



TRILHA PRINCIPAL

# Especificação de Requisitos de Sistemas Sensíveis a Contexto: Um Mapeamento Sistemático

## Requirements Specification of Context-Aware Systems: A Systematic Mapping

Mayke Ferreira Arruda, *Graduando em Ciência da Computação, INF-UFG,*  
Taciana Novo Kudo, *Ma. Profa. Assistente I, INF-UFG,*  
Renato de Freitas Bulcão-Neto, *Dr. Prof. Adjunto III, INF-UFG*  
*mayke.ferreira.inf@gmail.com, taciana@inf.ufg.br, renato@inf.ufg.br*

**Abstract**—Context-aware computing covers research on computational systems which adapt their behaviour so as to provide services or information to users according to their context of interaction. The literature has reported the complexity of development of such systems in the light of their features such as adaptation and heterogeneity of devices and information. This paper describes the execution of a systematic mapping on the requirements specification activity of context-aware systems, which is one of the early stages of software development. The main result includes the identification of gaps and/or research trends on requirements specification of context-aware systems.

**Index Terms**—Requirements Specification, Context Aware, Systematic Mapping.

**Resumo**—A Computação Sensível a Contexto estuda sistemas computacionais que adaptam o seu comportamento para oferecer serviços ou informações a usuários de acordo com o contexto de interação em que estes se encontram. A literatura tem reportado a complexidade de desenvolvimento de sistemas dessa natureza em função de suas características, como adaptação e heterogeneidade de dispositivos e informações. Nesse interim, este artigo apresenta uma investigação sobre o estado da arte sobre a etapa de especificação de requisitos de sistemas sensíveis a contexto, uma das primeiras etapas de desenvolvimento de software. Para isso, foi realizada um mapeamento sistemático sobre trabalhos dos últimos 10 anos, em que são identificadas lacunas e/ou tendências de pesquisa no referido assunto.

**Palavras-chave**—Especificação de Requisitos, Sensibilidade a Contexto, Mapeamento Sistemático.

### I. INTRODUÇÃO

COMPUTAÇÃO Sensível a Contexto é uma área de conhecimento em que se estuda o emprego de informações obtidas do contexto de interação do usuário, ou do ambiente onde este se encontra. O objetivo é personalizar e adaptar os serviços computacionais prestados ao usuário por meio da modificação do comportamento desses serviços no decorrer de sua execução [1] [2].

A literatura tem reportado a complexidade de desenvolvimento de sistemas sensíveis a contexto devido a suas características, como adaptação e heterogeneidade de dispositivos e informações [3] [4], entre outras.

Nesse sentido, os esforços na área de Engenharia de Software para propor métodos e processos, dentre outros elementos para apoiar a complexidade de sistemas sensíveis a contexto é crescente. A área de Engenharia de Requisitos, em especial, vem reportando inúmeras dificuldades nesta área [5] [6] [7] [8].

Os métodos tradicionais de Engenharia de Requisitos focam na resolução de ambiguidades em requisitos e defendem a especificação destes com detalhes em nível suficiente de forma que uma implementação possa ser verificada quanto a sua conformidade a essa especificação.

Sistemas sensíveis a contexto têm a capacidade de adaptar seu comportamento autônoma e dinamicamente em resposta a mudanças em condições externas. Uma dificuldade associada à engenharia de requisitos desse tipo de sistema é que esses métodos tradicionais citados não são suficientes para lidar com requisitos cujas mudanças não podem ser previstas em tempo de projeto. Quanto aos artefatos, o desafio principal é mantê-los atualizados como reflexo das mudanças ocorridas em sistemas sensíveis a contexto.

O objetivo geral deste trabalho consiste em produzir uma análise sobre a especificação de requisitos de sistemas sensíveis a contexto a fim de se extrair informações a respeito dessa área, quais as principais ferramentas e técnicas de modelagem usadas, a forma como o documento de especificação em geral é descrito e as técnicas mais utilizadas para se especificar os requisitos de contexto a fim de perceber os pontos que já estão bem definidos nesta área bem como as lacunas de pesquisa.

Para se alcançar o objetivo almejado, foi utilizado o método de mapeamento sistemático, que consiste em um estudo sobre uma série de artigos retornadas em bases de pesquisa a partir de uma mesma string de busca. Este estudo foi realizado seguindo um protocolo que guia todo o mapeamento sistemático [9] [10].

Ao fim deste artigo, são apresentados e analisados os dados quantitativos obtidos sobre questões de pesquisa referentes à especificação de requisitos de sistemas sensíveis a contexto. Sendo assim, o resultado deste trabalho pode ser um ponto de partida para trabalhos futuros na área.

O artigo está organizado como segue. A seção II discorre sobre a metodologia utilizada para a execução do trabalho, detalhando o que é e como funciona um mapeamento sistemático além do protocolo utilizado. A seção III apresenta os resultados obtidos durante o mapeamento sistemático. Por fim, a seção IV apresenta as conclusões elaboradas com base nos resultados obtidos e na análise dos mesmos.

## II. METODOLOGIA

**N**ESTA seção é apresentada em detalhes a metodologia utilizada para a realização desta pesquisa.

### A. Mapeamento Sistemático

Um mapeamento sistemático, assim como outro estudo de revisão, é uma forma de pesquisa que utiliza como fonte de dados a literatura sobre determinado tema [10]. É uma forma de identificar, avaliar e interpretar todas as pesquisas que foram encontradas e definidas como relevantes para uma determinada questão de pesquisa, área temática, ou fenômeno de interesse [9]. Trata-se de um método de revisão bibliográfica no qual a extração de dados tem como foco a classificação e categorização de resultados, sem avaliação qualitativa dos estudos primários [9] [10].

As principais características que diferenciam mapeamento sistemático e estudo bibliográfico convencional são:

- O mapeamento sistemático começa com a definição de um protocolo, que especifica a questão de pesquisa e os métodos a utilizar durante o mapeamento;
- O mapeamento sistemático é baseado em uma estratégia de busca, que visa a detectar o máximo de informação relevante sobre determinado tema;
- O mapeamento sistemático documenta a estratégia de busca para que os leitores possam avaliar o rigor, a completude e a repetibilidade do processo;
- O mapeamento sistemático requer a definição de critérios que possam justificar a exclusão de trabalhos considerados fora do escopo do mapeamento.

