



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

SABRINA CUNHA DE SOUSA

UMA REVISÃO SISTEMÁTICA DE LITERATURA DAS EXTENSÕES DA KAOS
(KNOWLEDGE ACQUISITION IN AUTOMATED SPECIFICATION)

QUIXADÁ
2019

SABRINA CUNHA DE SOUSA

UMA REVISÃO SISTEMÁTICA DE LITERATURA DAS EXTENSÕES DA KAOS
(KNOWLEDGE ACQUISITION IN AUTOMATED SPECIFICATION)

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de Informação
do Campus Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Enyo José Tavares
Gonçalves

QUIXADÁ

2019

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

S698r Sousa, Sabrina Cunha de.
Uma revisão sistemática de literatura das extensões da KAOS (Knowledge Acquisition in Automated Specification) / Sabrina Cunha de Sousa. – 2019.
67 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Sistemas de Informação, Quixadá, 2019.
Orientação: Prof. Dr. Enyo José Tavares Gonçalves.

1. Revisão Sistemática. 2. Modelagem Computacional. 3. Engenharia de Requisitos . I. Título.

CDD 005

SABRINA CUNHA DE SOUSA

UMA REVISÃO SISTEMÁTICA DE LITERATURA DAS EXTENSÕES DA KAOS
(KNOWLEDGE ACQUISITION IN AUTOMATED SPECIFICATION)

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de Informação
do Campus Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Sistemas de Informação.

Aprovada em: _____ \ _____ \ _____

BANCA EXAMINADORA

Prof. Dr. Enyo José Tavares Gonçalves (Orientador)
Universidade Federal do Ceará - Campus Quixadá
(UFC)

Prof. Dr. João Araújo
Faculdade de Ciências e Tecnologia - Universidade
Nova de Lisboa (FCT-UNL)

Profa. Dra. Ingrid Teixeira Monteiro
Universidade Federal do Ceará - Campus Quixadá
(UFC)

Dedico este trabalho ao meu irmão, minha admirável mãe e ao meu pai, aonde quer que esteja (*in memoriam*).

AGRADECIMENTOS

Agradeço à minha mãe, Bernadete Rabelo, por sempre estar ao meu lado nos bons e maus momentos juntamente do meu irmão, Igor Sousa, duas figuras extremamente importantes na minha vida. Agradeço-lhes por todo incentivo e esforço para que eu pudesse chegar até aqui.

Agradeço também ao meu pai, Antônio Ivo, que mesmo tendo convivido tão pouco, me ensinou coisas imensamente valiosas que irei levar para todo o sempre.

Agradeço ao meu primo, Wenndel Pinheiro, que considero um irmão. Obrigada por todo amor e cuidado.

Agradeço ao Prof. Dr. Enyo José Tavares Gonçalves, por aceitar o desafio de me guiar neste trabalho, pela excelente orientação exercida e pelo desprendimento em me ajudar sempre. E também à todos os docentes da Universidade Federal do Ceará - Campus Quixadá que participaram diretamente da minha formação acadêmica.

Por último, agradeço à todos os colegas que contribuíram na minha formação de alguma forma, em especial os colegas de turma, Daniel e Jéssica. Obrigada por todas as ajudas prestadas e estímulos.

O sucesso nasce do querer, da determinação e persistência em se chegar a um objetivo. Mesmo não atingindo o alvo, quem busca e vence obstáculos, no mínimo fará coisas admiráveis.

(José de Alencar)

RESUMO

Knowledge Acquisition in Automated Specification (KAOS) é uma abordagem da Engenharia de Requisitos Orientada a Objetivos que auxilia os engenheiros na modelagem de requisitos. KAOS é formada por um método, uma linguagem de modelagem e um ambiente de software. Linguagens de modelagem podem ser estendidas de modo a serem adaptadas aos diferentes domínios do software que se pretende modelar. A linguagem de modelagem KAOS vem sendo estendida para diversos domínios/áreas de aplicação como segurança de sistemas, sistemas adaptativos e aspectos. No entanto, identificar se há uma extensão para determinado domínio ou se uma lista de construtores já foi proposta, seja para uso da extensão existente ou para tomá-la como base para a criação de uma nova, pode ser uma tarefa desafiadora. Acreditamos que a linguagem KAOS continuará sendo estendida, portanto, identificar as extensões existentes da KAOS pode ser útil aos pesquisadores que criam novas extensões e aos engenheiros de requisitos que necessitam usar as existentes. O objetivo deste trabalho é apresentar uma Revisão Sistemática da Literatura (RSL) das extensões da KAOS através da identificação e análise das extensões existentes e seus construtores. Os resultados apontam para 22 extensões identificadas e 127 construtores propostos. As extensões e construtores foram analisados de acordo com questões definidas. A classificação das extensões da KAOS por área de aplicação é apresentada como um dos resultados de nossa análise. Os resultados apontam para a necessidade de um desenvolvimento mais completo, consistente e cuidadoso das extensões da KAOS. Finalmente, esse trabalho apresenta discussões e orientações futuras para esta área de pesquisa.

Palavras-chave: Revisão Sistemática de Literatura. Linguagem de Modelagem. Knowledge Acquisition in Automated Specification (KAOS). Extensões.

ABSTRACT

Knowledge Acquisition in Automated Specification (KAOS) is a Goal-Oriented Requirements Engineering approach that supports requirements engineers in modelling the systems. KAOS is composed by a method, a modelling language, and a software tool. Modelling languages can be extended to different domains. The modelling language KAOS has been extended to various domains/application areas such as system security, adaptive systems and aspects. However, identifying whether there is an extension for it domain or a proposed list constructs has been proposed, to use of an existing extension or as a basis for creating a new one, can be a challenging task. We believe the KAOS will continue to be extended, thus to identify existing KAOS extensions can be helpful to researchers who create new extensions and requirements engineers who need to use existing ones. This work aims to present a Systematic Literature Review (SLR) of KAOS extensions through the identification and analysis of existing extensions and their constructs. Results point to 22 identified extensions and 127 proposed constructs. Extensions and constructs were analysed according to defined issues. We classified KAOS extension by application area as one of the results of our analysis. The results point to the need for a more complete, consistent and careful development of KAOS extensions. Finally, this work presents future discussions and guidance for this research area.

Keywords: Systematic Literature Review. Modelling Language. Knowledge Acquisition in Automated Specification (KAOS). Extensions.

LISTA DE FIGURAS

Figura 1 – A sequência de passos para definição ou extensão de uma linguagem de modelagem.	18
Figura 2 – Exemplo do modelo de objetivo KAOS.	22
Figura 3 – Exemplo do modelo de responsabilidade KAOS.	23
Figura 4 – Exemplo do modelo de objeto KAOS.	24
Figura 5 – Exemplo do modelo de operação KAOS.	25
Figura 6 – Modelo de requisitos para a extensão de (FAVERI <i>et al.</i> , 2016).	29
Figura 7 – Processo da revisão sistemática.	33
Figura 8 – Fluxograma de seleção de estudos.	40
Figura 9 – Trabalhos selecionados divididos por ano de publicação.	41
Figura 10 – Trabalhos selecionados por tipo de publicação.	42
Figura 11 – Trabalhos selecionados por tipo de validação.	43
Figura 12 – Trabalhos selecionados por área de aplicação.	45
Figura 13 – Trabalhos selecionados por agrupamento das áreas de aplicação.	45
Figura 14 – Trabalhos selecionados por definição de conceitos.	47
Figura 15 – Representação dos níveis de sintaxe das extensões.	48
Figura 16 – Completude do metamodelo.	49
Figura 17 – Distribuição do tipo de contrutores introduzidos pelas extensões identificadas.	50
Figura 18 – Suporte de ferramenta case.	51

LISTA DE TABELAS

Tabela 1 – Estudos selecionados. 64

LISTA DE QUADROS

Quadro 1 – Comparativo dos trabalhos relacionados e o proposto.	32
Quadro 2 – Questões de pesquisa e suas descrições.	35
Quadro 3 – Fontes de pesquisa automática.	36
Quadro 4 – Termos de pesquisa.	36
Quadro 5 – Lista de critérios de qualidade.	37
Quadro 6 – Resultados da avaliação da qualidade.	41
Quadro 7 – Principais autores com mais artigos selecionados.	42
Quadro 8 – Principais conferências que mais publicaram artigos de extensão.	43
Quadro 9 – Extensões distribuídas por área de aplicação.	44
Quadro 10 – Distribuição de artigos por extensão base.	46
Quadro 11 – Distribuição de artigos por definição de conceitos.	47
Quadro 12 – Distribuição de artigos por nível de sintaxe da extensão.	48
Quadro 13 – Distribuição de artigos por completude de metamodelo.	49
Quadro 14 – Distribuição de extensões por classificação.	50
Quadro 15 – Quantidade de construtores identificados e ferramenta case.	51
Quadro 16 – String de busca para cada base de dados.	60
Quadro 17 – Procedimentos para a busca nas bases de dados.	61
Quadro 18 – Formulário de extração.	62
Quadro 19 – Avaliação de qualidade.	63

LISTA DE ABREVIATURAS E SIGLAS

CATIE	<i>A Catalogue of iStar Extensions</i>
DSML	<i>Domain-Specific Modeling Languages</i>
ER	<i>Engenharia de Requisitos</i>
GMLs	<i>Goal Modeling Languages</i>
GORE	<i>Goal-Oriented Requirements Engineering</i>
GPML	<i>General-Purpose Modeling Languages</i>
KAOS	<i>Knowledge Acquisition in Automated Specification</i>
MBE	<i>Model-Based Engineering</i>
MDE	<i>Model-Driven Engineering</i>
RSL	Revisão Sistemática da Literatura
SI	<i>Sistema de Informação</i>
SLR	Systematic Literature Review
StArt	<i>State of the Art thought Systematic Reviews</i>
UML	<i>Unified Modeling Language</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivos	16
1.2	Organização	16
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	Linguagens de Modelagem: Criação e Extensão	17
2.2	A Linguagem de Modelagem KAOS e suas Extensões	19
2.2.1	<i>O Modelo de Objetivo KAOS</i>	20
2.2.2	<i>O Modelo de Responsabilidade KAOS</i>	22
2.2.3	<i>O Modelo de Objeto KAOS</i>	23
2.2.4	<i>O Modelo de Operação KAOS</i>	24
2.2.5	<i>Glossário da linguagem de modelagem KAOS</i>	26
2.2.6	<i>Obtendo o Documento de Requisitos a partir do Modelo de Requisitos</i>	28
2.2.7	<i>Validando os Requisitos por meio da Análise do Modelo</i>	28
2.2.8	<i>Extensões da linguagem de modelagem KAOS</i>	28
3	TRABALHOS RELACIONADOS	31
3.1	Uma Revisão Sistemática das Extensões de iStar	31
3.2	Análise da Integração de Modelos de Objetivos com Artefatos de Outras Fases	31
3.3	Um Mapeamento da Literatura sobre Engenharia de Requisitos Orientada por Modelos	32
4	PROCEDIMENTOS METODOLÓGICOS	33
4.1	Identificação da Necessidade da Revisão Sistemática	33
4.2	Formulação do Protocolo da RSL	34
4.2.1	<i>Questões de Pesquisa</i>	34
4.2.2	<i>Seleção de Fontes de Busca</i>	35
4.2.3	<i>String de Busca</i>	36
4.2.4	<i>Critérios de Inclusão e Exclusão e Avaliação de Qualidade</i>	37
4.2.5	<i>Seleção de Estudos</i>	38
4.2.6	<i>Extração e Síntese de Dados</i>	38
4.3	Execução da RSL	39

5	RESULTADOS	40
5.1	Visão Geral dos Resultados	41
5.2	QP1: Quais as áreas de aplicação das extensões?	44
5.3	QP2: Para as extensões selecionadas, quais são derivadas de outras extensões?	46
5.4	QP3: Os trabalhos apresentam uma definição de conceitos envolvidos nas extensões?	46
5.5	QP4: Quais os níveis de extensão da KAOS (somente sintaxe concreta, somente sintaxe abstrata ou ambas as sintaxes)?	47
5.6	QP5: Como a extensão da sintaxe abstrata foi proposta (conservativa ou não conservativa)?	48
5.7	QP6: Como a extensão da sintaxe concreta foi proposta?	50
5.8	QP7: Foram definidos mecanismos de extensão para KAOS?	52
5.9	Análises e Discussão	52
6	CONCLUSÕES E TRABALHOS FUTUROS	54
	REFERÊNCIAS	56
	APÊNDICE A – String de Busca para cada Base de Dados	60
	APÊNDICE B – Procedimentos para a Busca nas Bases de Dados	61
	APÊNDICE C – Formulário de Extração	62
	APÊNDICE D – Avaliação de Qualidade	63
	APÊNDICE E – Estudos Selecionados	64

1 INTRODUÇÃO

Linguagens de Modelagem de Objetivos (do inglês, *Goal Modeling Languages* (GMLs)) são linguagens de modelagem visuais usadas principalmente durante os estágios iniciais do desenvolvimento de software, etapa conhecida como *Engenharia de Requisitos* (ER). A ER procura identificar os requisitos que o futuro *Sistema de Informação* (SI) tem que cumprir e as restrições sob as quais ele tem que operar. Não entender os requisitos das partes interessadas ainda é o principal motivo de falha no projeto. Nessa conjuntura, as GMLs são utilizadas para facilitar a identificação, estruturação e validação de requisitos (MATULEVIČIUS; HEYMANS, 2007). Além disso, essas linguagens introduzem novas abstrações, mais notavelmente a noção de meta, para suplementar outras abstrações mais tradicionais usadas na modelagem de dados e processos.

