama & Kurosawa, 2013]	4
Informação	Organização da Informação da RM	[Mani et al., 2012, Otoom et al., 2016]	2
Usuário	Recomendação de RM	[Malheiros et al., 2012, Wang & Sarma, 2011]	2
Ferramenta	Busca de RM	[Liu & Tan, 2014]	1
Ferramenta	Monitoramento de RM	[Aggarwal et al., 2014]	1

3.3.2.1 Melhorias Propostas na Dimensão Ferramenta

Este ramo do esquema de classificação inclui os estudo que possuem relação com tópicos como Localização do Problema e Visualização de RMs.

Localização do Problema: Os estudos incluídos neste tópico focam em localizar a origem de um problema de software com base nos dados da RM [Hovemeyer & Pugh, 2004]. Com objetivo de melhorar a eficiência da Localização do Problema diversas informações contidas nas RMs estão sendo utilizadas. As abordagens propostas utilizam informações como cadeia de registros de ativação (stack-trace) [Wong et al., 2014], descrição e campos estruturados das RMs [Thung et al., 2014c] e os históricos de versões do código fonte do sistema [Bangcharoensap et al., 2012, Corley et al., 2011, Romo & Capiluppi, 2015]. Algumas das melhorias propostas foram incluídas em ferramentas largamente em-

pregadas no mercado utilizando as propriedades de extensão que elas oferecem [Thung et al., 2014c, Corley et al., 2011].

Visualização de RM: Os estudos neste tópico estão relacionados com melhoria da visualização da informação contida na RM. Com o objetivo de apresentar diferentes formas de visualizar os dados de uma RM novos conceitos estão sendo propostos. No estudo de Hora e outros [Hora et al., 2012] é apresentado o conceito de Mapas de Defeitos (Bugs Maps) que mostra a RM e uma ligação dela com outros artefatos de software, como por exemplo o histórico de versões. No estudo de Lanza e Dal Sasso [Dal Sasso & Lanza, 2014] o conceito de hiperligação entre documentos é utilizado para permitir a navegação entre os artefatos que estão relacionados a uma RM.

Verificamos ainda no artigo proposto por Takama e Kurosawa [Takama & Kurosawa, 2013] a aplicação de tecnologias de visualização de informação empregada para o monitoramento das informações contidas nas RMs. Uma FGRM atualiza quase toda informação de uma RM como texto, sem maiores estruturações. A solução proposta pelo autores visa suportar o monitoramento das RMs apresentando ao usuário da FGRM, seja ele o afetado por uma falha, o responsável por relatar a requisição ou mesmo um membro da equipe de manutenção, mediante animações produzidas com base nas atualizações ocorridas na RM.

Por conta da natureza das melhorias propostas neste tópico de pesquisa, verificamos que diversos estudos foram prototipados. Desta forma, é possível avaliar as propostas contidas neste tópico através das ferramentas como o bugMaps [Hora et al., 2012] e In* Bug [Dal Sasso & Lanza, 2014].

3.3.2.2 Melhorias Propostas na Dimensão Informação

Apresentamos os trabalhos relacionados à melhoria da qualidade da informação de uma RM. A melhoria pode envolver suporte ao registro de uma RM antes que ela seja armazenada na base de dados de uma FGRM; a melhoria também pode envolver a organização da informação contida no relato de uma RM para facilitar o entendimento pelos desenvolvedores e demais profissionais envolvidos na manutenção de software.

Suporte ao Registro da RM: A pesquisa visando a melhoria da qualidade da informação fornecida nas RMs tem como premissa estimar uma medida relacionada a alguma métrica com base no texto contido no relato da RM. A determinação do que seria um boa descrição de um problema de software foi obtida mediante uma pesquisa com profissionais de manutenção de software no estudo de Betten-

burg e outros [Bettenburg et al., 2008a]. Em outro estudo os próprios autores definiram as métricas que posteriormente foram utilizadas para avaliar o relato da RM [Tu & Zhang, 2014].