Sendo assim, um mapeamento sistemático é de grande utilidade, pois com ele é possível resumir as evidências já existentes, que são relevantes a algo como, por exemplo, para levantar os dados sobre o estado atual da especificação de requisitos de sistemas sensíveis a contexto.

Um mapeamento sistemático também permite identificar eventuais lacunas no tema de pesquisa, a fim de sugerir novas áreas de investigação, fornecendo assim uma direção, e posicionando apropriadamente novas atividades de pesquisa na área.

### B. Protocolo do Mapeamento Sistemático

O protocolo seguido na realização deste mapeamento sistemático é composto de cinco fases, sendo elas: planejamento, execução, seleção, extração e conclusão.

1) *Planejamento*: fase em que é definido o protocolo a ser seguido nas fases seguintes do mapeamento. Nesta fase é feita a definição do protocolo, que é planejado levando em consideração a *formulação das questões de pesquisa*, a *seleção das fontes de pesquisa* além da *seleção de estudos*.

A formulação das questões de pesquisa foi feita levando em conta o foco do estudo: “identificar o estado da arte da atividade de especificação de requisitos na construção de sistemas sensíveis a contexto”. Com isso foram levantadas questões relevantes à pesquisa, sendo elas:

- Quais são os principais artefatos gerados na especificação de requisitos de sistemas sensíveis a contexto?
- Existem modelos de especificação de requisitos próprios para sistemas sensíveis a contexto?
- Quais são os principais desafios para a criação de artefatos de especificação de requisitos de sistemas sensíveis a contexto?
- Quais são as questões em aberto sobre especificação de requisitos de sistemas sensíveis a contexto?
- Quais são as principais ferramentas ou normas utilizadas como apoio da especificação de requisitos de sistemas sensíveis a contexto?

Para o desenvolvimento de um mapeamento sistemático também são necessárias fontes de pesquisa confiáveis. Com base nisso, foram definidos critérios para a seleção das fontes de pesquisa, como sugerem Kitchenham et al.[10]:

- Fornecer mecanismos de busca por meio de *strings* com suporte a expressões lógicas;
- Produzir os mesmos resultados sempre que a mesma *string* de busca for inserida;
- Disponibilizar mecanismo de consulta via web;
- Estar relacionada a temas de Computação, incluindo Ciência da Computação, Sistemas de Informação, ou Engenharia de Software; e
- Permitir filtro por ano de publicação

Com base nos critérios definidos acima, foram escolhidas sete fontes de pesquisa que satisfizeram os critérios estabelecidos, sendo:

- ACM Digital Library
- Biblioteca Digital Brasileira de Teses e Dissertações (BDTD)
- Engineering Village
- IEEEExplorer
- Scopus
- Springer
- Web of Science

Ainda na fase de planejamento, foi definido que seriam considerados todos os estudos escritos em Inglês ou Português com ano de publicação entre 2005 e 2014.

Após terem sido definidos a questão de pesquisa, as fontes escolhidas e critérios para a seleção, o próximo passo foi decidir as *strings* de busca a serem executadas em todas as fontes de pesquisa escolhidas. Também ficou decidido que as *strings* de busca seriam aplicadas no título e *abstract/resumo* de cada trabalho.

Seguindo a recomendação de Kitchenham et al.[10], a definição das *strings* de busca devem ser antecedidas

de testes com diferentes *strings*. O objetivo é analisar os resultados de busca obtidos e verificar a precisão desses em relação às *strings* utilizadas. Esse trabalho foi realizado por três pesquisadores, para melhor refinamento das *strings*, até que os resultados se aproximassem do objetivo desta pesquisa. Assim, foram definidas as duas *strings* de busca a seguir:

*String* para artigos em Inglês: (“requirements specification” AND “context-aware” AND software)

*String* para artigos em Português: (“especificação de requisitos” AND (“sensível a contexto” OR “sensível ao contexto”) AND software)

Também foram definidos os critérios usados para a exclusão dos estudos, sendo eles:

- trabalhos não escritos nos idiomas definidos neste protocolo;
- trabalhos cujo texto completo não esteja disponível para acesso gratuitamente na web ou no portal de periódicos da CAPES;
- trabalhos não relacionados a área de Engenharia de Requisitos;
- trabalhos que não abordam especificação de requisitos;
- trabalhos não relacionados a aplicações sensíveis a contexto.

Também ficou definido que seriam considerados os trabalhos que fossem teses, dissertações, capítulos de livros com *abstract* e artigos de periódicos ou de conferências que satisfizessem os critérios de inclusão e que não se enquadrassem nos critérios de exclusão de estudos já definidos neste protocolo.

2) *Execução*: esta fase consistiu em executar as *strings* de busca nas fontes de pesquisa que foram definidas anteriormente, e em preparar todas as informações retornadas.

Foi definido que os resultados retornados pelas ferramentas de busca deveriam ser armazenados em uma planilha eletrônica, utilizando o software *LibreOffice Calc*. Para armazenar os dados, foi necessário exportar os resultados de cada ferramenta de busca. Para essa tarefa foi utilizado o mecanismo disponível na maioria das ferramentas de busca, que gera um arquivo no formato *CSV*.

Naquelas ferramentas de busca em que não havia este mecanismo de exportação, foi utilizado o *plug-in* externo *Zotero* para realizar esta exportação adequadamente.

3) *Seleção*: esta fase consistiu em selecionar os estudos, com base no título e *abstract*/resumo de cada trabalho, levando em consideração os critérios de inclusão e exclusão já apresentados neste protocolo.

Uma nova coluna foi criada na tabela, e após ler cada artigo, este é marcado com o status adequado (I = Incluído, E = Excluído).

Para cada artigo marcado com o status Excluído, foi atribuído um dos critérios de exclusão a seguir:

- 1) Não responde à questão de especificação de requisitos;
- 2) Não é sobre aplicação sensível a contexto;
- 3) Não é sobre especificação de requisitos, nem sobre aplicação sensível a contexto (combinação de 1 e 2);
- 4) Artigo duplicado;
- 5) Não é um artigo, nem tese ou dissertação; e
- 6) Documento não disponível.