Várias GMLs foram propostas para modelar requisitos, iStar (YU, 1995) e KAOS (DARDENNE *et al.*, 1993) são exemplos dessas linguagens. De acordo com Dardenne *et al.* (1993), Aquisição de Conhecimento em Especificação Automatizada (do inglês, Knowledge Acquisition in Automated Specification (KAOS)) é uma abordagem usada para modelagem de objetivos, premissas, agentes, objetos e operações. KAOS consiste em uma linguagem de modelagem, um método e um ambiente de software (MATULEVIČIUS; HEYMANS, 2007).

As linguagens de modelagem podem ser classificadas como Linguagens de Modelagem de Domínio Específico (do inglês, *Domain-Specific Modeling Languages* (DSML)), que possuem construtores - nós e links - para modelar um determinado domínio e Linguagens de Modelagem de Próposito Geral (do inglês, *General-Purpose Modeling Languages* (GPML)), que são propostas para modelar qualquer domínio (BRAMBILLA *et al.*, 2012). Estender uma linguagem de modelagem é ampliar sua capacidade de representação, e, tanto as DSML quanto as GPML podem ter sua sintaxe alterada por várias razões. No entanto, GPML são mais propensas para extensões do que as DSML, porque elas são propostas para qualquer domínio. Em geral, as DSML são propostas para especificar um único domínio específico e as GPML são propostas para modelar uma grande variedade de sistemas. A linguagem de modelagem KAOS pode ser classificada como GPML.

As extensões enriquecem as linguagens adicionando novos tipos de representações, tornando-as mais adequadas para uma determinada área de aplicação. Nesse contexto, uma série de informações sobre as extensões da KAOS que podem ser úteis aos extensores (pesquisadores que criam extensões) e aos usuários das extensões não estão disponíveis facilmente. A lista das

extensões existentes e os construtores envolvidos nessas extensões, os principais pesquisadores e os problemas que podem estar ocorrendo com essas extensões são exemplos de informações relevantes.

Considerando o cenário apresentado, esta Revisão Sistemática da Literatura (RSL) tem como problema de pesquisa a necessidade de se identificar, analisar e caracterizar extensões da linguagem KAOS, levantando questões de pesquisa que serão apresentadas e respondidas nos capítulos seguintes.

1.1 Objetivos

Este trabalho tem como objetivo geral identificar evidências de como a KAOS vem sendo estendida. Tal objetivo pode ser decomposto nos seguintes objetivos específicos:

- Identificar as extensões da KAOS e analisar seus autores, meios de publicação e áreas de aplicação;
- Analisar as formas de representação dos novos construtores das extensões da KAOS nas sintaxes abstrata e concreta;
- Identificar mecanismos de extensão para KAOS;
- Identificar questões em aberto relacionadas às extensões para KAOS.

1.2 Organização

Este trabalho está organizado da seguinte maneira: o Capítulo 2 apresenta uma visão geral da linguagem de modelagem KAOS e extensão em linguagens de modelagem. Os trabalhos relacionados são descritos no Capítulo 3. O Capítulo 4 detalha o método usado nesta RSL. O Capítulo 5 apresenta os resultados e discussão desta revisão sistemática. Finalmente, no capítulo 6 temos a conclusão.

2 FUNDAMENTAÇÃO TEÓRICA

As seções subsequentes deste capítulo apresentam os elementos essenciais nos quais este estudo se apoia. Na Seção 2.1, são apresentados os conceitos de linguagens de modelagem e a melhor forma de se criar uma nova linguagem de modelagem. A Seção 2.2 traz a definição da linguagem de modelagem KAOS, os modelos e construtores que a compõem e alguns exemplos de extensões criadas para KAOS.

2.1 Linguagens de Modelagem: Criação e Extensão

Segundo Brambilla *et al.* (2012), Engenharia Baseada em Modelos (do inglês, *Model-Based Engineering* (MBE)) é um processo no qual os modelos de software desempenham papel importante, mas eles não são necessariamente os principais artefatos do desenvolvimento. Um exemplo típico que envolve o uso de MBE é um software desenvolvido onde os modelos são criados para documentar o sistema, eles são uma base para o desenvolvimento do software, e nenhuma geração automática de código executável está envolvida. Nesse processo, os modelos ainda desempenham um papel importante, mas não são os artefatos centrais do desenvolvimento.

Engenharia Dirigida por Modelos (do inglês, *Model-Driven Engineering* (MDE)) (BRAMBILLA *et al.*, 2012) é um paradigma de engenharia de software que visa reduzir a complexidade acidental dos sistemas de software, promovendo o uso de modelos que se concentram na complexidade essencial dos sistemas. Nesse caminho, os modelos em MDE são artefatos de primeira classe do processo de desenvolvimento de software, dos quais tipicamente parte significativa da aplicação é derivada.

Mussbacher *et al.* (2014) apontam as linguagens de modelagem como grande área de crescimento no MDE. A Figura 1 ilustra a sequência de atividades necessárias para definir uma nova linguagem de modelagem e, conseqüentemente, ela precisa ser seguida quando uma linguagem de modelagem é estendida. Pesquisadores que trabalham na área de linguagens de modelagem têm focado no Desafio de Abstração: Que tipo de modelagem constroi e que estrutura é necessária para suportar o desenvolvimento de construtores de linguagem de domínio ou de problema que são considerados elementos de modelagem de primeira classe em uma linguagem? (FRANCE; RUMPE, 2007) (Atividade 1 na Figura 1)

Uma linguagem de modelagem é definida por sua sintaxe abstrata (metamodelo e regras de boa formação) e sintaxe concreta. Para Kelly e Tolvanen (2008), construtores de

uma linguagem de modelagem devem ser formalizados e isso é melhor especificado definindo seu metamodelo. O metamodelo de uma linguagem de modelagem mapeia os conceitos da sua área de aplicação para vários elementos da linguagem, suas propriedades e suas conexões, especificadas como links e os papéis que os elementos desempenham neles (Atividade 2 na Figura 1).

Juntamente com os conceitos de modelagem, também devem ser identificadas várias regras de domínio, restrições e necessidades de consistência que uma linguagem deve seguir. Essas regras obviamente precisam ser definidas também. Ter regras na linguagem fornece muitos benefícios para prevenção de erros prontamente, orientação para padrões de design preferíveis, verificação de completude, informando sobre peças em falta, minimiza a ação de modelar o trabalho por convenções e valores padrão e mantém as especificações consistentes (KELLY; TOLVANEN, 2008) (Atividade 3 na Figura 1).

Para uma linguagem de modelagem ser usada por designers de software, é necessário definir um conjunto de modelos e seus elementos gráficos e textuais. Eles são usados para renderizar os elementos do modelo e usar a sintaxe abstrata como ponto de partida para a definição da sintaxe concreta (BRAMBILLA *et al.*, 2012) (Atividade 4 na Figura 1).

Segundo France e Rumpe (2007), ferramentas de modelagem desempenham um papel importante na redução das complexidades associado à compreensão e uso de linguagens de modelagem. A implementação desse tipo de software envolve a representação de sintaxes abstratas e concretas usando frameworks específicos como o Eclipse GMF (Graphical Modeling Framework) (Atividade 5 na Figura 1).

Figura 1 – A seqüência de passos para definição ou extensão de uma linguagem de modelagem.



Fonte: adaptada de (GONCALVES *et al.*, 2018a)

Uma extensão pode ser classificada de acordo com seu impacto na sintaxe original da linguagem, como *conservativa*, que mantém os construtores da sintaxe original, ou *não conservativa*, que altera ou reduz construtores da sintaxe original. Além disso, uma extensão

pode incluir uma proposta para representação de um domínio / área de aplicação em particular ou para melhorar aspectos práticos de uma linguagem. Um fluxo de extensão é composto pelos mesmos passos seguidos para definir uma linguagem de modelagem. O desafio da abstração é tratado fornecendo uma linguagem GPML que tem suporte para personalizar a linguagem para uma área de aplicação específica. Exemplo de personalizações são perfis (por exemplo, perfis *Unified Modeling Language* (UML)), processos de modelagem específicos de domínio, e, em um nível refinado, o uso de formas sintáticas especializadas e restrições de elementos de modelagem específicos. O desafio da formalidade pode ser tratado pelo mapeamento da linguagem de modelagem para uma linguagem formal, ou anotações podem ser adicionadas a linguagem de modelagem no nível de metamodelo para restringir propriedades entre elementos de linguagem (MUSSBACHER *et al.*, 2014).

Goncalves *et al.* (2018a) consideram uma extensão quaisquer alterações realizadas em uma sintaxe abstrata de uma linguagem de modelagem (metamodelo e regras de boa formação), sintaxe concreta ou ambas. Uma extensão que envolve a sintaxe abstrata implica introduzir novas metaclasses, propriedades ou relacionamentos no metamodelo ou criar novas regras de boa formação. Enquanto uma extensão envolvendo sintaxe concreta implica a criação de uma nova representação gráfica de novos nós ou links, pode também envolver uma representação gráfica complementar nos construtores já existentes, como compartimentos ou marcadores textuais (como estereótipos).

Algumas GPML têm mecanismos de extensão bem definidos, como a UML ((MILES; HAMILTON, 2006)). De um modo geral, há duas maneiras de realizar extensões: peso leve e peso pesado. De acordo com Miles e Hamilton (2006), os mecanismos leves da UML são uma maneira de propor extensões com baixo impacto sintático usando marcadores textuais para representar estereótipos, restrições e valores marcados. Extensões de peso pesado introduzem novas representações gráficas e mudam seu metamodelo, por isso tem alto impacto em uma linguagem de modelagem.

2.2 A Linguagem de Modelagem KAOS e suas Extensões

KAOS é uma abordagem da Engenharia de Requisitos Orientada a Objetivos (do inglês, *Goal-Oriented Requirements Engineering* (GORE)), originário de uma cooperação entre a University of Oregon e University of Louvain (DARDENNE *et al.*, 1993). Tal abordagem permite modelar formalmente requisitos funcionais e não funcionais em termos de metas, restrições,

premissas, objetos, eventos, ações, agentes etc. Além disso, essa linguagem de modelagem para ER viabiliza que os analistas construam modelos de requisitos e obtenham documentos de requisitos dos modelos da KAOS (IT, 2007).

A modelagem usando KAOS envolve a construção do modelo de requisitos, por meio das seguintes etapas:

1. Construir um Modelo de Objetivos que descreva os requisitos na forma de objetivos no gráfico;
2. Construir um Modelo de Responsabilidade para atingir esses objetivos através da ajuda de agentes;
3. Criar um Modelo de Objeto junto com a construção de todo o glossário consistente e completo dos termos relacionados ao problema que são usados para escrever os requisitos;
4. Construir um Modelo Operacional descrevendo o comportamento dos agentes responsáveis por atingir os objetivos pelos quais são responsáveis;
5. Criar o documento de requisitos com base no modelo de requisitos;
6. Validar suas necessidades revisando primeiro o modelo.

A KAOS usa a combinação de quatro modelos: modelo de objetivo, modelo de responsabilidade, modelo de objeto e modelo de operação. (FATIMA; FATIMA, 2015). Esses modelos são baseados em objetivos, requisitos, agentes, expectativas, obstáculos, propriedade do domínio, operações, entidades, eventos, relacionamentos e associações entre esses conceitos.

A Respect-IT (acrônimo para Requirements Engineering and SPECification Techniques for Information Technology), uma empresa originária da University of Louvain, detém uma ferramenta chamada *Objectiver* que foi projetada para apoiar a linguagem de modelagem da KAOS. Além disso, a empresa disponibiliza um tutorial de como utilizar tal ferramenta (A KAOS Tutorial (IT, 2007)).

A seguir, descrições detalhadas e um exemplo visual, modelado com o auxílio da ferramenta *Objectiver* e retirado do A KAOS Tutorial ((IT, 2007)), referente à cada modelo da linguagem de modelagem KAOS, apresentação de glossário com conceitos da linguagem e exemplos de extensões da KAOS.

2.2.1 O Modelo de Objetivo KAOS

O modelo de objetivo é considerado a base e o ponto de partida de todo o método. Ele declara os objetivos do sistema composto e, portanto, forma a base para obter todos os outros

modelos por meio desses objetivos. O modelo de objetivo representa um conjunto de diagramas de objetivo inter-relacionados que são usados para lidar com um problema. A ideia principal por trás dessa abordagem é representar os requisitos do sistema como metas e objetivos de negócios e, assim, concentrar-se na realização desses objetivos de negócios.

Objetivos são tipicamente todos os requisitos funcionais e não funcionais que devem ser incorporados no sistema que está sendo desenvolvido, muitas vezes através da assistência de alguns agentes. Os objetivos são a representação das necessidades do cliente na forma de propriedades que o sistema / aplicação deve conter. Essas necessidades podem ser sacadas dos *stakeholders* por meio de quaisquer técnicas de elicitação existentes e listadas na forma das funcionalidades requeridas no software. (FATIMA; FATIMA, 2015)

Após a análise preliminar do sistema e a identificação dos objetivos pelo engenheiro de requisitos, esses objetivos são refinados em objetivos progressivamente mais simples até que possam ser facilmente implementados. Assim, sub-objetivos são derivados dos objetivos de alto nível e são refinados em sub-objetivos mais concretos. Os sub-objetivos podem ser independentes ou estar associados uns aos outros.