Um segundo nicho de estudos visa suportar a reprodução da falha do software. Estes estudos incluem tanto registrar o conjunto de ações que resultaram na falha [White et al., 2015a], quanto em autocompletar o texto que compõe o relato de um erro [Moran et al., 2015]. Um ponto em comum destes dois estudos é que eles foram pensados para o ambiente de desenvolvimento de aplicações para dispositivos móveis, mais especificamente para dispositivos baseados no sistema Android³. Uma possível justificativa para o foco em aplicações móveis pode ser a dificuldade em registrar uma falha neste tipo de ambiente [White et al., 2015a, Moran et al., 2015].

Muitos dos estudos resultaram em ferramentas com a finalidade de realizar uma prova de conceito sobre como ajudar ao Reportador a fornecer um relato de boa qualidade [Tu & Zhang, 2014, Bettenburg et al., 2008a, Kaiser & Passonneau, 2011, White et al., 2015a, Moran et al., 2015].

Organização da Informação da RM: Em diversas situações o desenvolvedor precisa examinar manualmente o relato das RMs que podem variar em tamanho e complexidade [Mani et al., 2012]. Neste contexto, o resumo (sumarização) automático do texto contido em uma RM é uma maneira de reduzir a quantidade de dados que o desenvolvedor precisa analisar. A ferramenta denominada AUSUM [Mani et al., 2012] propõe uma abordagem, utilizando técnicas não supervisionadas de aprendizado de máquina, para aprender e sumarizar o conjunto de relatos.

3.3.2.3 Melhorias Propostas na Dimensão Processo

Identificação de RMs Duplicadas O processo de identificação de RMs duplicadas consiste em avaliar se determinado relato já foi realizado em algum outro momento. A abordagem adotada da literatura para tratar o problema pode ser dividida em dois tipos[Kaushik & Tahvildari, 2012, Tian et al., 2012b]:

- (i) remoção de duplicatas
- (ii) identificação de duplicatas

No primeiro tipo, o objetivo é evitar que RMs duplicadas entrem na base de dados de uma FGRM e, desta forma, evitar o esforço e o tempo extra necessário para

³<https://www.android.com/>

identificá-la posteriormente. Por outro lado, no segundo tipo o objetivo é sugerir uma lista de possíveis duplicatas durante o processo de registro de uma nova RM. Um ponto importante é que o segundo tipo se baseia na premissa que registrar um mesmo problema por mais de uma vez nem sempre é problemático tendo em vista que pode fornecer informações úteis [Bettenburg et al., 2008c]. É importante que novas abordagens tentem equilibrar estes dois tipos de tratamento para evitar o tempo extra para análise de uma RM bem como apoiar os desenvolvedores com informações adicionais [Lerch & Mezini, 2013, Thung et al., 2014a].

Uma forma de tratar o problema é utilizar modelos de espaços vetoriais para medir a similaridade entre as RMs [Liu & Tan, 2014, Sun et al., 2010, Thung et al., 2014a, Tomašev et al., 2013]. Outros trabalhos tentam utilizar técnicas em que os termos específicos do domínio do projeto de software, contidos no relato das RMs, são essenciais na determinação da probabilidade de que duas requisições são duplicadas [Hindle et al., 2016, Alipour et al., 2013]. Em resumo verificamos que o relato contido na RM é utilizado como fonte primária de informação para técnicas de Recuperação da Informação visando determinar a similaridade entre duas RMs.