O processo de inclusão/exclusão de artigos foi conduzido entre três pessoas: o aluno, o orientador e a co-orientadora do trabalho. É de conhecimento dos autores que esse processo deva incluir mais pessoas para reduzir distorções nos critérios e exclusões indevidas de artigos relevantes para a pesquisa.

4) *Extração*: nesta etapa ocorreu a leitura do texto de cada artigo incluído na íntegra. Com isso foi possível esclarecer dúvidas e permitir uma decisão mais embasada a respeito de sua manutenção ou exclusão, considerando os critérios estabelecidos.

Nessa etapa também foram extraídos os dados relevantes de cada trabalho a fim de tentar responder as cinco questões de pesquisa levantadas na fase de planejamento deste protocolo.

5) *Conclusão*: nesta fase foi possível emitir relatórios e gráficos com base nos resultados encontrados pelas ferramentas de busca, e então elaborar algumas conclusões.

### III. ANÁLISE DOS RESULTADOS

NESTA seção serão apresentados e analisados os resultados obtidos em cada uma das fases do mapeamento sistemático.

Como resultado da fase de *execução*, em que foram executadas as *strings* em diferentes bibliotecas digitais, foram retornados um total de 462 artigos, conforme apresentam as Tabelas I e II.

Tabela I: Artigos retornados na fase de Execução classificados por ferramenta de busca.

Ferramenta de Busca	Artigos Retornados
ACM DL	101
Engineering Village	4
IEEE Explorer	100
Science Direct	34
Scopus	79
Springer	144
Web of Science	0
BDTD	0
<b>Total</b>	<b>462</b>

Na fase de *seleção*, os artigos foram marcados com os status de incluído ou excluído, e para cada artigo marcado como excluído foi atribuído um critério de exclusão. A Tabela III apresenta as ocorrências de cada critério de exclusão na fase de seleção.

Tabela II: Artigos retornados na fase de Execução classificados por ano de publicação.

Ano	Artigos Retornados
2005	25
2006	38
2007	36
2008	48
2009	48
2010	65
2011	67
2012	52
2013	64
2014	19

Tabela III: Critérios de exclusão de artigos e respectivas ocorrências na fase de Seleção

Critério de Exclusão	Artigos Excluídos
1	109
2	50
3	134
4	88
5	15
6	17
<b>Total excluídos</b>	<b>413</b>

Analisando a Tabela III, foi possível perceber que a maioria dos artigos excluídos ocorreu pelos critérios de 1 a 4, o que pode sugerir um problema na *string* de busca, sendo que as eliminações com os critérios de 1 a 3 (70%) aconteceram devido ao fato de apenas mencionarem os termos “*requirements specification*” ou “*context-aware*”, e não a conjunção deles no *abstract*. Houve também um número significativo de eliminações pelo critério 4 (21%) devido à duplicidade de artigos (de mesmo título, autores e ano de publicação) que foram retornados por mecanismos de busca diferentes.

Ao fim da fase de seleção, restaram 49 artigos que são apresentados nas Tabelas IV e V: a Tabela IV apresenta-os agrupados por base de busca, enquanto que a Tabela V, agrupados por ano de publicação.

Tabela IV: Artigos restantes após a fase de Seleção classificados por ferramenta de busca.

Ferramenta de Busca	Artigos Restantes
ACM DL	10
Engineering Village	0
IEEE Explorer	17
Science Direct	3
Scopus	9
Springer	10
Web of Science	0
BDTD	0
<b>Total</b>	<b>49</b>

Tabela V: Artigos restantes após a fase de Seleção classificados por ano de publicação.

Ano	Artigos Restantes
2005	3
2006	5
2007	4
2008	6
2009	4
2010	9
2011	8
2012	4
2013	5
2014	1

Após a leitura dos 49 artigos restantes da fase de *seleção*, foi possível determinar se um artigo era de alguma forma relevante ou não à pesquisa.

Foi definido que seria classificado como irrelevante para esta pesquisa o artigo que não contivesse proposta de documento de especificação ou de ferramenta de apoio à especificação, bem como não fazer o uso de alguma ferramenta de apoio à especificação, de normas internacionais, nem de fornecer detalhes sobre como requisitos são especificados.

Após todas as eliminações de artigos sem relevância, dos 49 artigos que iniciaram a fase de extração, 17 artigos foram eliminados, restando assim 32 artigos com algum tipo de informação relevante para a pesquisa. A partir destes artigos é possível extrair algumas informações úteis.

A Tabela VI apresenta os artigos restantes após a fase de extração agrupados por base de busca, enquanto que a Tabela VII apresenta os mesmos artigos agrupados por ano de publicação.

Tabela VI: Artigos restantes após a fase de Extração classificados por ferramenta de busca.

Ferramenta de Busca	Artigos Restantes
ACM DL	8
Engineering Village	0
IEEE Explorer	12
Science Direct	1
Scopus	6
Springer	5
Web of Science	0
BDTD	0
<b>Total</b>	<b>32</b>

Os Gráficos 1 a 4 apresentam os dados referentes às Tabelas VI e VII a fim de proporcionar uma melhor estruturação e facilitar a percepção dos resultados.

O Gráfico 1 apresenta o resultado do artigos resultantes ao fim das fases de Execução, Seleção e Extração agrupados por ferramenta de busca. O Gráfico 2 apresenta os mesmos resultados agrupados por ano de publicação.

Tabela VII: Artigos restantes após a fase de Extração classificados por ano de publicação.

Ano	Artigos Restantes
2005	3
2006	5
2007	4
2008	6
2009	4
2010	9
2011	8
2012	4
2013	5
2014	1

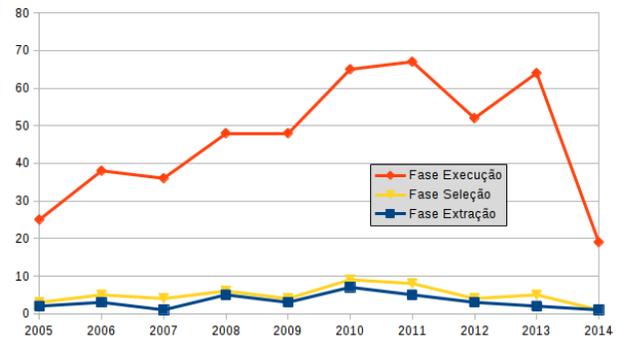


Gráfico 2: Artigos restantes após as três fases, classificados por ano de publicação.