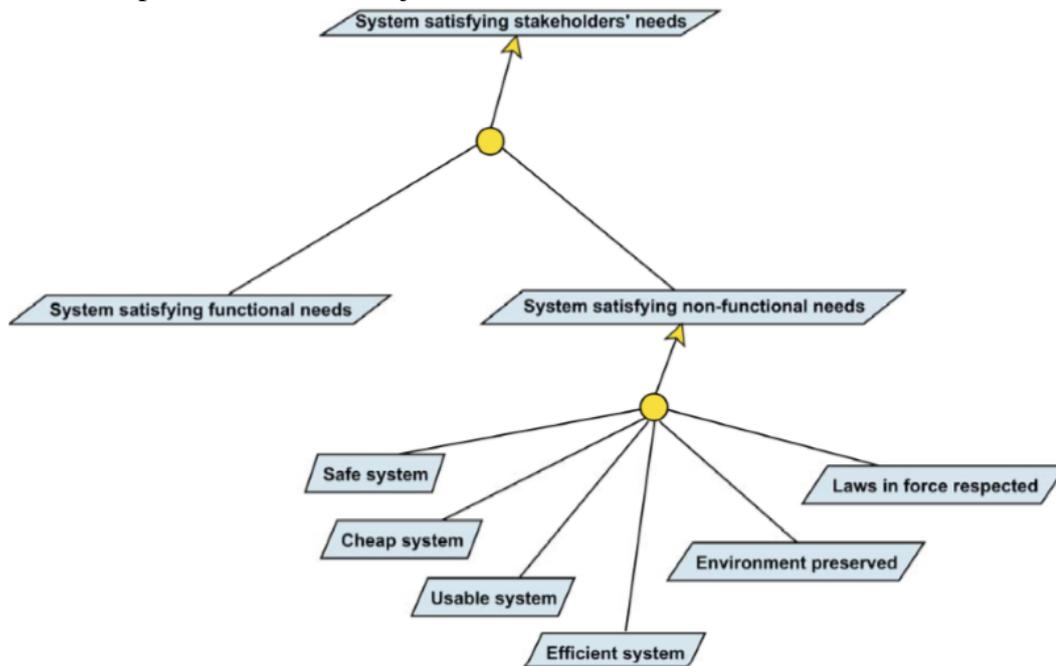
Segundo IT (2007), cada objetivo (exceto as folhas) é refinado como um conjunto de sub-objetivos, informando como o objetivo refinado pode ser alcançado. Os nós principais são objetivos estratégicos para o negócio e os nós de nível inferior representam os requisitos de baixo nível, com a raiz do diagrama sendo os objetivos finais do negócio. O processo de identificação dos penúltimos objetivos seguidos pelos objetivos mais baixos continua até que os analistas atinjam os objetivos básicos. Os objetivos mais baixos são vinculados aos objetivos pai, indicando que a conclusão do objetivo mais baixo com êxito causará a conclusão definitiva de seu objetivo pai.

O gráfico de objetivos tem dois tipos de decomposição de objetivo: a decomposição AND e a decomposição OR. A decomposição AND significa que se todos os sub-objetivos forem alcançados, seu objetivo pai pode ser alcançado ou satisfeito, enquanto que na decomposição OR, a conquista de pelo menos um sub-objetivo leva à conquista de seu objetivo pai. (FATIMA; FATIMA, 2015)

No exemplo da Figura 2, cada paralelogramo na figura representa um objetivo. Os círculos amarelos representam refinamentos de um objetivo principal (apontado pela seta amarela) e uma lista de sub-objetivos.

O diagrama pode ser lido da seguinte maneira: o objetivo do sistema é criar um

Figura 2 – Exemplo do modelo de objetivo KAOS.



Fonte: (IT, 2007)

sistema que satisfaça as necessidades de todas as partes interessadas (funcionais e não funcionais). Os objetivos não funcionais são classificados da seguinte forma:

- O sistema deve ser seguro, barato, utilizável e eficiente;
- O sistema deve preservar seu ambiente;
- O sistema deve respeitar as leis em vigor.

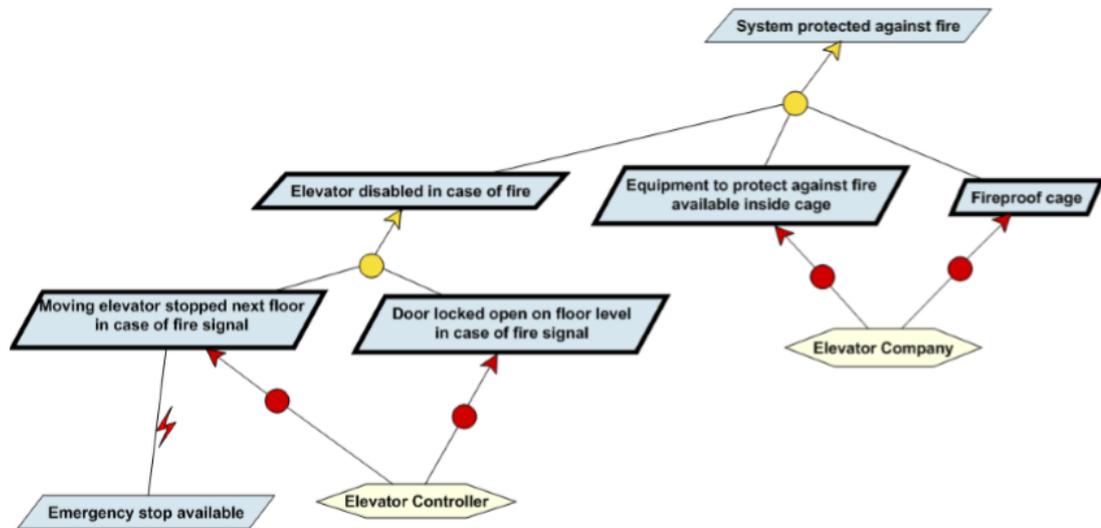
2.2.2 O Modelo de Responsabilidade KAOS

O modelo de responsabilidade da KAOS é uma compilação de diagramas de responsabilidade derivados. Envolve entidades chamadas agentes, que podem ser seres humanos ou componentes automatizados que estão preocupados em atingir os objetivos / requisitos. A atribuição dos agentes para cumprir o objetivo específico é feita de acordo com o modelo de objetivo. Os objetivos são sempre atribuídos a vários agentes. No entanto, sempre que houver uma única resposta do agente para o objetivo, isso indica que não há espaço para qualquer refinamento de objetivo adicional e essa diferença fornece ao analista um critério para parar de refinar objetivos em sub-objetivos. (FATIMA; FATIMA, 2015)

Um diagrama de responsabilidade descreve para cada agente, os requisitos e expectativas pelas quais ele é responsável ou que foram atribuídos a ele. No exemplo da Figura 3, cada hexágono na figura representa um agente. Paralelogramos com bordas em negrito são requisitos,

objetivos atribuídos aos agentes do sistema. Os círculos vermelhos representam responsabilidade, relacionamento que conecta um agente (apontado pela seta vermelha) a um requisito pelo qual ele é responsável. O símbolo do raio vermelho retrata conflito entre os objetivos, ou seja, sob alguma condição de limite, os objetivos não puderem ser alcançados.

Figura 3 – Exemplo do modelo de responsabilidade KAOS.



Fonte: (IT, 2007)

O diagrama do Sistema de Proteção contra Incêndio de um elevador pode ser lido da seguinte maneira:

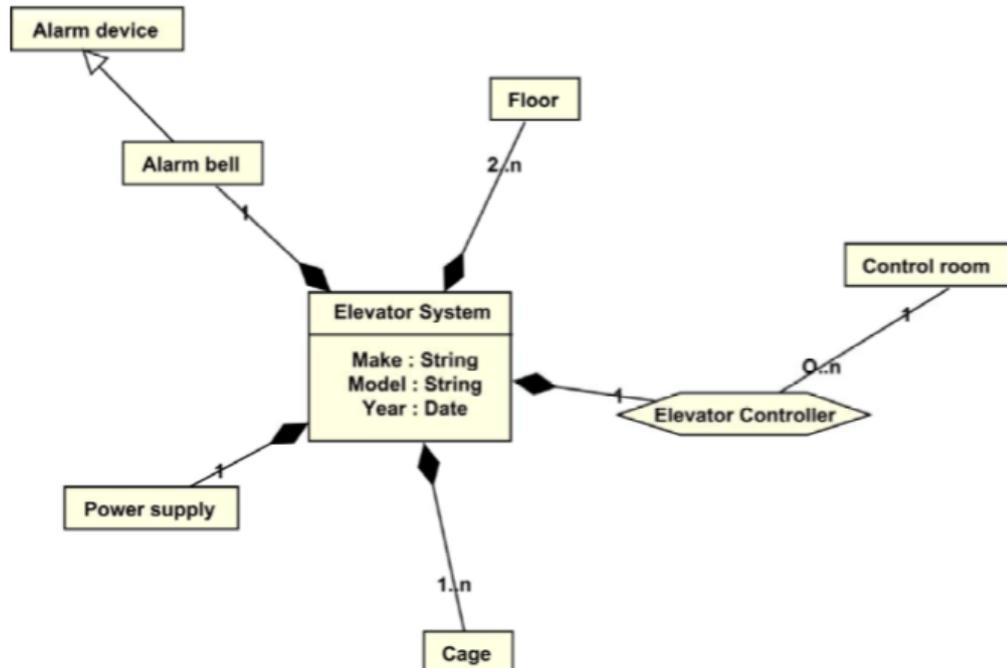
- O agente *Elevator controller* é responsável por manter as portas trancadas quando um alarme de incêndio é acionado e por interromper todos os elevadores em movimento assim que possível. Eles atingem qualquer piso;
- O agente *Elevator company* que instalará o sistema do elevador é responsável por fornecer equipamentos à prova de fogo (gaiola e equipamento).

2.2.3 O Modelo de Objeto KAOS

De acordo com Fatima e Fatima (2015), o modelo de objeto está basicamente relacionado à vinculação do domínio da aplicação e ao estabelecimento de restrições no sistema. Os objetos podem ser categorizados como entidades, agentes e associações onde *entidades* descrevem e traduzem o estado do objeto, mas não realizam operações; os *agentes* estão preocupados com a execução das operações, enquanto *associações* são entidades que dependem do objeto e não têm a capacidade de realizar as operações.

No exemplo da Figura 4, a caixa dupla na figura representa uma entidade. Cada retângulo é um objeto. As demais setas são ligações entre entidade-objeto, entidade-agente, objeto-objeto e agente-objeto.

Figura 4 – Exemplo do modelo de objeto KAOS.



Fonte: (IT, 2007)

O diagrama descreve os componentes de um sistema de Elevador e pode ser lido da seguinte maneira: o sistema do elevador (*elevator system*) é composto pelos componentes

- Uma ou várias gaiolas (*cage*);
- Pelo menos 2 andares (*floor*);
- Uma campainha de alarme (*alarm bell*), que é uma especialização de um dispositivo de alarme (*alarm device*);
- Um controlador de elevador (*elevator controller*, localizado em uma sala de controle (*control room*);
- Uma fonte de alimentação (*power supply*).

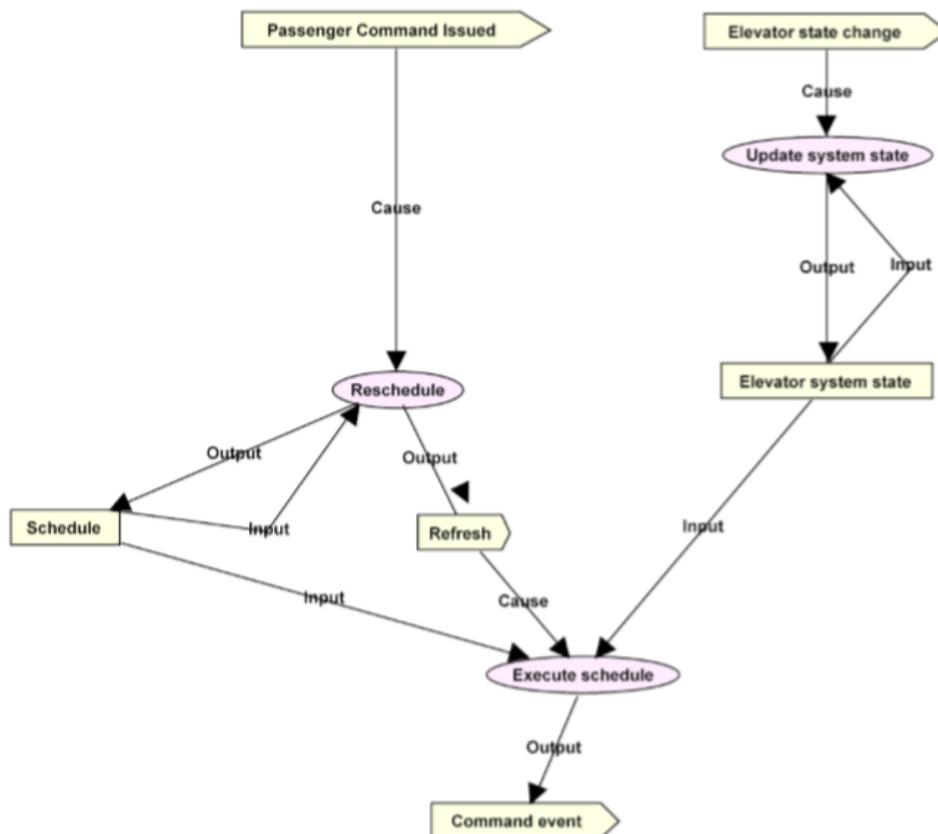
2.2.4 O Modelo de Operação KAOS

O modelo de operação representa todos os comportamentos que os agentes devem ter para atender suas necessidades. Comportamentos são basicamente operações realizadas por agentes. Essas operações são usadas para manipular os objetos descritos no modelo de

objeto: eles podem criar objetos, provocar transições de estado de objeto ou acionar outras operações através de eventos enviados e recebidos (IT, 2007). Essas operações dependem das necessidades das partes interessadas. Na KAOS, um diagrama de operação normalmente compreende operações realizadas por um ou vários agentes para atender um requisito. As composições são feitas por meio de fluxos de dados (a saída de uma operação se torna a entrada de outra operação) ou fluxo de controle (um evento enviado por uma operação aciona ou interrompe outra operação). Um diagrama de operações, portanto, descreve como os agentes precisam cooperar para que o sistema funcione. (FATIMA; FATIMA, 2015)

No exemplo da Figura 5, as operações são representadas por elipses. Objetos em questão são conectados às operações por meio dos links de entrada e saída. Eventos são representados como sinais de trânsito usados para indicar direções. Eventos podem ser externos ou produzidos por operações (eles são feitos por uma saída da operação). Eles podem iniciar (causar) ou parar as operações. Por exemplo, no diagrama, o evento "Refresh" é produzido pela operação "Reschedule" e inicia a execução da operação "Execute schedule".