Atribuição (Triagem) de RM: Os estudos apresentados neste tópicos foram classificados pela pertinência com a automatização do processo de encontrar o desenvolvedor mais apto para solucionar determinada RM. A principal fonte de informação utilizada na abordagens propostas é o relato contido na RM, todavia, outros dados são utilizados, tais como nível de prioridade [Tian et al., 2015], registros (log) do sistema de controle de versão [Shokripour et al., 2012, Hu et al., 2014] e itens do contexto do projeto como por exemplo: o perfil e preferências do desenvolvedor, quantidade de RMs atribuídas e tempo estimado de correção [Hosseini et al., 2012]. Em outros trabalhos verificamos que é explorada a característica colaborativa que existe no processo de triagem das RMs. Naqueles estudos é desenvolvida uma “rede de colaboração” que possibilita determinar o desenvolvedor mais apto [Zhang et al., 2014, Zanetti et al., 2013, Wu et al., 2011].

Em geral técnicas de Recuperação da Informação são utilizadas pelo estudos que compõem este tópico. Dentre elas podemos destacar a utilização de modelos de espaço vetorial e ranqueamento de páginas. Nos estudos em que o caráter colaborativo é explorado, verificamos a prevalência de métodos de classificação do tipo K-Nearest-Neighbor.

Classificação da RM: Os estudos neste tópico visam automatizar o processo de classificação de uma RM. Este tipo de abordagem faz uso da funcionalidade de atribuição

de rótulo (labels) que é comum a grande parte das FGRMs (vide Subseção 2.3.4.3). Em muitos projetos esta atividade é realizada manualmente pelo Agente de Triagem, Desenvolvedor ou Analista de Qualidade, o que pode resultar em classificações equivocadas.

Esta classificação pode ser realizada pelo tipo de manutenção (Corretiva, Adaptativa, Perfectiva e Preventiva), se a RM trata de questões relativas à segurança do sistema [Gegick et al., 2010, Behl et al., 2014] ou pelo nível de prioridade que a RM deve ser analisada [Behl et al., 2014].

Estimativa de Esforço da RM: Identificamos três tipos de estimativas de esforço relacionadas a uma RM: determinar o tempo para solucionar novas RMs; definir os artefatos que são impactados por uma RM; prever o número de novas RMs que poderão fazer parte do projeto.

Nos estudos que tratam da primeira forma de estimativa a preocupação é o tempo necessário para tratar a mudança solicitada em determinada requisição. A principal complexidade está em produzir uma estimativa precisa em função das diferentes atividades envolvidas e dos diferentes níveis de capacitação do responsável pela execução das tarefas [Xia et al., 2015]. No segundo grupo temos os artigos que tentam identificar previamente o conjunto de artefatos que serão impactados pela tarefa de manutenção [Nagwani & Verma, 2010b]. Neste mapeamento o foco foi em estudos onde as RMs são o ponto de partida para a análise de impacto. O último grupo de estudos discute técnicas sobre como prever o número de RMs que possivelmente serão relatadas em futuras versões do sistema. A predição do que será relatado inclui RMs que não existiam em versões anteriores como aquelas que serão reabertas, ou seja, problemas que não foram solucionados previamente mesmo as suas RMs dizendo o contrário [Xia et al., 2015].

3.3.2.4 Melhorias Propostas na Dimensão Usuário

Recomendação de RM: Os estudos contidos neste tópico dão suporte aos desenvolvedores com pouca experiência no projeto correspondente mediante a redução da curva de aprendizagem. Para facilitar a inclusão de novos desenvolvedores alguns estudos tratam do desenvolvimento de sistemas de recomendação de RMs [Malheiros et al., 2012, Wang & Sarma, 2011]. Estes sistemas podem ajudar o recém-chegado a solucionar uma RM mediante a apresentação do código fonte potencialmente relevante que o ajudará na solução do problema [Malheiros et al., 2012].

O segundo tipo de abordagem pode ser vista como ambiente de exploração do repositório de RMs. Esta funcionalidade permite que novos desenvolvedores pesquisem descrições das requisições que possam ser do seu interesse bem como dos artefatos relativos àquela RM (por exemplo, arquivos relacionados, desenvolvedores contribuintes, registros de comunicação) [Wang & Sarma, 2011].