Percebe-se de acordo com o Gráfico 3, que dos artigos relevantes, mais de 60% são oriundos das bibliotecas digitais da ACM e do IEEE, o que sugere que conferências e/ou periódicos com chancelas dessas instituições atraem pesquisadores de Engenharia de Requisitos aplicada à Computação Sensível a Contexto.

Pode-se perceber também que houve um acréscimo de trabalhos na área ao longo dos dois últimos quinquênios: 44% (14 artigos) foram publicados entre 2005 e 2009 e 56% (18 artigos) foram publicados entre 2010 e 2014. Como esta pesquisa ocorreu em meados de 2014, esta não incluiu todos os trabalhos publicados nesse ano. Dessa maneira, esse crescimento de 44 para 56% pode sugerir que Engenharia de Requisitos de Sistemas Sensíveis a Contexto é um tema interessante para a comunidade (Gráfico 4).

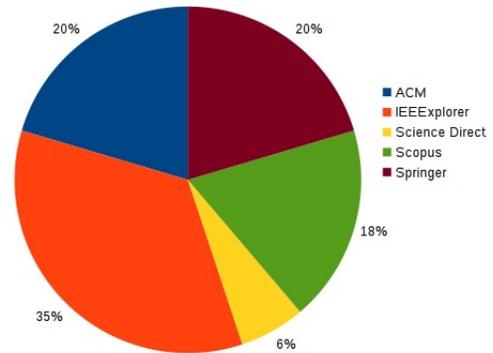


Gráfico 3: Artigos relevantes classificados por biblioteca digital.

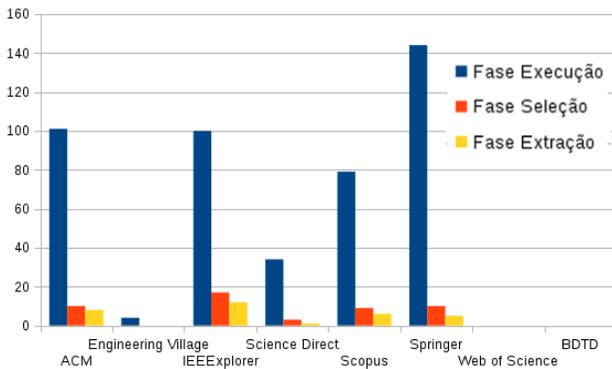


Gráfico 1: Artigos restantes após as três fases, classificados por ferramenta de busca.

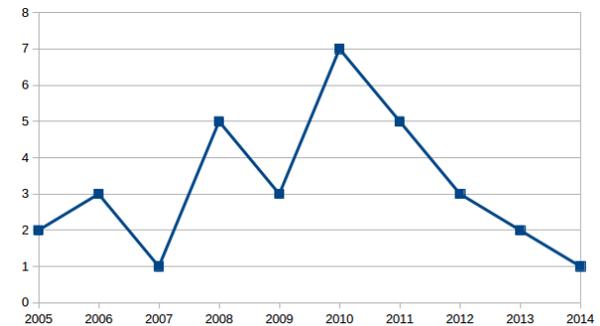


Gráfico 4: Artigos relevantes classificados por ano de publicação.

De acordo com o Gráfico 5, apenas 13% dos artigos apresentam proposta de modelo de artefato de especificação de requisitos e 13% dos artigos não apresentam modelo de artefato, mas utilizam alguma norma internacional para especificação de requisitos. Esses dados podem sugerir que essas pesquisas não têm como foco a proposição de artefatos e/ou normas inovadoras para especificação de requisitos, utilizando as já conhecidas na área de engenharia de requisitos.

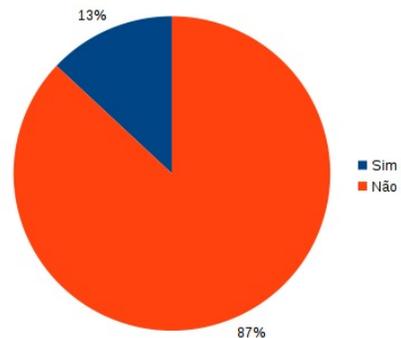


Gráfico 5: Quantidade de trabalhos que apresentam uma nova proposta de documento de especificação de requisitos.

As normas utilizadas pelos autores nos trabalhos analisados incluem:

- ISO/IEC 9126-1: norma para qualidade de produto de software, estabelecendo modelos de qualidade para processo, produto externo e qualidade em uso [11]. Foi substituída pela norma ISO/IEC 25000 [12];
- ISO/IEC 25020: norma que fornece modelos de referência e um guia para medir e avaliar os requisitos de qualidade de software [13];
- ISO/IEC 25030: norma que fornece requisitos e recomendações para a especificação dos requisitos de qualidade de software [14];
- IEEE 830: norma que descreve recomendações para a especificação de requisitos de software [15]. Foi substituída pela norma IEEE Std 29148 [16];
- ISO 10303 - 233: norma que define requisitos, escopo e informação para vários estágios de desenvolvimento durante o projeto de um sistema [17].

Quanto à especificação e modelagem de requisitos (Gráfico 6), 47% dos trabalhos empregam a linguagem natural. A descrição formal corresponde a 30%, e em menor número (19%) fazem o uso de descrição formal com ontologias. A menor parte (4%) faz o uso da linguagem UML para especificar os requisitos. Isto sugere que a forma mais fácil de especificar os requisitos ainda é a linguagem natural, porém com a necessidade de especificar requisitos em tempo de execução e sem ambiguidade, descrições formais estão sendo muito usadas, já que as duas formas de descrição formal, com alguma linguagem e com ontologias se somadas já são a grande maioria (49%).