Figura 5 – Exemplo do modelo de operação KAOS.



Fonte: (IT, 2007)

O diagrama descreve as operações de um sistema de Elevador e pode ser lido da seguinte maneira: quando um passageiro aperta um botão do sistema do elevador, o sistema atualiza a lista de instruções (*Reschedule*) que o controlador do elevador deve executar. O novo cronograma será imediatamente utilizado pelo controlador do elevador. Para executar um cronograma, o controlador do elevador precisa conhecer o cronograma e o estado mais recente do sistema de elevadores. O estado do sistema do elevador é atualizado graças a outro evento específico (*Elevator state change*).

2.2.5 Glossário da linguagem de modelagem KAOS

O glossário a seguir explica brevemente um conjunto de conceitos que se pode encontrar na linguagem KAOS.

- **Agente:** Objeto ativo que realiza operações para atingir metas. Agentes podem ser o software considerado como um todo ou partes dele. Também podem vir do ambiente do software estudado. Agentes humanos estão no ambiente;
- **Associação:** Associações são objetos passivos dependentes. *Dependente* porque suas descrições referem-se a outros objetos. Objeto cuja definição depende de outros objetos vinculados pela associação;
- **Sistema Composto:** O software em estudo e seu ambiente;
- **Conflito:** Os objetivos são conflitantes se, sob alguma condição de limite, estes objetivos não puderem ser alcançados;
- **Propriedade de Domínio:** São propriedades relevantes para o domínio da aplicação. Eles são usados em refinamentos para provar que o refinamento está completo;
- **Entidade:** Representam objetos passivos independentes. *Independente* significa que suas descrições não precisam se referir a outros objetos do modelo. *Passivos* significa que eles não podem executar operações. Objeto autônomo, isto é, a definição de qual não depende de outros objetos;
- **Ambiente:** Parte do universo capaz de interagir com o software em estudo;
- **Evento:** Os eventos podem ser externos ou produzidos por operações (são a saída da operação). Eles podem começar (causar) ou parar operações. É um objeto instantâneo (isto é, um objeto ativo em apenas um estado) que aciona operações executadas por agentes;
- **Expectativa:** Objetivo atribuído a um agente no ambiente;
- **Modelo Formal:** Modelo em que os conceitos foram matematicamente formalizados. No

Objectiver, o modelo formal é construído sobre um modelo semi-formal. Apenas uma parte ou todo o modelo semi-formal pode ser formalizado. O Objectiver usa lógica temporal de primeira ordem com extensões em tempo real para formalizar conceitos;

- **Objetivo:** Asserção prescritiva que captura algum objetivo a ser alcançado pela cooperação de agentes. Prescreve um conjunto de comportamentos desejados. Requisitos e expectativas são objetivos;
- **Modelo:** Representação abstrata de um sistema composto. Um modelo no Objectiver representa um sistema composto por meio de conceitos de diferentes tipos, principalmente objetos, propriedades desejadas ou indesejáveis (objetivos, obstáculos) e comportamentos (operações);
- **Objeto:** Artefato de interesse no sistema composto sendo modelado, cujas instâncias podem ser distintamente identificadas e podem evoluir de estado para estado. Agentes, eventos, entidades e associações são objetos;
- **Obstáculo:** Condição (que não seja um objetivo) cuja satisfação pode impedir que algum objetivo seja alcançado. Define um conjunto de comportamentos indesejados;
- **Operação:** Especifica transições de estado de objetos, que são entradas e / ou saídas da operação. Operações são realizadas por agentes;
- **Operacionalização:** Relacionamento que vincula um requisito às operações. É requerido quando cada execução das operações (possivelmente restrita a essa intenção) implicará o requisito;
- **Refinamento:** Relacionamento que vincula um objetivo a outros objetivos que são chamados de sub-objetivos. A conjunção de todos os sub-objetivos deve ser uma condição suficiente que implique o objetivo, eles refinam objetivos. Aponta para um objetivo pai;
- **Requisito:** Objetivo atribuído a um agente do software em estudo;
- **Responsabilidade:** Relacionamento que conecta um agente a um requisito pelo qual ele é responsável;
- **Modelo Semi-Formal:** Modelo no qual os conceitos não são matematicamente formalizados. No Objectiver, cada conceito no modelo recebe um nome, um tipo, uma definição textual, valores para atributos e uma representação gráfica.

2.2.6 *Obtendo o Documento de Requisitos a partir do Modelo de Requisitos*

Na KAOS, existe um documento de modelo que contém todas as informações extraídas dos quatro modelos para especificar o documento de requisitos. A parte do glossário é derivada do modelo de objeto; os requisitos são especificados de acordo com o objetivo do topo (os objetivos comerciais / estratégicos) para o fundo (os requisitos). Requisitos na arquitetura do sistema são obtidos a partir do modelo de responsabilidade e requisitos para o comportamento do sistema a partir do modelo de operação. Todo esse processo leva à extração dos requisitos do sistema e, finalmente, resulta em um documento completo, consistente e sem ambiguidade. Assim, a saída dos quatro modelos é um documento de requisitos completo. (FATIMA; FATIMA, 2015)

2.2.7 *Validando os Requisitos por meio da Análise do Modelo*

Os requisitos recolhidos através da abordagem KAOS podem ser validados revedo- os de forma a construir um produto de alta qualidade organizando revisões coletivas do modelo KAOS, ao invés de pedir às pessoas que leiam um longo documento técnico por conta própria e depois marcar reuniões para discutir suas observações. Reuniões virtuais usando telas compartilhadas em locais distantes das partes interessadas também podem ser usadas. (IT, 2007)

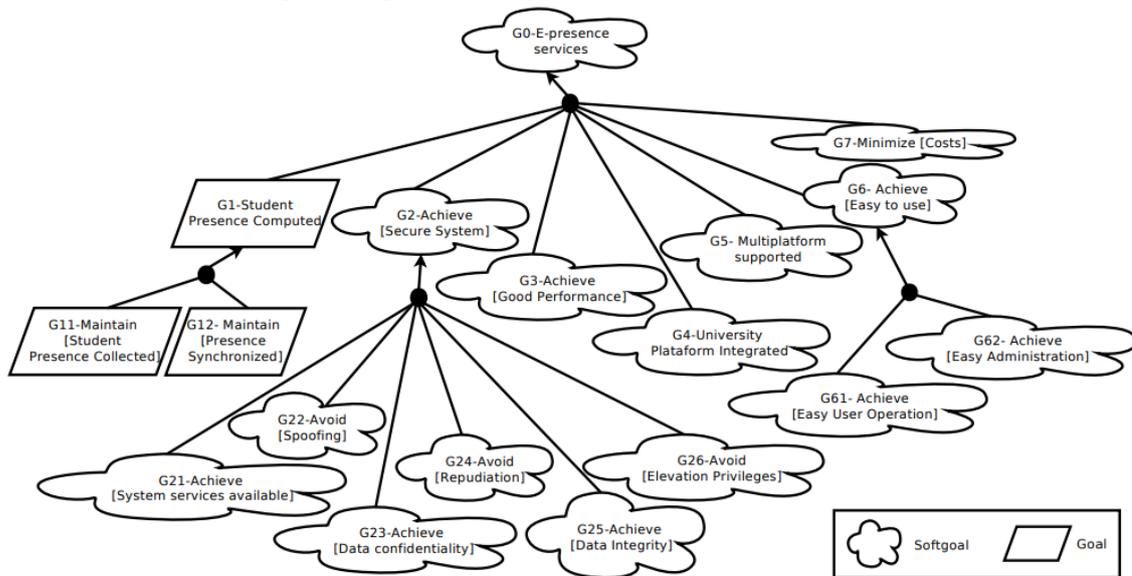
2.2.8 *Extensões da linguagem de modelagem KAOS*

Algumas extensões da linguagem KAOS foram identificadas e utilizadas durante o teste da string de busca para validar a consistência da string, verificando se a busca as retornavam. Nesta subseção, elas serão apresentadas para exemplificar as extensões da KAOS.

A abordagem de Faveri *et al.* (2016) propõe um método sistemático para incorporar técnicas e táticas de *deception* no design de software. Deception é um conceito de segurança de sistemas que consiste em aplicar técnicas para ludibriar invasores mal intencionados. Ele integra meta e modelos de ameaças construídos durante atividades de ER e produz um modelo de variabilidade expressando um repertório de estratégias de deception. Os autores escolheram a KAOS para modelar o sistema e-Presence (sistema de controle de presença de alunos na sala de aula e evitar fraude durante os exames, e que foi requerido pelo conselho de administração da Universidade Nova de Lisboa (FCT/Nova)) porque a linguagem suporta modelagem de ameaças. Na KAOS, os objetivos representam propriedades de um sistema que são satisfeitas por diferentes

agentes. O refinamento do objetivo na KAOS cessa quando a responsabilidade da satisfação do objetivo pode ser atribuída a um único agente. Esse objetivo representa um requisito. Objetivos satisfeitos por agentes humanos são tratados como expectativas. Na Figura 6 é possível observar um modelo de requisitos proposto nessa extensão.

Figura 6 – Modelo de requisitos para a extensão de (FAVERI *et al.*, 2016).



Fonte: retirada de (FAVERI *et al.*, 2016).

Essa figura mostra uma representação parcial dos objetivos de alto nível do sistema e-Presence. Os objetivos de nível superior são descritos de *Student Presence Computed* (G1) para *Minimize [Costs]* (G7). O predicado *Maintain* denota um objetivo comportamental especificando uma propriedade que deve sempre ser mantida. O inverso do objetivo *Maintain* é o objetivo *Avoid*. Conseqüentemente o objetivo *Maintain [Presence Synchronized]* (G12) pode ser indicado pelo objetivo *Avoid [Presence Unsynchronized]*. Outra classe de objetivos comportamentais é denotada pelo predicado *Achieve*, indicando uma propriedade que eventualmente deve reter. *Cease* representa uma propriedade que deve ser eliminada no futuro. Objetivos flexíveis podem ser prefixados pelos predicados do usuário para indicar padrões distintos (por exemplo, *Minimize [..]*, *Maximize [..]*, *Improve [..]*) (FAVERI *et al.*, 2016).

O modelo estendido ilustrado na Figura 6 nos apresenta novas representações para os elementos presentes no modelo KAOS. Para representar objetivos, os extensores utilizaram paralelogramos brancos, subobjetivos são representados por nuvens e refinamentos são setas com círculos pretos. Outras extensões propostas para outros tipos de domínios podem ser vistas

abaixo.

Em Semmak *et al.* (2008), os autores propõem uma extensão ao metamodelo da KAOS introduzindo conceitos de variabilidade a fim de poder especificar o *Modelo da Família de Requisitos*. Esse modelo será habilitado para derivar diferentes modelos específicos de acordo com as necessidades das partes interessadas. Neste trabalho, a proposta de extensibilidade é planejada para o domínio Cycab. Cycabs são pequenos veículos elétricos, projetados para zonas de acesso restrito (centros históricos da cidade, aeroportos ou estações de trem). As extensões feitas para KAOS devem mostrar *o que varia* nos sub-modelos KAOS e *como isso varia*. O *o que varia* é representado graças ao conceito de ponto de variação, e *como isso varia* é descrito pelos conceitos de faceta e variante. Esses conceitos formam o que é chamado de modelo de variabilidade e o ponto de variação relaciona esse modelo com os sub-modelos KAOS.

Lamddi (2017) apresenta uma abordagem para aplicar um método estruturado à integração entre *Safety* e *Security*, criando um modelo de domínio de Segurança e Proteção (SaS). Além disso, demonstra que é possível usar a linguagem orientada a objetivos KAOS na análise de ameaças e perigos para cobrir os domínios de segurança e proteção, fazendo com que saídas, ou artefatos, sejam bem estruturados e abrangentes, o que resulta em confiabilidade devido à abrangência da análise. A abordagem estruturada pode, assim, atuar como uma interface para interações ativas no gerenciamento de risco e perigo em termos de cobertura universal, encontrando soluções para diferenças e contradições que podem ser superadas integrando os domínios de segurança e proteção e usando uma técnica de análise unificada de sistema KAOS que vai resultar na centralidade da análise. Aplicaram o alinhamento entre os modelos de domínio de segurança e proteção (SaS) para o gerenciamento de riscos com a linguagem de modelagem KAOS, o que deu a possibilidade de um melhor método para derivar os requisitos de segurança e proteção nos estágios iniciais desde o início do ciclo de vida de desenvolvimento do sistema. O alinhamento entre o modelo de domínio SaS e a KAOS aumenta a cooperação e facilita a comunicação e a interação entre as partes interessadas.