Com base nos estudos que compõem esta categoria, verificamos que modelos de IR vêm sendo utilizados para possibilitar a recomendação de RM. Neste contexto, técnicas bem conhecidas na literatura tais como VSM [Wang & Sarma, 2011] e o modelo estatístico PPM [Malheiros et al., 2012].

3.3.3 Suporte à Papéis da Manutenção de Software

Nesta subseção são discutidos os estudos que foram considerados relacionados com papéis discutidos na Subseção 2.1.3, o qual é dado suporte. A Tabela 3.4 exibe o total de artigos por papel que a funcionalidade proposta visa dar suporte. Como pode ser observado verificamos um maior número de estudos para os papéis de Agente de Triagem e Desenvolvedor. Ainda é possível observar a prevalência de estudos nos tópicos “Localização de RMs Duplicadas” e “Atribuição [Triagem] de RM”, o que é natural tendo em vista que há um mapeamento entre o papel desempenhado na manutenção com as atividades executadas por aquele papel.

Papel	Total de Artigos
Agente de Triagem	37
Desenvolvedor	26
Analista de Qualidade	13
Gerente de Requisição de Mudança	11
Reportador	6
Líder da Manutenção	4
Todos	3

Tabela 3.4: Total de artigos por papel na manutenção de software

Agente de Triagem: Esta função tem como principal objetivo a atribuição das RMs para o desenvolvedor mais apto [Banitaan & Alenezi, 2013]. Neste mapeamento, os estudos recuperados focam em apresentar soluções de atribuição automática [Banitaan & Alenezi, 2013, Shokripour et al., 2012, Somasundaram & Murphy, 2012, Naguib et al., 2013, Zhang et al., 2014, Zanetti et al., 2013]; classificação automatizada [Gegick et al., 2010, Liu & Tan, 2014, Behl et al., 2014, Chawla & Singh, 2015, Tian et al., 2015]; visualização da

fila de RMs [Izquierdo et al., 2015]; agrupamento (clustering) das requisições [Liu & Tan, 2014]; identificação do tempo necessário para solucionar a RM (time to fix) [Hosseini et al., 2012, Bhattacharya & Neamtiu, 2011]; sumarização das informações contidas na RM [Mani et al., 2012]; determinação de RMs duplicadas [Sun et al., 2011, Kaiser & Passonneau, 2011].

Desenvolvedor: Apresentamos aqui os trabalhos com foco em aspectos de codificação, depuração e testes. No suporte ao desenvolvedor identificamos estudos que propõem à atribuição de RMs a um conjunto de desenvolvedores, em contraposição da tradicional atribuição a um único programador [Banitaan & Alenezi, 2013], visando minimizar os problemas decorrentes da propriedade de código e proporcionar um maior nivelamento de informações entre os membros da equipe. Não obstante, o maior grupo de estudos nesta categoria está relacionado com a ajuda ao desenvolvedor na vinculação de determinado problema do software à sua efetiva origem, que nesta dissertação foi denominado como Localização do Problema [Corley et al., 2011, Wong et al., 2014, Thung et al., 2014c, Nguyen et al., 2012, Thung et al., 2013, Romo & Capiluppi, 2015]. Nesta mesma categoria verificamos estudos que dão suporte ao desenvolvedor em classificar a RM que lhe foi atribuída, em especial aquelas que estão relacionadas às questões de segurança do sistema [Gegick et al., 2010] ou aquelas RMs que impendem a resolução de outras (blocking-bugs) [Valdivia Garcia & Shihab, 2014].