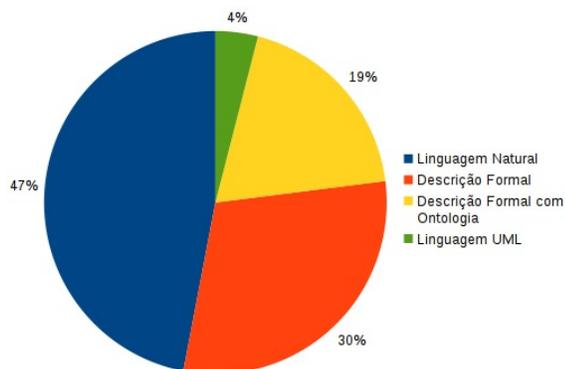


Gráfico 6: Como é feita a especificação e modelagem dos requisitos nos trabalhos analisados.

No Gráfico 7, pode-se perceber que a maioria dos artigos considerados relevantes (62%) apresenta alguma proposta para apoiar a especificação de requisitos de aplicações sensíveis a contexto (ferramenta, *framework*, linguagem, etc), sugerindo assim que o desenvolvimento de novas ferramentas de apoio ainda é uma área que exige pesquisa.

Ao analisar as propostas apresentadas pelo autores referentes a ferramentas de apoio à especificação, pode-se concluir que 50% dos trabalhos se preocupam especificamente com a modelagem dos requisitos e 16% dos trabalhos possuem o foco de pesquisa na definição dos requisitos

não funcionais. Além disso, 33% focam os esforços de pesquisa em como permitir que requisitos sejam incluídos e atualizados em tempo de execução nas aplicações sensíveis a contexto. Sendo assim, pode-se perceber que atualmente os principais esforços são para *modelar os requisitos da forma mais eficiente possível e também na especificação e atualização dos requisitos em tempo de execução*, o que é um dos desafios da área.

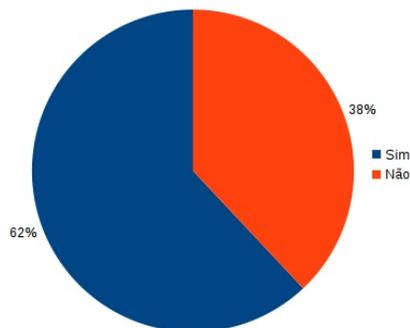


Gráfico 7: Trabalhos que propõem alguma ferramenta de apoio à especificação de requisitos.

No Gráfico 8, pode-se perceber que a maioria dos artigos considerados relevantes (62%) não fez o uso de qualquer ferramenta de apoio à especificação de requisitos de aplicações sensíveis a contexto, números inversos à parcela dos trabalhos que propõem ferramentas. Isso pode sugerir que, no geral, trabalhos que não propõem uma ferramenta nova, fazem o uso de uma já existente para outro fim.

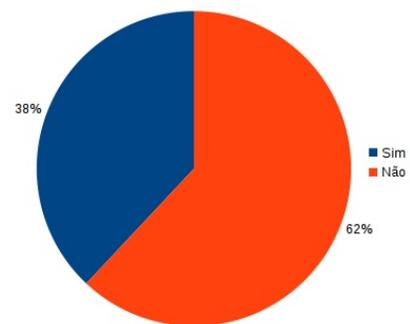


Gráfico 8: Trabalhos que fazem uso de alguma ferramenta de apoio à especificação de requisitos.

A partir dos 32 artigos relevantes, foi possível descobrir que aproximadamente 30% (i.e. 9) são originados de grupos de pesquisa na Itália. Em particular da Universidade de L'Aquila, na Itália, destaca-se a pesquisadora Paola Inverardi com 3 dessas 9 publicações encontradas.

Pesquisadores norte-americanos reúnem 4 publicações relevantes quanto ao tema abordado, porém sem uma predominância em determinado grupo de pesquisa. Por outro lado, dos 5 artigos que envolvem pesquisadores do Reino Unido, em 2 deles existe a co-autoria de pesquisadores italianos. Há também 3 publicações relacionadas ao assunto de pesquisadores brasileiros, sendo uma também com parceria de pesquisador italiano.

A Tabela VIII estrutura todos os resultados mostrados nesta seção com a finalidade de facilitar a leitura e compreensão dos resultados finais.

Tabela VIII: Resumo dos resultados da pesquisa.

Informação	Dados obtidos
Artigos retornados na fase de Execução	462 artigos
Artigos restantes após a fase de Seleção	49 artigos
Artigos restantes após a fase de Extração (considerados relevantes)	32 artigos
Base de busca mais relevantes	IEEEExplorer (38%) seguida de ACM DL (25%)
Pesquisador(a) de destaque	Paola Inverardi, Universidade de L'Aquila, Itália
Evolução do tema ao longo dos anos	Dos artigos relevantes, 44% foram publicados entre 2005 e 2009 e 56% entre 2010 e 2014
Quanto à especificação dos requisitos	Linguagem natural (47%), descrição formal (30%), ontologias (19%) e linguagem UML (4%)
Artigos que propõem modelo de artefato de especificação de requisitos	13% dos relevantes
Artigos que usam normas internacionais para especificação de requisitos	13% dos relevantes
Artigos que propõem ferramentas para apoiar especificação de requisitos	62% dos relevantes
Quanto aos artigos que propõem ferramentas para apoiar especificação de requisitos	Modelagem dos requisitos (50%); captura, inclusão e atualização de requisitos em tempo de execução (34%); foco em requisitos não funcionais (16%)
Questões em aberto	Requisitos em tempo de execução para sistemas sensíveis a contexto

#### IV. CONCLUSÕES E TRABALHOS FUTUROS

**E**STE trabalho descreve uma análise sobre o estado da arte da especificação de requisitos de sistemas sensíveis a contexto, resumida nas questões de pesquisa a seguir, cujas respostas encontram-se na Tabela VIII:

- Existem modelos de documentos de especificação?
- Como são especificados os requisitos?
- Utilizam norma internacional para especificação?
- Propõem (ou fazem uso de) alguma ferramenta de apoio à especificação?
- Quais são as questões em aberto sobre especificação?

Para realizar o objetivo almejado, foi utilizada o método de mapeamento sistemático, que é uma forma de pesquisa que utiliza como fonte de dados a literatura sobre determinado tema e segue um protocolo com fases bem definidas.

Ao fim deste trabalho, foi obtida uma síntese sobre especificação de requisitos para sistemas sensíveis a contexto, indicando de forma quantitativa, com o auxílio de números, tabelas e gráficos, como se encontra o estado atual desta área em relação ao que já está consolidado e aos pontos que futuras pesquisas devem abordar.