3 TRABALHOS RELACIONADOS

A seguir, alguns estudos relacionados ao presente trabalho, sendo uma RSL sobre as extensões da linguagem de modelagem iStar, uma análise de integração de modelos de objetivos com artefatos de outras fases e um mapeamento sobre GORE.

3.1 Uma Revisão Sistemática das Extensões de iStar

Em Goncalves *et al.* (2018a), os autores procuram entender como a linguagem de modelagem iStar é estendida através da identificação e análise das extensões existentes e seus nós e links. Nesse trabalho, foi conduzida uma RSL para identificar, investigar e caracterizar extensões do iStar. Pesquisa automática e manual foram utilizadas. Além disso, a prática de *snowballing* com os autores selecionados foi feita com base em trabalhos já avaliados como relevantes para capturar documentos não identificados na pesquisa automática. Através dessa prática as referências bibliográficas de cada trabalho selecionado foram analisadas. Especialistas em extensões iStar também foram contactados. A RSL identificou 1964 trabalhos iniciais. Após passar pelas etapas da RSL, a extração dos dados foi realizada em 96 artigos. Nos resultados, são apresentadas as extensões do iStar e suas evoluções, segmentadas por áreas de aplicação. Os resultados ainda apontam para a necessidade de um desenvolvimento mais completo, consistente e cuidadoso das extensões do iStar. A metodologia de pesquisa seguida nesta RSL é a mesma utilizada em Goncalves *et al.* (2018a). O escopo de Goncalves *et al.* (2018a) é semelhante ao estudo desta RSL, uma vez que o objetivo deste trabalho é identificar evidências de como a linguagem de modelagem KAOS vem sendo estendida. O que diferencia os dois é a linguagem de modelagem alvo de cada RSL, pois em Goncalves *et al.* (2018a) o foco é o iStar, enquanto aqui é a KAOS.

3.2 Análise da Integração de Modelos de Objetivos com Artefatos de Outras Fases

O trabalho de Horkoff *et al.* (2015) relata sobre os progressos realizados na integração de modelos de objetivos com artefatos de outras fases de desenvolvimento de software. Os autores apresentam uma revisão sistemática sobre o assunto para fornecer roteiro para abordagens que mapeiam, transformam ou integram modelos de objetivos a outros artefatos do ciclo de vida. Complementarmente, identificaram 99 artigos por *snowballing* para análise. Os resultados mostram que existe uma grande variedade de propostas com uma variedade de modelos de

objetivos e metas, cobrindo múltiplos paradigmas, motivados por diversos propósitos. Os autores concluem que, embora muito trabalho tenha sido feito nessa área, ele é fragmentado, seguindo várias vertentes separadas de orientação a objetivos, e muitas vezes ainda está em estágios iniciais de maturidade. A metodologia de pesquisa seguida nesta RSL é semelhante a utilizada em Horkoff *et al.* (2015). O escopo do artigo é diferente do estudo desta RSL, uma vez que este trabalho não se preocupa com a análise de desenhos de transformações para fatos de outras fases do processo de desenvolvimento. Além disso, a linguagem envolvida na pesquisa do artigo são todas as linguagens de modelagem baseadas em objetivos, enquanto o foco aqui está na KAOS.

3.3 Um Mapeamento da Literatura sobre Engenharia de Requisitos Orientada por Modelos

Em Horkoff *et al.* (2016), um mapeamento sistemático da literatura forneceu visão geral da GORE. O trabalho apresenta o progresso da GORE e traz uma visão geral da abordagem, tópicos e linguagens de modelagem envolvidas. Foi mostrado um gráfico com a evolução das extensões da GORE, mas não analisaram como essas extensões foram realizadas. O mapeamento abrange 246 trabalhos mais citados, de acordo com o Scopus, de conferências e periódicos relacionados ao GORE. Os resultados da análise foram usados para fazer recomendações sobre futuras pesquisas da GORE. Esse trabalho relaciona-se com a presente pesquisa pois faz um mapeamento em publicações, categorizando e analisando as informações dos estudos encontrados, a fim de responder a várias questões de pesquisa. Porém, não há diferenciação de linguagens de modelagem nas extensões, uma vez que o artigo cobre todas as linguagens de modelagem que as pesquisas conseguiram trazer.

Abaixo, o Quadro 1 apresenta uma análise comparativa dos trabalhos relacionados apresentados neste capítulo e esta RSL.

Quadro 1 – Comparativo dos trabalhos relacionados e o proposto.

Critério	(GONCALVES <i>et al.</i>, 2018a)	(HORKOFF <i>et al.</i>, 2015)	(HORKOFF <i>et al.</i>, 2016)	Trabalho Proposto
Revisão Sistemática da Literatura	X	X		X
Identifica e Analisa extensões	X		X	X
KAOS				X

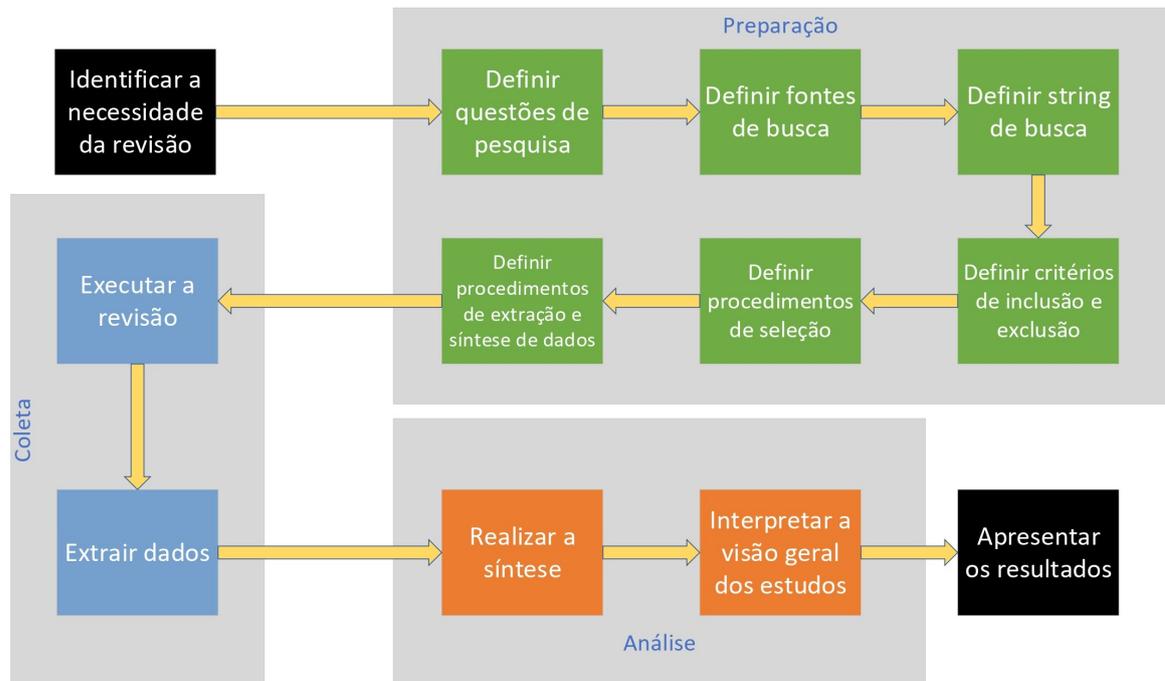
Fonte: elaborado pela autora.

4 PROCEDIMENTOS METODOLÓGICOS

Este capítulo mostra as etapas da realização da pesquisa definidas neste trabalho. A Figura 7 mostra o processo que foi seguido na construção desta pesquisa. Os passos seguidos podem ser resumidos da seguinte forma:

1. Identificação da necessidade de uma revisão sistemática;
2. Formulação do protocolo da RSL (preparação);
3. Execução da RSL e extração de dados (coleta);
4. Síntese e compilação de dados (análise);
5. Apresentação dos resultados obtidos.

Figura 7 – Processo da revisão sistemática.



Fonte: elaborada pela autora.

4.1 Identificação da Necessidade da Revisão Sistemática

A primeira etapa deste estudo consistiu na identificação de necessidade da execução da RSL. Essa necessidade surgiu como uma forma de fazer uma análise da KAOS semelhante a realizada na RSL das extensões de iStar (GONCALVES *et al.*, 2018a). Em seguida, foram identificadas algumas extensões da KAOS e foram realizados alguns testes iniciais de *strings* de

busca, que mostraram ser viável a execução desta RSL. Conseqüentemente, esta fase consiste na revisão de literatura da RSL, bem como definição do tema, objetivos e metodologia.

De acordo com BA e Charters (2007), uma RSL tem como objetivo identificar, avaliar e interpretar os resultados de uma determinada pesquisa relacionada a questões, área temática, ou fenômeno e reunir evidências para basear conclusões. Para a condução desta RSL foram seguidas as diretrizes definidas por (BA; CHARTERS, 2007) e sua versão atualizada (KITCHENHAM; BRERETON, 2013).

Os conceitos relacionados a este estudo foram apresentados nas Seções 2.1-2.2. Considera-se extensões novas versões da linguagem com novos construtores (nós, links ou ambos) em qualquer nível de sintaxe (abstrato, concreto ou ambos).

A ferramenta de software *State of the Art thought Systematic Reviews* (StArt) (LAPES, 2010) foi usada para apoiar a execução desta RSL. Essa ferramenta é usada para guiar pesquisadores conduzindo revisões. StArt foi avaliada empiricamente e foi demonstrado que ela teve resultados positivos na execução de revisões sistemáticas (HERNANDES *et al.*, 2012).

4.2 Formulação do Protocolo da RSL

O protocolo desta RSL envolveu a determinação das questões de pesquisa, fontes de seleção, *string* de busca, critérios de análise de qualidade e critérios de extração de dados.

Nas seções seguintes, são detalhadas as atividades desse protocolo e o resultado final determinado em cada uma delas que serviram de base para as etapas seguintes.

4.2.1 Questões de Pesquisa

KAOS é uma abordagem da GORE muito popular. Além disso, a linguagem de modelagem KAOS é uma GPML, por isso, várias extensões são propostas à ela. Assim, esta RSL pretende responder a seguinte questão de pesquisa principal: Quais são as extensões existentes da KAOS?

Essa questão é o ponto chave para identificar as extensões da KAOS e é o ponto de partida para analisar como essa linguagem é estendida. Os resultados dessa questão principal de pesquisa podem ajudar extensores da KAOS a ter uma melhor compreensão sobre como estendê-la, assim como pode contribuir com os demais usuários de extensões da KAOS. Também é útil aos pesquisadores, pois ajuda-os a definir uma forma sistemática de estender a KAOS e

criar mecanismos de extensão padrão em novas versões da linguagem.

Posteriormente, na apresentação geral dos resultados, essa questão será respondida e tratada mais especificamente pelas questões de pesquisa que se seguem, conforme listado no Quadro 2. Algumas questões de pesquisa foram adaptadas do trabalho de (GONCALVES *et al.*, 2018a) e outras foram elaboradas pelos autores desta RSL.

Quadro 2 – Questões de pesquisa e suas descrições.

Questão de Pesquisa	Descrição e Motivação
QP1. Quais as áreas de aplicação das extensões?	O objetivo desta pergunta é classificar as extensões e agrupá-las de acordo com a área.
QP2. Para as extensões selecionadas, quais são derivadas de outras extensões?	Esta pergunta busca verificar se as extensões encontradas são derivadas de outras extensões.
QP3. Os trabalhos apresentam uma definição de conceitos envolvidos nas extensões?	Esta pergunta pretende detectar quais extensões apresentaram a descrição dos conceitos introduzidos nas extensões. A descrição dos conceitos melhora o entendimento da extensão proposta.
QP4. Quais os níveis de extensão da KAOS (somente sintaxe concreta, somente sintaxe abstrata ou ambas as sintaxes)?	Esta questão visa identificar o nível que as extensões para KAOS abordam.
QP5. Como a extensão da sintaxe abstrata foi proposta (conservativa ou não conservativa)?	O objetivo desta pergunta é analisar a sintaxe abstrata das extensões. Se há representação completa dos conceitos da KAOS no metamodelo das extensões e a classificação das extensões (conservativa ou não conservativa).
QP6. Como a extensão da sintaxe concreta foi proposta?	Esta questão tem como objetivo analisar a representação gráfica dos construtores introduzidos em sintaxe concreta. Os construtores são classificadas de acordo com a sua utilização. Também analisamos extensões que propunham ferramentas CASE (Computer-Aided Software Engineering) para modelar as extensões.
QP7. Foram definidos mecanismos de extensão para KAOS?	O objetivo desta questão é verificar se existem mecanismos de extensão definidos em KAOS.

Fonte: elaborado pela autora.

4.2.2 Seleção de Fontes de Busca

Uma busca automática foi realizada em bases de dados eletrônicas. Essas bases de dados são listadas no Quadro 3. Tais bases foram selecionadas porque incluem periódicos e conferências de qualidade na área de engenharia de software. Os critérios de seleção foram:

- (i) Todas as bases de dados devem estar disponíveis para acesso completo dos seus repositórios pela Internet na Universidade Federal do Ceará;
- (ii) Devem ter operadores "AND" e "OR";

- (iii) Devem ser relevantes para a ciência da computação / engenharia de software;
- (iv) Devem permitir exportar dados em formato *BibTex* ou *CSV*.