Analista de Qualidade: Cabe ao Analista de qualidade avaliar se uma RM definida como solucionada foi corretamente resolvida. De maneira similar ao que ocorre nos estudos que estão relacionados com o tópico de classificação “Desenvolvedor” verificamos uma prevalência dos estudos que visam determinar uma ligação entre um problema de software e o código fonte [Corley et al., 2011, Wong et al., 2014, Thung et al., 2014c, Nguyen et al., 2012, Thung et al., 2013, Romo & Capiluppi, 2015]. Verificamos ainda estudos que tentam predizer a probabilidade que determinada RM será reaberta [Xia et al., 2015], o que pode ajudar ao Analista de Qualidade na priorização das requisições com alta possibilidade de retorno.

Gerente de Requisição de Mudança: O papel que representa esta classe está vinculado à gestão do processo de manutenção de software, em especial por decidir se uma RM será aceita ou rejeitada. Neste contexto, melhorias relacionadas à classificação quanto ao nível de segurança [Gegick et al., 2010, Zhang & Lee, 2011, Valdivia Garcia & Shihab, 2014], identificação de duplicadas [Hindle et al., 2016,

Sun et al., 2010, Alipour et al., 2013, Banerjee et al., 2012] pode ajudar no desempenho desta atividade.

Reportador: Os estudos que fazem parte desta categoria partem da premissa que melhorar a qualidade da informação fornecida na RM é o ponto de partida para tratar outros problemas relacionados ao processo de manutenção de software [Moran et al., 2015, Moran, 2015, Bettenburg et al., 2008a]. Neste sentido verificamos trabalhos para autocompletar as informações fornecidas pelo Reportador [Moran et al., 2015], suporte à reprodução do problema [Moran, 2015]; análise da qualidade da informação fornecida [Bettenburg et al., 2008a, Tu & Zhang, 2014]. Esta categoria também contempla um estudo que visa detectar se o problema relatado por uma RM corresponde ao problema relatado por outra que já foi registrada [Thung et al., 2014a].

Chefe da Manutenção: O Chefe da Manutenção tem por responsabilidade definir os padrões e procedimentos que compõem o processo de manutenção que será utilizado. Para ajudar nesta tarefa alguns estudos propõem melhorar a alocação de tarefas do processo de resolução das Requisições de Mudanças [Netto et al., 2010]. Outros estudos visam estimar o esforço necessário para solucionar determinada RM [Vijayakumar & Bhuvaneswari, 2014, Nagwani & Verma, 2010b], estes aspectos têm o potencial de ajudar o Chefe de Manutenção no planejamento de liberações de novas versões do sistema que está sendo mantido.

Todos Esta categoria abrange os estudos para o qual a melhoria proposta possui impacto positivo para todos os papéis envolvidos na manutenção de software. A definição que o foco da melhoria é geral decorre do que foi dito como objetivo dos autores dos estudos que fazem parte desta categoria ou ainda por não ser possível determinar uma atividade específica sendo beneficiada.

Conforme pode ser observado, os estudos estão relacionados principalmente com a melhoria da visualização das informações contidas nas RMs [Hora et al., 2012, Takama & Kurosawa, 2013, Dal Sasso & Lanza, 2014]. Os aperfeiçoamentos podem estar vinculados a questões de usabilidade das ferramentas, como por exemplo a navegaabilidade entre as RMs [Dal Sasso & Lanza, 2014].

3.4 Discussão

Ao realizarmos este mapeamento verificamos uma prevalência de estudos na dimensão *Processo* especialmente para os tópicos de *Localização de RMs Duplicadas, Atribuição*

(*Triagem*) de RM e *Classificação de RM*, respectivamente. A prevalência destes tópicos pode estar relacionado ao fato de que eles afetam projetos de diferentes tipos e tamanhos. No caso da *Localização de RM Duplicados* apesar de ser o de maior frequência não é um problema identificado como o de impacto mais significativo por alguns desenvolvedores [Bettenburg et al., 2008a].