Dos 462 artigos retornados inicialmente, um total de 32 foi considerado relevante à pesquisa (7%). Destes considerados relevantes<sup>1</sup>, foi possível perceber que:

- as bibliotecas digitais ACM DL e IEEEExplorer incluem a maioria dos artigos relevantes sobre o tema;
- pesquisas no tema deste trabalho não têm como foco o desenvolvimento de modelos de especificação de documentos ou normas específicas;
- quanto à especificação de requisitos, linguagem natural ainda é a forma mais adotada dentre os artigos relevantes para esta pesquisa;
- o desenvolvimento de novas ferramentas de apoio à especificação de requisitos ainda é um tema de interesse da comunidade acadêmica;
- uma grande parcela das pesquisas atuais tem como foco a manipulação de requisitos em tempo de execução para adaptação do comportamento de sistemas sensíveis a contexto;
- as principais referências no tema Requisitos em Tempo de Execução são pesquisadores de universidades italianas, que também incluem parcerias com grupos de pesquisa da Inglaterra e do Brasil.

Sendo assim, como contribuição final pode-se perceber que o tema Especificação de Requisitos de Sistemas Sensíveis a Contexto continua sendo relevante, com tendência para a especificação de requisitos em tempo de execução, em função das características do sistema-alvo da pesquisa.

Finalmente, com base nas conclusões apresentadas durante este trabalho e com o objetivo de dar continuidade a pesquisas sobre este assunto, são sugeridos os seguintes trabalhos futuros:

- 1) Um trabalho investigativo quanto ao tratamento de requisitos em tempo de execução para sistemas sensíveis a contexto, desde a sua captura até a adaptação do comportamento do sistema;
- 2) Uma ferramenta que apoie a captura, inclusão e atualização de requisitos de sistemas sensíveis a contexto em tempo de execução; uma ferramenta com essas capacidades constitui um desafio na área.

Vale ressaltar que ambos os itens supracitados são objetos de pesquisa do grupo do qual fazem parte os autores. Essas pesquisas exploram o uso de ontologias na especificação e modelagem de requisitos em tempo de execução para sistemas autoadaptativos [6] [18].

<sup>1</sup>Os artigos relevantes à pesquisa (título, autores e ano de publicação) encontram-se na Tabela IX do apêndice deste artigo.

## REFERÊNCIAS

- [1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith e P. Steggles, “Towards a better understanding of context and context-awareness”, em *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, Karlsruhe, Germany: Springer-Verlag, 1999, pp. 304–307.
- [2] A. K. Dey, “Understanding and using context”, *Personal Ubiquitous Comput.*, vol. 5, n° 1, pp. 4–7, jan. de 2001.
- [3] R. B. de Araujo, “Computação ubíqua: Princípios, tecnologias e desafios”, em *XXI Simpósio Brasileiro de Redes de Computadores*, 2004, pp. 45–114.
- [4] A. A. Loureiro et al, “Computação ubíqua ciente de contexto: Desafios e tendências”, em *XXVII Simpósio Brasileiro de Reschooldes de Computadores e Sistemas Distribuídos*, 2009, pp. 99–149.
- [5] V. Vieira, P. Tadesco e C. Salgado, “Modelos e processos para o desenvolvimento de sistemas sensíveis ao contexto”, em *Jornadas de Atualização em Informática (JAI'09)*, 2009, pp. 1–44.
- [6] N. Bencomo, J. Whittle, P. Sawyer, A. Finkelstein e E. Letier, “Requirements reflection: Requirements as runtime entities”, em *Software Engineering, 2010 ACM/IEEE 32nd International Conference on*, vol. 2, 2010, pp. 199–202.
- [7] S. H. Siadat e M. Song, “Understanding requirement engineering for context-aware service-based applications”, *Journal of Software Engineering and Applications*, vol. 5, n° 8, pp. 536–544, 2012.
- [8] J. a. Pimentel, K. Angelopoulos, V. E. S. Souza, J. Mylopoulos e J. F. B. Castro, “From requirements to architectures for better adaptive software systems”, em *Proceedings of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, San Francisco, California: IEEE Press, jun. de 2013, pp. 23–32.
- [9] B. Kitchenham e S. Charters, “Guidelines for performing systematic literature reviews in software engineering”, Keele University e Durham University Joint Report, rel. téc., 2007.
- [10] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey e S. Linkman, “Systematic literature reviews in software engineering - a systematic literature review”, *Information and Software Technology*, vol. 51, n° 1, pp. 7–15, 2009, Special Section - Most Cited Articles in 2002 and Regular Research Papers.
- [11] ISO/IEC, “Software engineering – product quality, {iso/iec} 9126-1”, International Organization for Standardization, rel. téc., 2001.
- [12] —, “Iso/iec 25000 – software engineering – software product quality requirements and evaluation (square) – guide to square”, rel. téc., 2005.
- [13] “Software engineering – software product quality requirements and evaluation (square) – measurement reference model and guide”, rel. téc., 2007.
- [14] “Software engineering – software product quality requirements and evaluation (square) – quality requirements”, rel. téc., 2007.
- [15] “Ieee recommended practice for software requirements specifications”, *IEEE 830*, pp. 1–40, 1998.
- [16] “Systems and software engineering – life cycle processes – requirements engineering”, *IEEE Std 29148*, pp. 1–94, 2011.
- [17] “Industrial automation systems and integration – product data representation and exchange – part 233: Application protocol: Systems engineering”, rel. téc., 2012.
- [18] P. Sawyer, N. Bencomo, J. Whittle, E. Letier e A. Finkelstein, “Requirements-aware systems: A research agenda for re for self-adaptive systems”, em *Proceedings of the 2010 18th IEEE International Requirements Engineering Conference*, Washington, DC, USA: IEEE Computer Society, 2010, pp. 95–103.
- [19] S. Wang, S. K. Birla e S. Neema, “A modeling language for vehicle motion control behavioral specification”, em *Proceedings of the 2006 International Workshop on Software Engineering for Automotive Systems*, sér. SEAS '06, Shanghai, China: ACM, 2006, pp. 53–60.
- [20] K. Oliveira, J. Castro, E. Santos, R. do Nascimento Fidalgo, S. España e O. Pastor, “A multi level approach to autonomic business process.”, em *SBES*, IEEE Computer Society, 2012, pp. 91–100.
- [21] J. Dehlinger e R. R. Lutz, “A product-line requirements approach to safe reuse in multi-agent systems”, em *Proceedings of the Fourth International Workshop on Software Engineering for Large-scale Multi-agent Systems*, St. Louis, Missouri: ACM, 2005, pp. 1–7.
- [22] H. Ben Hadji, S.-K. Kim e H.-J. Choi, “A representation model for reusable assets to support user context”, em *Service-Oriented System Engineering, 2008. SOSE '08. IEEE International Symposium on*, 2008, pp. 91–96.
- [23] V. Castelli, P. Thomas, R. Bertone e A. Oliveros, “A requirements engineering process extended to context information management”, em *Research Challenges in Information Science (RCIS), 2011 Fifth International Conference on*, 2011, pp. 1–6.
- [24] “A requirements-led approach for specifying qos-aware service choreographies: An experience report”, em *Requirements Engineering: Foundation for Software Quality*, sér. Lecture Notes in Computer Science, C. Salinesi e I. van de Weerd, eds., vol. 8396, Springer International Publishing, 2014, pp. 239–253, ISBN: 978-3-319-05842-9.
- [25] M. Galster e E. Bucherer, “A taxonomy for identifying and specifying non-functional requirements in service-oriented development”, em *Services - Part I, 2008. IEEE Congress on*, 2008, pp. 345–352.
- [26] H. Molina e L. Olsina, “Assessing web applications consistently: A context information approach”, em