Quadro 3 – Fontes de pesquisa automática.

Fonte	Site
ACM	https://dl.acm.org/
El Compedex	https://www.engineeringvillage.com/
IEEE Xplore	https://ieeexplore.ieee.org/
Science Direct	https://www.sciencedirect.com/
SCOPUS	https://www.scopus.com/
Springer	http://link.springer.com/

Fonte: elaborado pela autora.

4.2.3 *String de Busca*

O período de busca começa em 1990, quando a KAOS foi proposta por Axel van Lamsweerde, e termina em 2019, ano atual no qual esta RSL está sendo realizada. A pesquisa foi realizada com base nos termos apresentados no Quadro 4. Esses termos de pesquisa foram combinados na seguinte forma: (T1 e T2 e T3 e (T4 ou T5)). A string foi adaptada para cada base de dados eletrônica; é possível visualizar essas adaptações e o passo-a-passo realizado na busca nos Apêndices A e B. KAOS de forma abreviada e sua ampliação 'Knowledge Acquisition in Automated Specification' foram utilizados como termos (de população) relacionados à linguagem de modelagem. A segunda parte da cadeia de caracteres (T2 e T3) está relacionada à área de aplicação de termos (de intervenção) como requisitos, modelagem, modelagem de objetivos e objetivo orientado. A terceira parte da cadeia de caracteres (T4 e T5) está relacionada a termos relacionados para extensões ou perfis.

Quadro 4 – Termos de pesquisa.

Termo de Classificação	Palavras Relacionadas
Termos de População	T1: "Kaos"OR "Knowledge Acquisition in Automated Specification"
Termos de Intervenção	T2: "Requirements", T3: "Goal modeling"OR "Goal modelling"OR "Goal-oriented", T4: "Extension"OR "Extends"OR "Extended"OR "Extensibility", T5: "Patterns"OR "Profile"OR "Approach"OR "Process"

Fonte: elaborado pela autora.

4.2.4 Critérios de Inclusão e Exclusão e Avaliação de Qualidade

A lista de critérios de inclusão para selecionar estudos para responder às questões de pesquisa é a seguinte:

1. Artigos escritos em inglês;
2. Publicações a partir de 1990 até 2019;
3. Estudos primários, ou seja, peças de pesquisa original;
4. Artigos que estendem a KAOS, ou seja, aqueles que incluíram novos conceitos com impacto em um ou mais níveis de linguagem;
5. Artigos relacionados a mecanismos de extensão, padrões de extensão e perfis.

Os critérios de exclusão de artigos não relacionados às questões de pesquisa foram:

1. Estudos secundários, ou seja, outras RSL's ou mapeamentos sistemáticos não foram selecionados;
2. Estudos que não definiram extensões, mecanismos de extensibilidade ou processo de extensão para KAOS, mas apenas o uso da KAOS sem mudanças em sua sintaxe. Trabalhos que apresentam apenas um estudo de caso da KAOS sem extensão são exemplos de artigos excluídos por este critério;
3. Trabalhos duplicados foram excluídos.

Os Critérios de Avaliação da Qualidade (CAQ) são apresentados no Quadro 5.

Quadro 5 – Lista de critérios de qualidade.

ID	Perguntas	Possíveis Respostas
CAQ1	É um trabalho de pesquisa ou um relatório de lições aprendidas? (DYBÅ; DINGSØYR, 2008)	SC, NS
CAQ2	Existe motivação para a realização do estudo? (DYBÅ; DINGSØYR, 2008)	SC, NS, SP
CAQ3	Os objetivos da pesquisa estão bem definidos? (DYBÅ; DINGSØYR, 2008)	SC, NS, SP
CAQ4	A extensão/mecanismos de extensão ou processo está bem definido? (GONCALVES <i>et al.</i> , 2018a)	SC, NS, SP
CAQ5	O contexto (industrial ou acadêmico) é descrito? (DYBÅ; DINGSØYR, 2008)	SC, NS, SP
CAQ6	O estudo foi empiricamente testado? (DERMEVAL <i>et al.</i> , 2016)	SC, NS
CAQ7	Os resultados foram discutidos? (DERMEVAL <i>et al.</i> , 2016)	SC, NS, SP
CAQ8	As limitações foram discutidas? (DING <i>et al.</i> , 2014)	SC, NS, SP

Fonte: elaborado pela autora.

As perguntas CAQ1, CAQ2, CAQ3, CAQ5, CAQ6, CAQ7 e CAQ8 foram adaptadas dos estudos de (DERMEVAL *et al.*, 2016), (DING *et al.*, 2014) e (DYBÅ; DINGSØYR, 2008), enquanto o CAQ4 foi proposto por (GONCALVES *et al.*, 2018a). Os valores são 0 - Não satisfaz (NS), 0,5 - Satisfaz parcialmente (SP) e 1 - Satisfaz completamente (SC). Os critérios CAQ1 e CAQ6 permitem satisfazer completamente ou não satisfazer como resposta e os outros CAQ permitem satisfazer parcialmente como terceira opção.

4.2.5 Seleção de Estudos

A seleção dos artigos se dá em 3 etapas, sendo auxiliada pela ferramenta StArt. Inicialmente, os artigos duplicados e redundantes são removidos por uma verificação automática no StArt. Essa ferramenta tem um recurso que verifica a porcentagem de similaridade entre os artigos identificados. O próximo passo é a leitura do título, resumo e palavras-chave de todos os trabalhos restantes selecionados e não duplicados. Os critérios de seleção e exclusão são utilizados nessa avaliação e trabalhos que não tiveram uma decisão em ser incluído ou excluído são mantidos. O último passo na seleção é a leitura completa dos documentos e aplicação dos critérios de inclusão e exclusão e avaliação de qualidade.

O item número 3 dos critérios de exclusão cita estudos que não definiram extensões para KAOS. Não são considerados como uma extensão qualquer trabalho que usou a KAOS sem mudanças na sintaxe abstrata (mudanças no metamodelo ou regras de validação) e sintaxe concreta (nova representação gráfica) porque neste caso a KAOS está sendo usado com sua sintaxe padrão sem nenhuma alteração (extensão).

4.2.6 Extração e Síntese de Dados

Após a seleção e a análise da qualidade, a extração de dados foi realizada nos artigos selecionados com base em BA e Charters (2007). Nesta etapa, os dados foram extraídos de cada um dos 22 estudos incluídos nesta RSL de acordo com um formulário de extração predefinido. É possível visualizar o formulário de extração no Apêndice C. Este formulário nos permitiu registrar detalhes completos dos artigos em análise para especificar sobre como cada um deles abordaram as questões de pesquisa. A extração de dados foi auxiliada pela ferramenta StArt. Uma planilha foi usada para extrair alguns campos não suportados pelo StArt.

O objetivo da síntese dos dados era agrupar evidências dos estudos selecionados, uma vez que a agregação pode facilitar e melhorar a generalização dos resultados. A síntese

narrativa foi adotada para sintetizar e resumir os dados relacionados às questões de pesquisa. Consiste em organizar os dados de maneira consistente com as questões de pesquisa. Para melhorar a apresentação dessas descobertas, usamos ferramentas de visualização como gráficos de barras, gráficos de pizza e quadros.

4.3 Execução da RSL

A terceira etapa respalda-se na realização da revisão sistemática que se constitui na aplicação do protocolo definido.

Na fase de busca, foi definida uma *string* de busca para cada origem de seleção. Tal *string* foi testada previamente em um estudo realizado para observar se um conjunto de trabalhos previamente sugeridos foram devolvidos. Os trabalhos usados para verificar os resultados da pesquisa foram apresentados na Subseção 2.2.8, são eles: (SEMMAK *et al.*, 2008), (FAVERI *et al.*, 2016) e (LAMDDI, 2017). Alguns termos foram combinados para melhorar a *string* de busca e atenuar o problema de ter muitos artigos irrelevantes. Os termos da *string* de busca foram apresentados na Subseção 4.2.2.

Em seguida, após exportar os resultados da *string* de busca em cada base de dados selecionada nos formatos *.bib* ou *.csv*, trabalhos duplicados foram identificados com a ajuda do StArt (LAPES, 2010), essa ferramenta detecta o nível de similaridade entre os trabalhos e facilita a identificação de repetições. Quando dois trabalhos foram considerados duplicados, a escolha de quem deveria permanecer foi realizada com base no número de citações de cada um deles no Google Acadêmico.

A seleção consiste em 2 etapas:

1. Seleção (1) Título, resumo e palavras-chave;
2. Seleção (2) Leitura dos documentos completos.

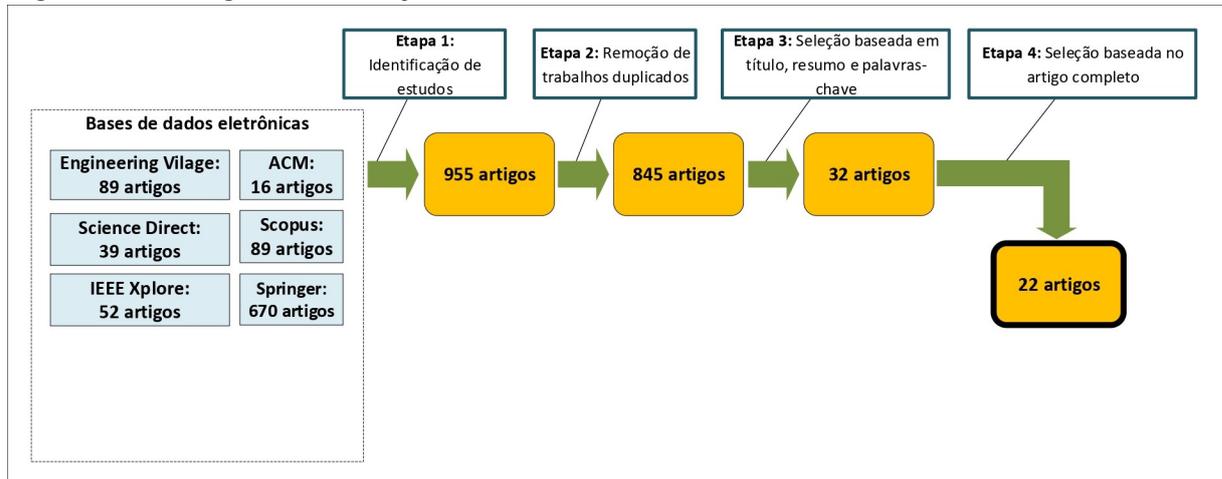
No estágio de seleção, com o auxílio da ferramenta StArt, os artigos possuem critérios de classificação. Eles são distribuídos entre artigos aceitos, artigos rejeitados, artigos duplicados e artigos não classificados. Os artigos foram previamente verificados e selecionados pela autora desta RSL; no caso de dúvidas se deveria ou não ser incluído no estudo, tais artigos foram mantidos e alocados para não classificados. Nesse caso, recebendo a verificação do orientador deste trabalho.

Finalmente, a análise de qualidade, extração e síntese de dados foram realizados.

5 RESULTADOS

Os resultados da execução da busca nas bases de dados são apresentados neste capítulo. Esses resultados podem ser vistos na Figura 8. O retângulo pontilhado no lado esquerdo apresenta as bases de dados eletrônicas e a quantidade de trabalhos identificados em cada um delas. Então resultados de identificação e seleção são apresentados. Os retângulos de borda azul identificam as etapas e os retângulos com cantos arredondados representam a quantidade de trabalhos resultantes de cada etapa. O passo 1 está relacionado com a identificação de estudos de bases de dados eletrônicas. Nessa etapa, a RSL identificou 955 trabalhos das 6 bases de dados eletrônicas selecionadas. O passo 2 está relacionado à remoção de trabalhos duplicados. Após a identificação dos artigos, 110 trabalhos duplicados foram removidos e 855 foram mantidos. Na etapa 3, as revisões são feitas com base no título, resumo e palavras-chave e os artigos não relacionados são removidos. Após essa análise foram mantidos 32 trabalhos e excluídos 813. No passo 4, analisa-se o texto completo dos artigos. Nessa etapa, foram mantidos 22 trabalhos e excluídos 10. As seleções de trabalhos nas etapas 3 e 4 foram baseadas nos critérios de inclusão e exclusão apresentados na Subseção 4.2.4. A extração dos dados foi realizada nos 22 trabalhos selecionados.

Figura 8 – Fluxograma de seleção de estudos.



Fonte: elaborada pela autora.

Os resultados da avaliação da qualidade são mostrados no Quadro 6, onde foram agrupados os trabalhos selecionados de acordo com a pontuação dos critérios de qualidade que eles obtiveram. *Pontuação CAQ* é resultante do somatório de valores numéricos referentes a possível resposta para cada critério. No Apêndice D é apresentado um quadro contendo a

resposta obtida em todas as perguntas de critérios de qualidade para cada trabalho analisado.

Quadro 6 – Resultados da avaliação da qualidade.