Dos estudos que fizeram parte do mapeamento um total de 10 foram implementados como extensões ou protótipos de uma FGRMs. No nosso entendimento este número poderia ser maior a fim de permitir avaliações pelos profissionais envolvidos em manutenção de software. Cabe ressaltar que no escopo de um estudo pode não estar prevista a efetiva transformação da melhoria proposta de modo a ser utilizada efetivamente pelo seu público-alvo, como por exemplo, a criação ou melhoria de uma funcionalidade em determinada FGRM. Ademais, não está no objetivo deste estudo avaliar ou discutir a facilidade que as FGRMs possuem para criar novas funcionalidades ou melhorias.

Contudo, o nosso entendimento é de que um maior número de melhorias propostas na literatura sendo utilizadas pelos profissionais envolvidos em manutenção de software poderia melhorar a qualidade das soluções mediante a redução da diferença entre o estado-da-prática com o estado-da-arte.

Quando analisamos o esquema de Classificação por Papéis, verificamos uma prevalência de estudos com foco no papel de *Agente de Triagem*. Existe possivelmente uma crença que é possível melhorar a produtividade do processo de manutenção de software reduzindo o esforço de encontrar o desenvolvedor mais apto. Os estudos que fazem parte desta classe destacam que um considerável conhecimento sobre o projeto é necessário bem como a capacidade de negociação com os desenvolvedores e demais partes interessadas são importantes para o desempenho do papel. Todavia, tendo em vista o esforço e tempo gasto por esta tarefa, especialmente quando realizada manualmente, seria importante que as FGRMs automatizassem algumas destas atividades. Um outro ponto a destacar é que as FGRMs deveriam dar suporte ao Reportador que, na maioria da vezes, é o primeiro a registrar as informações que serão necessárias à solução da RM.

3.5 Limitações e Ameaças à Validade

Alguns dos procedimentos adotados neste trabalho não acompanharam exatamente as diretrizes existente na literatura para condução de uma Mapeamento Sistemático. Um único investigador selecionou os estudos candidatos e este mesmo revisor teve a

responsabilidade de analisar o artigos que seriam incluídos ou excluídos.

O mapeamento realizado nesta pesquisa utilizou o método de aplicação de sentenças de busca nas bases de dados selecionadas para coletar os estudos primários. Outros estudos, além da estratégia descrita, fazem uso de uma técnica conhecida como “bola de neve” (snowballing) [Wohlin, 2014] em que as referências dos estudos primários podem ser usadas para compor o conjunto de artigos do mapeamento. Neste sentido, ao usarmos uma única estratégia podemos ter perdido artigos relevantes e, portanto, subestimar a extensão dos resultados encontrados. Em particular, por termos optado por escolher artigos apenas em língua inglesa também pode ter havido falta de material publicado em revistas e conferências nacionais. Assim, nossos resultados devem ser considerados apenas com base em artigos em inglês contidos nas bases de dados escolhidas e em especial publicados nas principais conferências da área de Engenharia de Software.

O fato de um único pesquisador ter sido o responsável pela análise dos estudos pode significar que alguns dos dados coletados podem ser errôneos. O processo de seleção e validação dos estudos primários pode levar a problemas de extração e agregação das informações quando há um grande número de artigos ou os dados são complexos [Keele, 2007]. No entanto, neste estudo secundário, houve poucos estudos primários e os dados recuperados eram razoavelmente objetivos. Desta forma, não esperamos erros de extração. Cabe ressaltar que apesar do processo de validação ter sido executado por um único pesquisador, os critérios de qualidade foram avaliados independentemente por dois pesquisadores, desta forma minimizando a inclusão de trabalhos cuja qualidade comprometa os resultados.

No tocante as questões deste estudo é possível que as perguntas de pesquisa definidas possam não abranger completamente o campo de investigação sobre as funcionalidades das FGRMs. No entanto, algumas discussões com membros do projeto e especialistas em Manutenção de Software foram realizadas para validar as perguntas. Assim, mesmo que não tenhamos considerado o melhor conjunto de questões, tentamos abordar as indagações mais frequentes e abertas no campo, tanto do ponto de vista do praticante como do investigador.