- Web Engineering, 2008. ICWE '08. Eighth International Conference on*, 2008, pp. 224–230.
- [27] N. Qureshi e A. Perini, “Continuous adaptive requirements engineering: An architecture for self-adaptive service-based applications”, em *Requirements@Run.Time (RE@RunTime), 2010 First International Workshop on*, 2010, pp. 17–24.
- [28] K. E. Kjær, “Designing middleware for context awareness in agriculture”, em *Proceedings of the 5th Middleware Doctoral Symposium*, Leuven, Belgium: ACM, 2008, pp. 19–24.
- [29] N. Šerbedžija, T. Bureš e J. Keznlk, “Engineering autonomous systems”, em *Proceedings of the 17th Panhellenic Conference on Informatics*, sér. PCI '13, Thessaloniki, Greece: ACM, 2013, pp. 128–135.
- [30] P. Inverardi e M. Mori, “Feature oriented evolutions for context-aware adaptive systems”, em *Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)*, Antwerp, Belgium: ACM, 2010, pp. 93–97.
- [31] A. Coronato e G. De Pietro, “Formal design of ambient intelligence applications”, *Computer*, vol. 43, n° 12, pp. 60–68, 2010.
- [32] A. Coronato e G. D. Pietro, “Formal specification of wireless and pervasive healthcare applications”, *ACM Trans. Embed. Comput. Syst.*, vol. 10, n° 1, pp. 1–18, ago. de 2010.
- [33] P. Inverardi e M. Mori, “Model checking requirements at run-time in adaptive systems”, em *Proceedings of the 8th Workshop on Assurances for Self-adaptive Systems*, sér. ASAS '11, Szeged, Hungary: ACM, 2011, pp. 5–9.
- [34] A. Lapouchnian e J. Mylopoulos, “Modeling domain variability in requirements engineering with contexts”, em *Proceedings of the 28th International Conference on Conceptual Modeling*, Gramado, Brazil: Springer-Verlag, 2009, pp. 115–130.
- [35] A. Frece e M. B. Juric, “Modeling functional requirements for configurable content- and context-aware dynamic service selection in business process models.”, *J. Vis. Lang. Comput.*, vol. 23, n° 4, pp. 223–247, 2012.
- [36] A. Kofod-Petersen e J. Cassens, “Modelling with problem frames: Explanations and context in ambient intelligent systems”, em *Modeling and Using Context*, sér. Lecture Notes in Computer Science, M. Beigl, H. Christiansen, T. Roth-Berghofer, A. Kofod-Petersen, K. Coventry e H. Schmidtke, eds., vol. 6967, Springer Berlin Heidelberg, 2011, pp. 145–158.
- [37] N. Koay, P. Kataria, R. Juric, P. Oberndorf e G. Terstyanszky, “Ontological support for managing non-functional requirements in pervasive healthcare”, em *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, 2009, pp. 1–10.
- [38] A. Sutcliffe, S. Fickas e M. Sohlberg, “Pc-re: A method for personal and contextual requirements engineering with some experience”, *Requirements Engineering*, vol. 11, n° 3, pp. 157–173, 2006.
- [39] P.-A. Hsiung, S.-W. Lin e C.-S. Lin, “Real-time embedded software design for mobile and ubiquitous systems”, *J. Signal Process. Syst.*, vol. 59, n° 1, pp. 13–32, abr. de 2010.
- [40] R. Ali, F. Dalpiaz e P. Giorgini, “Reasoning with contextual requirements: Detecting inconsistency and conflicts”, *Inf. Softw. Technol.*, vol. 55, n° 1, pp. 35–57, jan. de 2013.
- [41] P. Inverardi e M. Mori, “Requirements models at run-time to support consistent system evolutions”, em *Requirements@Run.Time (RE@RunTime), 2011 2nd International Workshop on*, 2011, pp. 1–8.
- [42] B. Larson, J. Hatcliff, S. Procter e P. Chalin, “Requirements specification for apps in medical application platforms”, em *Software Engineering in Health Care (SEHC), 2012 4th International Workshop on*, 2012, pp. 26–32.
- [43] B.-Y.-S. Lau, C. Pham-Nguyen, C.-S. Lee e S. Garlatti, “Semantic web service adaptation model for a pervasive learning scenario”, em *Innovative Technologies in Intelligent Systems and Industrial Applications, 2008. CITISIA 2008. IEEE Conference on*, 2008, pp. 98–103.
- [44] J. Pimentel, J. Castro e X. Franch, “Specification of failure-handling requirements as policy rules on self-adaptive systems”, em *WER'11*, 2011.
- [45] S. S. Yau, D. Huang, H. Gong e Y. Yao, “Support for situation awareness in trustworthy ubiquitous computing application software”, *Software: Practice and Experience*, vol. 36, n° 9, pp. 893–921, 2006.
- [46] T. Broens, D. Quartel e M. van Sinderen, “Towards a context binding transparency”, em *Dependable and Adaptable Networks and Services*, vol. 4606, Springer Berlin Heidelberg, 2007, pp. 9–16.
- [47] U. Bellur e N. Narendra, “Towards service orientation in pervasive computing systems”, em *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, vol. 2, 2005, pp. 289–295.
- [48] R. O. Spínola, F. C. d. R. Pinto e G. H. Travassos, “Ubicheck: An approach to support requirements definition in the ubicomp domain”, em *Proceedings of the 2010 ACM Symposium on Applied Computing*, Sierre, Switzerland: ACM, 2010, pp. 306–310.
- [49] S. Adam, J. Doerr, M. Eisenbarth e A. Gross, “Using task-oriented requirements engineering in different domains - experiences with application in research and industry”, em *Proceedings of the 2009 17th IEEE International Requirements Engineering Conference, RE*, Washington, DC, USA: IEEE Computer Society, 2009, pp. 267–272.