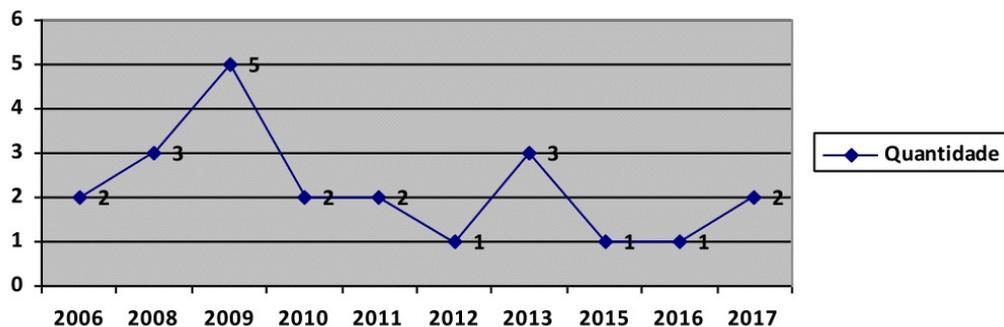
Pontuação CAQ	Qualidade	Quantidade de artigos	Lista de artigos
8	100%	1	(FAVERI <i>et al.</i> , 2016)
7,5	93,75%	4	(CHENG <i>et al.</i> , 2009), (CAILLIAU; LAMSWEERDE, 2013), (LEDRU <i>et al.</i> , 2011) e (SUTCLIFFE; SAWYER, 2013)
7	87,5%	2	(GIL; ARAÚJO, 2009) e (BRUNET <i>et al.</i> , 2008)
6,5	81,25%	4	(LAMDDI, 2017), (PONSARD; TOUZANI, 2017), (ARENAS <i>et al.</i> , 2015) e (AHMAD <i>et al.</i> , 2012)
6	75%	4	(HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (DIAS <i>et al.</i> , 2009) e (ISLAM, 2011)
5,5	68,75%	4	(DÍAZ <i>et al.</i> , 2006), (BARESI <i>et al.</i> , 2010), (BROWN <i>et al.</i> , 2006) e (HASSAN <i>et al.</i> , 2008)
5	62,5%	3	(SEMMAK <i>et al.</i> , 2009a), (SEMMAK <i>et al.</i> , 2008) e (NAGEL <i>et al.</i> , 2013)

Fonte: elaborado pela autora.

5.1 Visão Geral dos Resultados

No Apêndice E, são apresentadas informações dos 22 estudos selecionados através de uma tabela contendo a referência, título, autores, ano e o nome da conferência/periódico de cada estudo. A Figura 9 mostra a divisão de trabalhos por ano de publicação.

Figura 9 – Trabalhos selecionados divididos por ano de publicação.



Fonte: elaborada pela autora.

Através da distribuição mostrada na Figura 9 podemos observar que 2009 foi o ano com o maior número de extensões propostas, 5 no total. Além disso, é notável uma tendência de

queda nas proposições de novas extensões, não tendo sido identificadas extensões nos últimos 2 anos que antecederam essa RSL.

Foram analisados os autores com mais artigos selecionados. A lista dos principais autores (com o número de trabalhos selecionados maior que 1) é apresentado no Quadro 7. Christophe Gnaho e Régine Laleau estão nas duas primeiras posições com 5 trabalhos selecionados, seguidos por Farida Semmak, em segundo lugar com 4 trabalhos selecionados. Betty H.C. Cheng, Christophe Ponsard, João Araújo, Pete Sawyer e Vasco Amaral aparecem nas posições seguintes, com 2 trabalhos selecionados cada.

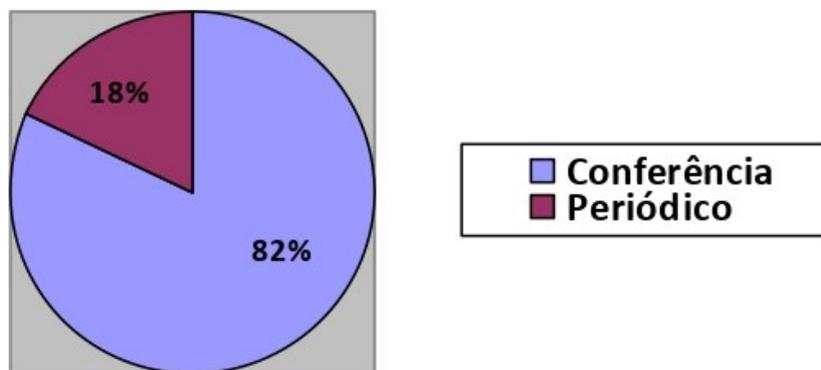
Quadro 7 – Principais autores com mais artigos selecionados.

Autor	Quantidade de artigos
Christophe Gnaho	5
Régine Laleau	5
Farida Semmak	4
Betty H.C. Cheng	2
Christophe Ponsard	2
João Araújo	2
Pete Sawyer	2
Vasco Amaral	2

Fonte: elaborado pela autora.

A Figura 10 apresenta a distribuição dos trabalhos selecionados por tipo de publicação. Foram considerados três valores nessa análise: Capítulo de Livro, Conferência (este valor inclui conferência e workshop) e Periódico. A maioria dos trabalhos são de conferências (n = 18; 82%) e trabalhos de periódico (n = 4; 18%). Não foram encontrados nenhum livro (n = 0; 0%).

Figura 10 – Trabalhos selecionados por tipo de publicação.



Fonte: elaborada pela autora.

As conferências com mais artigos identificados são apresentadas no Quadro 8.

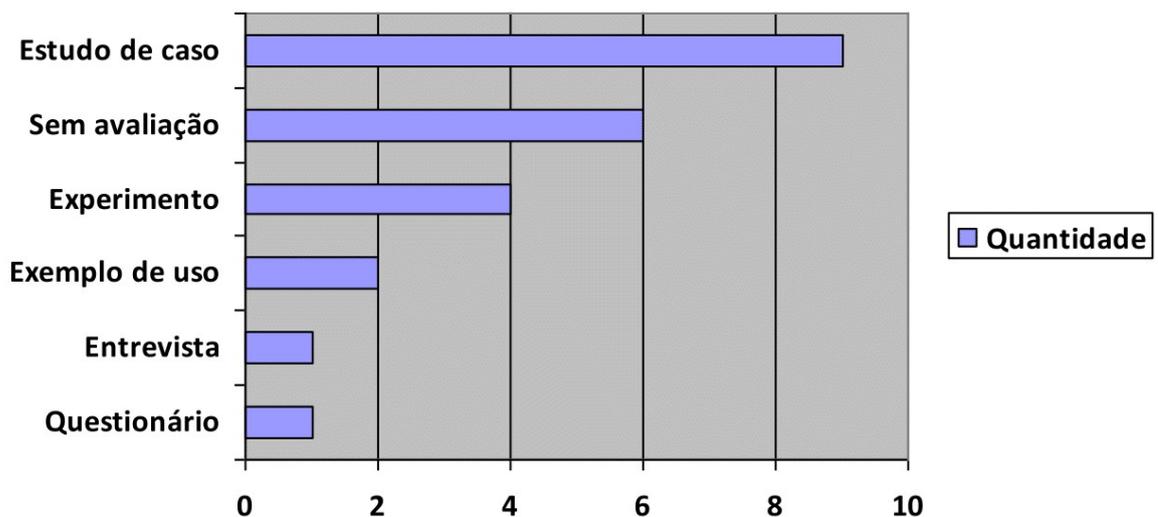
Quadro 8 – Principais conferências que mais publicaram artigos de extensão.

Conferência	Quantidade de artigos
International Conference on Advanced Information Systems Engineering	2
International Conference on Research Challenges in Information Science	2

Fonte: elaborado pela autora.

Analisamos como a validação da proposta de extensão foi realizada. Em cada publicação classificamos o formulário de validação de acordo com a classificação apresentada no artigo. Os trabalhos usaram as seguintes categorias: Estudo de caso, Exemplo de uso, Experimento, Questionário e Entrevista. A Figura 11 apresenta os resultados dessa análise. Dos 22 trabalhos analisados, 1 apresentou dois tipos de validação: estudo de caso e entrevista. Logo, no gráfico apresentado a soma da quantidade de estudos é igual a 23. Estudo de caso é a validação mais usada (usada por 9/22 trabalhos), o experimento é o segundo formulário mais utilizado para validar os trabalhos com 4/22 artigos. Exemplo de uso é utilizado por 2/22 artigos para validar os resultados. Questionário e Entrevista são utilizados por 1/22 trabalhos cada. Os trabalhos que não apresentaram nenhum tipo de validação foram classificados como sem avaliação (6/22).

Figura 11 – Trabalhos selecionados por tipo de validação.



Fonte: elaborada pela autora.

A seguir, serão descritos os resultados obtidos para cada questão de pesquisa apresentada na subseção 4.2.1.

5.2 QP1: Quais as áreas de aplicação das extensões?

Esta questão de pesquisa tem como objetivo classificar as extensões e agrupá-las para tornar a pesquisa de extensões mais objetiva.

Os 22 estudos analisados também foram classificados em 11 áreas de aplicação diferentes. Nós consideramos as áreas de aplicação apresentadas no artigo (YU, 2009) e extraímos algumas áreas de aplicação de acordo com a descrição dada pelos trabalhos selecionados. O Quadro 9 traz a distribuição de extensões selecionadas por área de aplicação a qual pertencem.

Quadro 9 – Extensões distribuídas por área de aplicação.

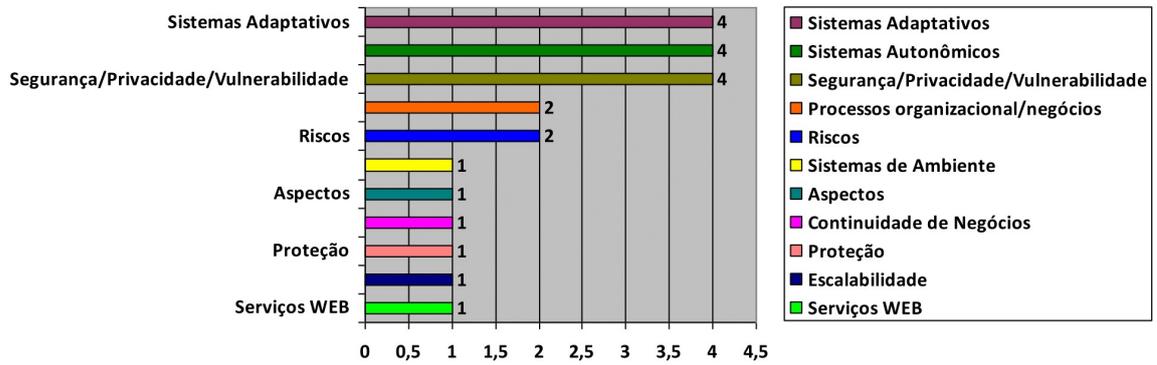
Área de aplicação	Lista de artigos
Aspectos	(GIL; ARAÚJO, 2009)
Continuidade de Negócios	(ARENAS <i>et al.</i> , 2015)
Escalabilidade	(DIAS <i>et al.</i> , 2009)
Processos organizacional/negócio	(PONSARD; TOUZANI, 2017) e (NAGEL <i>et al.</i> , 2013)
Proteção	(LAMDDI, 2017)
Riscos	(CAILLIAU; LAMSWEERDE, 2013) e (ISLAM, 2011)
Segurança/privacidade/vulnerabilidade	(LEDRU <i>et al.</i> , 2011), (FAVERI <i>et al.</i> , 2016), (HASSAN <i>et al.</i> , 2008) e (HUANG <i>et al.</i> , 2009)
Serviços WEB	(DÍAZ <i>et al.</i> , 2006)
Sistemas adaptativos	(CHENG <i>et al.</i> , 2009), (BROWN <i>et al.</i> , 2006), (BARESI <i>et al.</i> , 2010) e (SUTCLIFFE; SAWYER, 2013)
Sistemas autônômicos	(SEMMAK <i>et al.</i> , 2009a), (SEMMAK <i>et al.</i> , 2008), (SEMMAK <i>et al.</i> , 2009b) e (BRUNET <i>et al.</i> , 2008)
Sistemas de ambiente	(AHMAD <i>et al.</i> , 2012)

Fonte: elaborado pela autora.

Sistemas adaptativos, sistemas autônômicos e segurança/privacidade/vulnerabilidade são as áreas de aplicação mais descritas com 4 trabalhos cada (aparecem em 12/22 trabalhos); processos organizacional/negócios e riscos aparecem em segundo lugar com 2 trabalhos cada (aparecem em 4/22 trabalhos); Sistemas de ambiente, aspectos, continuidade de negócios, proteção, escalabilidade e serviços WEB são áreas de aplicação que vêm em seguida sendo descritos por 1 artigo cada (aparecem em 6/22 trabalhos). É possível visualizar o resultado dessa extração na Figura 12.

Em seguida, na Figura 13, as áreas de aplicação foram agrupadas por área / similaridade. Por exemplo, as áreas de aplicação de sistemas adaptativos, sistemas autônômicos e

Figura 12 – Trabalhos selecionados por área de aplicação.

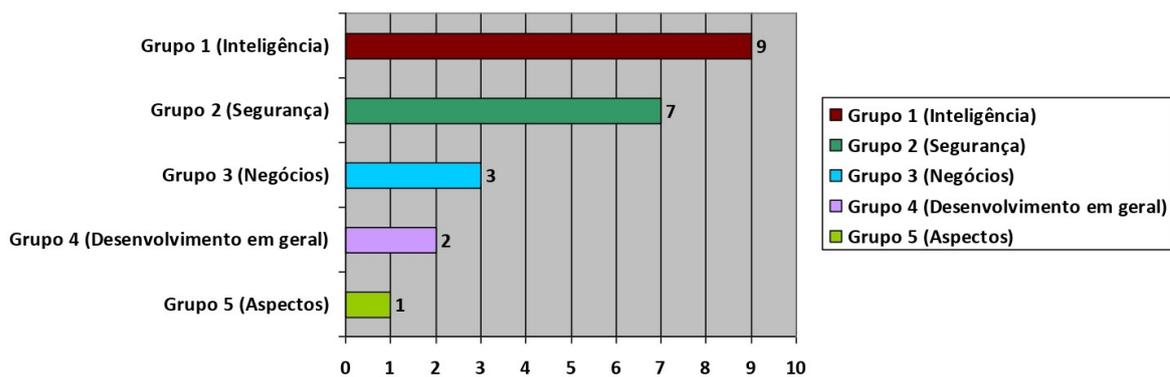


Fonte: elaborada pela autora.

systemas de ambiente estão relacionadas à inteligência artificial; portanto, foi criado um grupo chamado Inteligência para agrupá-los.