Como as bibliotecas digitais não funcionam com regras de pesquisa compatíveis entre si, todas as sequências de pesquisa foram adaptadas e calibradas em cada biblioteca. No entanto, não conhecemos todas as regras que as bibliotecas digitais utilizam para procurar um documento. Neste sentido, a forma que as sentenças de busca foram estruturadas pode não ser a mais otimizada para seleção do maior número de documentos relevantes para o estudo.

3.6 Trabalhos Relacionados

No estudo proposto por Kagdi e outros [Kagdi et al., 2012] foi realizada uma revisão da literatura sobre abordagens para mineração de repositórios de relatos de problema de software. No contexto daquele trabalho este tipo de repositório pode ser comparado à uma FGRM. O resultado foi uma taxonomia baseada em quatro classes: o tipo de repositório extraído (o que), o propósito (por que), o método proposto (como) e o método de avaliação (qualidade). No entanto, a sua classificação não fornece um entendimento extensivo sobre as investigações em repositórios de RM. De acordo com seus critérios de exclusão para estudos, eles estavam muito preocupados com artigos que abordavam mudanças evolutivas de artefatos de software investigando múltiplos repositórios de software. Como consequência, muitos trabalhos que usaram dados de um repositório único estavam além de seu escopo.

Por outro lado, o estudo realizado neste trabalho aumentou o escopo das funcionalidades oferecidas pelas FGRM possibilitando uma visão mais abrangente do estado da arte deste tipo de investigação. Uma outra diferença com o trabalho de Kagdi [Kagdi et al., 2012] é que sua taxonomia considera as técnicas e métodos para mineração de repositórios de software como o foco principal do seu estudo, por lado este trabalho considera as FGRM, sobre o prisma de suas funcionalidades, como entidades de primeira classe.

No estudo realizado por Cavalcanti e outros [Cavalcanti et al., 2014] houve a classificação de estudos sobre repositórios de RM em desafios e oportunidades. Desafios referem-se a problemas enfrentados na gestão das RMs, enquanto oportunidades referem-se às vantagens proporcionadas pelos dados obtidos das RMs para o desenvolvimento de software. Além disso os autores utilizam a classificação proposta por Canfora e Cerulo[Cerulo & Canfora, 2004]. Ela consiste em duas visões sobrepostas: uma taxonomia vertical que classifica os modelos de Recuperação da Informação (Information Retrieve - IR) com relação ao seu conjunto de características básicas; e uma taxonomia horizontal que classifica os objetos de IR com respeito as suas tarefas, forma e contexto.

Nosso trabalho estende a classificação realizada por Cavalcanti [Cavalcanti et al., 2014] tendo em vista que avalia as funcionalidades das FGRMs que encaixam no conceito de repositórios de RMs. Ou seja, o foco deles é no banco de dados de RMs, seja ele automatizado ou não. Contudo, o nosso objeto de estudo está em como as funcionalidades das FGRMs vêm sendo melhoradas em contrapartida do outro estudo que visa mapear os desafios e oportunidades de pesquisa na área.

3.7 Resumo do Capítulo

Neste capítulo realizamos um mapeamento sistemático com 64 estudos divididos em dois esquemas de classificação: dimensões de melhoria e suporte ao papel exercido na manutenção de software. Verificamos que tópicos como Localização de RMs Duplicadas, Atribuição (Triagem) de RMs e Classificação de RMs estão sendo tratados com maior frequência na literatura. Da mesma forma, o Agente de Triagem e Desenvolvedor possuem um maior numero de trabalhos que podem dar-lhes suporte. Neste mesmo contexto, verificamos ainda que é reduzido o número de estudos que efetivaram as melhorias propostas em protótipos de ferramentas. Esta última constatação pode causar um distanciamento entre o estado da arte e o estado da prática.