## APÊNDICE

Tabela IX: Tabela com os 32 artigos relevantes à pesquisa (após a fase de Extração).

Art.	Título	Autores	Ano
1	A Modeling Language for Vehicle Motion Control Behavioral Specification [19]	Wang, Shige; Birla, Sushil K. & Neema, Sandeep	2006
2	A Multi Level Approach to Autonomic Business Process [20]	Oliveira, K.; Castro, J.; Santos, E.; Fidalgo, R.; Espana, S.; Pastor, O.	2012
3	A Product-line Requirements Approach to Safe Reuse in Multi-agent Systems [21]	Dehlinger, Josh & Lutz, Robyn R.	2005
4	A Representation Model for Reusable Assets to Support User Context [22]	Ben Hadji, H.; Su-Kyoung Kim; HoJin Choi	2008
5	A requirements engineering process extended to context information management [23]	Castelli, V.; Thomas, P.; Bertone, R.; Oliveros, A	2011
6	A Requirements-Led Approach for Specifying QoS-Aware Service Choreographies: An Experience Report [24]	Maiden, N.; Lockerbie, J.; Konstantinos, Z.; Bertolino, A.; DE Angelis, G.; Lonetti, F.	2014
7	A Taxonomy for Identifying and Specifying Non-Functional Requirements in Service-Oriented Development [25]	Galster, M.; Bucherer, E.	2008
8	Assessing web applications consistently: A context information approachL. [26]	Molina H., Olsina	2008
9	Continuous adaptive requirements engineering: An architecture for self-adaptive service-based applications [27]	Qureshi, N.A.; Perini, A.	2010
10	Designing Middleware for Context Awareness in Agriculture [28]	Kjaer, Kristian Ellebaek	2008
11	Engineering Autonomous Systems [29]	Serbedzija, Nikola; Bures, Tomas & Keznikl, Jaroslav	2013
12	Feature oriented evolutions for context-aware adaptive systems [30]	Inverardi P., Mori M.	2010
13	Formal Design of Ambient Intelligence Application [31]	Coronato, A.; De Pietro, G.	2010
14	Formal Specification of Wireless and Pervasive Healthcare Applications [32]	Coronato, A.; DE Pietro, G	2010
15	Model Checking Requirements at Run-time in Adaptive Systems [33]	Inverardi, Paola & Mori, Marco	2011
16	Modeling domain variability in requirements engineering with contexts [34]	Lapouchnian A., Mylopoulos J.	2009
17	Modeling Functional Requirements for Configurable Content- and Context-aware Dynamic Service Selection in Business Process Models [35]	Frece, Ales & Juric, Matjaz B.	2012
18	Modelling with Problem Frames: Explanations and Context in Ambient Intelligent Systems [36]	Anders Kofod-Petersen, Jorg Cassens	2011
19	Ontological Support for Managing Non-Functional Requirements In Pervasive Healthcare [37]	Koay, N.; Kataria, P.; Juric, R.; Oberndorf, P.; Terstyanszky, G.	2009
20	PC-RE: a method for personal and contextual requirements engineering with some experience [38]	Alistair Sutcliffe; Stephen Fickas McKay ; Moore Sohlberg	2006
21	Real-Time Embedded Software Design for Mobile and Ubiquitous Systems [39]	Pao-Ann Hsiung; Shang-Wei Lin; Chao-Sheng Lin	2010
22	Reasoning with contextual requirements: Detecting inconsistency and conflicts [40]	Raian Ali; Fabiano Dalpiaz ; Paolo Giorgini	2013
23	Requirements models at run-time to support consistent system evolutions [41]	Inverardi, P.; Mori, M.	2011
24	Requirements specification for apps in medical application platforms [42]	Larson, B.; Hatcliff, J.; Procter, S.; Chalin, P.	2012

Tabela IX: Tabela com os 32 artigos relevantes à pesquisa (após a fase de Extração).

25	Requirements-aware systems: A research agenda for RE for self-adaptive systems [18]	Sawyer P., Bencomo N., Whittle J., Letie E., Finkelstein A.	2010
26	Semantic web service adaptation model for a pervasive learning scenario [43]	Lau, B-Y-S; Pham-Nguyen, C.; Lee, C-S; Garlatti, S.	2008
27	Specification of failure-handling requirements as policy rules on self-adaptive systems [44]	Pimentel J., Castro J., Franch X	2011
28	Support for situation awareness in trustworthy ubiquitous computing application software [45]	Yau S.S., Huang D., Gong H., Yao Y.	2006
29	Towards a Context Binding Transparency [46]	Tom Broens; Dick Quartel; Marten Van Sinderen	2007
30	Towards service orientation in pervasive computing systems [47]	Bellur, U.; Narendra, N.C.	2005
31	UbiCheck: An Approach to Support Requirements Definition in the Ubicomp Domain [48]	Spínola, R. O.; Pinto, F. C.	2010
32	Using Task-oriented Requirements Engineering in Different Domains Experiences with Application in Research and Industry [49]	Adam, S.; Doerr, J.; Eisenbarth, M.; Gross, A.	2009