1. **Grupo 1 (Inteligência):** Sistemas adaptativos, Sistemas autonômico e Sistemas de ambiente;
2. **Grupo 2 (Segurança):** Risco, Proteção e Segurança/Privacidade/Vulnerabilidade;
3. **Grupo 3 (Negócios):** Processos negócio/organizacional e Continuidade de negócios;
4. **Grupo 4 (Desenvolvimento em geral):** Escalabilidade e Serviços WEB;
5. **Grupo 5 (Aspectos):** Aspectos.

Figura 13 – Trabalhos selecionados por agrupamento das áreas de aplicação.



Fonte: elaborada pela autora.

5.3 QP2: Para as extensões selecionadas, quais são derivadas de outras extensões?

Esta pergunta busca verificar se as extensões encontradas são derivadas de outras extensões. Para responder essa questão de pesquisa foi adicionado um campo no formulário de extração chamado *possui extensão base*, com valores de resposta *sim* ou *não*. O primeiro valor foi atribuído a trabalhos que possuem uma extensão base, ou seja, são estendidos a partir de uma extensão já existente da linguagem KAOS. Enquanto que o segundo valor refere-se aos estudos que não são estendidos de outras extensões, mas sim diretamente dos modelos da KAOS.

A extensão proposta por Díaz *et al.* (2006) se baseia na extensão proposta em Ponsard e Touzani (2017). E a extensão proposta por Ahmad *et al.* (2012) se fundamenta em Cheng *et al.* (2009). As duas extensões base identificadas foram analisadas nesta RSL.

O Quadro 10 apresenta a distribuição dos artigos por extensão base.

Quadro 10 – Distribuição de artigos por extensão base.

Extensão base	Lista de artigos
Sim	(DÍAZ <i>et al.</i> , 2006) e (AHMAD <i>et al.</i> , 2012)
Não	(CHENG <i>et al.</i> , 2009), (GIL; ARAÚJO, 2009), (CAILLIAU; LAMSWEERDE, 2013), (SEMMAK <i>et al.</i> , 2009a), (LAMDDI, 2017), (SEMMAK <i>et al.</i> , 2008), (PONSARD; TOUZANI, 2017), (LEDROU <i>et al.</i> , 2011), (BARESI <i>et al.</i> , 2010), (FAVERI <i>et al.</i> , 2016), (ARENAS <i>et al.</i> , 2015), (BROWN <i>et al.</i> , 2006), (HASSAN <i>et al.</i> , 2008), (NAGEL <i>et al.</i> , 2013), (SUTCLIFFE; SAWYER, 2013), (HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (DIAS <i>et al.</i> , 2009), (ISLAM, 2011) e (BRUNET <i>et al.</i> , 2008)

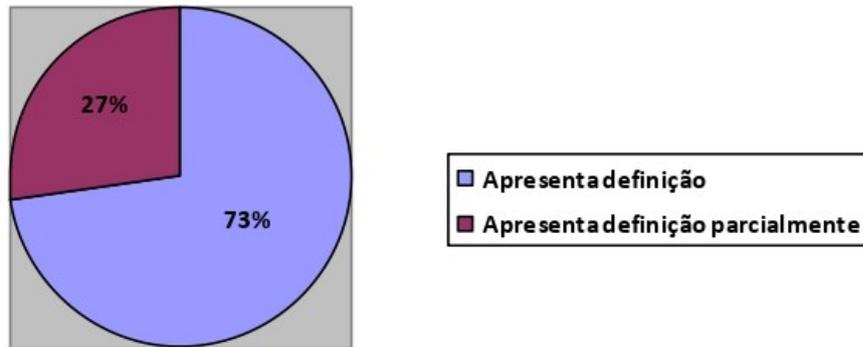
Fonte: elaborado pela autora.

5.4 QP3: Os trabalhos apresentam uma definição de conceitos envolvidos nas extensões?

Quando uma extensão é proposta, é importante definir os construtores da linguagem (ou novos conceitos introduzidos) (FRANCE; RUMPE, 2007). Portanto, essa pergunta se refere à presença de uma definição dos conceitos introduzidos pelas extensões. Usamos três valores possíveis: apresenta definição, não apresenta definição e apresenta definição parcialmente. O primeiro valor foi atribuído a trabalhos que definem todos os conceitos introduzidos pelas extensões, o segundo valor foi definido para trabalhos que não definem os conceitos introduzidos pelas extensões e a terceira quando o documento define pelo menos um mas não todos os conceitos.

Os resultados apresentados na Figura 14 apontam que 16/22 (73%) artigos apresentaram definição, 6/22 (27%) apresentaram definição parcialmente e nenhum artigo não apresentou definição.

Figura 14 – Trabalhos selecionados por definição de conceitos.



Fonte: elaborada pela autora.

O Quadro 11 mostra a distribuição de artigos por definição de conceitos.

Quadro 11 – Distribuição de artigos por definição de conceitos.

Definição de Conceitos	Lista de artigos
Apresenta definição	(CHENG <i>et al.</i> , 2009), (DÍAZ <i>et al.</i> , 2006), (GIL; ARAÚJO, 2009), (CAILLIAU; LAMSWEERDE, 2013), (SEMMAK <i>et al.</i> , 2009a), (SEMMAK <i>et al.</i> , 2008), (PONSARD; TOUZANI, 2017), (ARENAS <i>et al.</i> , 2015), (BROWN <i>et al.</i> , 2006), (HASSAN <i>et al.</i> , 2008), (SUTCLIFFE; SAWYER, 2013), (HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (DIAS <i>et al.</i> , 2009), (AHMAD <i>et al.</i> , 2012) e (BRUNET <i>et al.</i> , 2008)
Apresenta definição parcialmente	(LAMDDI, 2017), (LEDRU <i>et al.</i> , 2011), (BARESI <i>et al.</i> , 2010), (FAVERI <i>et al.</i> , 2016), (NAGEL <i>et al.</i> , 2013) e (ISLAM, 2011)

Fonte: elaborado pela autora.

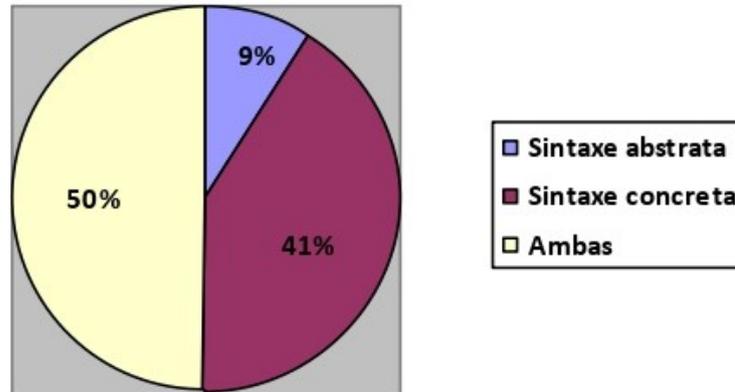
5.5 QP4: Quais os níveis de extensão da KAOS (somente sintaxe concreta, somente sintaxe abstrata ou ambas as sintaxes)?

Os níveis de uma representação da linguagem de modelagem foram mostrados na Seção 2.1, ou seja, sintaxes abstrata e concreta. Naturalmente, quando uma linguagem de modelagem é estendida, é importante aplicá-la nas duas sintaxes, pois são complementares e devem manter a consistência entre si.

Esta questão de pesquisa analisou em que nível a extensão foi proposta: sintaxe abstrata, sintaxe concreta ou ambas. A representação gráfica na Figura 15 indica que metade

dos artigos 11/22 (50%) estenderam a linguagem em ambos os níveis; 9/22 (41%) estenderam apenas na sintaxe concreta e 2/22 (9%) propuseram a extensão somente na sintaxe abstrata.

Figura 15 – Representação dos níveis de sintaxe das extensões.



Fonte: elaborada pela autora.

O Quadro 12 mostra a distribuição de artigos por nível de sintaxe proposta para a extensão.

Quadro 12 – Distribuição de artigos por nível de sintaxe da extensão.

Nível de sintaxe	Lista de artigos
Sintaxe abstrata	(GIL; ARAÚJO, 2009) e (LAMDDI, 2017)
Sintaxe concreta	(DÍAZ <i>et al.</i> , 2006), (CAILLIAU; LAMSWEERDE, 2013), (SEMMAK <i>et al.</i> , 2009a), (SEMMAK <i>et al.</i> , 2008), (PONSARD; TOUZANI, 2017), (ARENAS <i>et al.</i> , 2015), (BROWN <i>et al.</i> , 2006), (SUTCLIFFE; SAWYER, 2013) e (DIAS <i>et al.</i> , 2009)
Ambas (Sintaxe concreta e Sintaxe abstrata)	(CHENG <i>et al.</i> , 2009), (LEDRU <i>et al.</i> , 2011), (BARESI <i>et al.</i> , 2010), (FAVERI <i>et al.</i> , 2016), (HASSAN <i>et al.</i> , 2008), (NAGEL <i>et al.</i> , 2013), (HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (ISLAM, 2011), (AHMAD <i>et al.</i> , 2012) e (BRUNET <i>et al.</i> , 2008)

Fonte: elaborado pela autora.

5.6 QP5: Como a extensão da sintaxe abstrata foi proposta (conservativa ou não conservativa)?

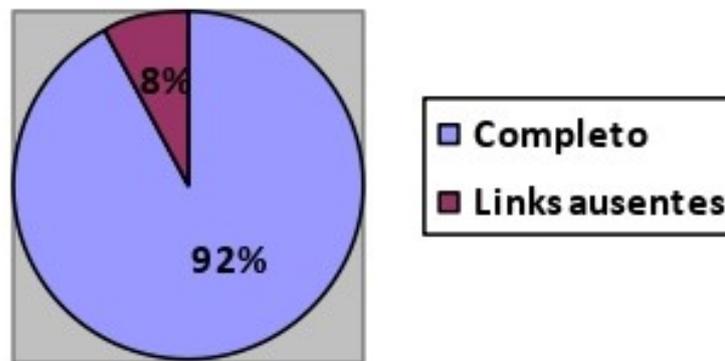
Esta questão de pesquisa pretende analisar a sintaxe abstrata das extensões. A sintaxe abstrata de uma linguagem de modelagem é composta por seu metamodelo e regras de boa formação. Para responder a essa pergunta foi realizada a análise de abrangência da representação dos conceitos da KAOS. Analisamos apenas os artigos que apresentaram representação de sintaxe abstrata, ou seja, 11 artigos que estendiam ambas as sintaxes e 2 artigos que estenderam apenas

a sintaxe abstrata, totalizando 13 artigos analisados.

Algumas obras definiram um metamodelo para KAOS, mas não existe um metamodelo padrão. Por esse motivo, a análise de completude focou na identificação da presença de conceitos da KAOS nos metamodelos.

A Figura 16 apresenta os resultados desta análise: 12/13 (92%) trabalhos criaram um metamodelo com todos os conceitos de definição da KAOS. 1/13 (8%) trabalhos estão incompletos. O trabalho, cujo modelo está incompleto não apresenta links.

Figura 16 – Completude do metamodelo.



Fonte: elaborada pela autora.

O Quadro 13 mostra a distribuição de artigos por completude de metamodelo.

Quadro 13 – Distribuição de artigos por completude de metamodelo.

Completude do metamodelo	Lista de artigos
Completo	(CHENG <i>et al.</i> , 2009), (GIL; ARAÚJO, 2009), (LEDRU <i>et al.</i> , 2011), (BARESI <i>et al.</i> , 2010), (FAVERI <i>et al.</i> , 2016), (HASSAN <i>et al.</i> , 2008), (NAGEL <i>et al.</i> , 2013), (HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (ISLAM, 2011), (AHMAD <i>et al.</i> , 2012) e (BRUNET <i>et al.</i> , 2008)
Nós ausentes	-
Links ausentes	(LAMDDI, 2017)
Nós e links ausentes	-

Fonte: elaborado pela autora.

As possíveis classificações em relação à sintaxe original para uma extensão foram apresentadas na Seção 2.1. Uma extensão pode ser classificada como conservativa ou não conservativa. Para responder essa questão de pesquisa foi adicionado um campo no formulário de extração chamado *tipo de extensão* para identificar o impacto da extensão na sintaxe original, com valores de resposta *conservativa* ou *não conservativa*. O primeiro valor foi atribuído a extensões

que, quando estendidas, mantém a sintaxe original da linguagem, sem alterações. Enquanto que o segundo valor refere-se às extensões que, quando estendidas, alteram ou reduzem a sintaxe original da linguagem.

O Quadro 14 apresenta a distribuição das extensões da sintaxe abstrata por classificação, de acordo com seu impacto na sintaxe original da linguagem KAOS.

Quadro 14 – Distribuição de extensões por classificação.

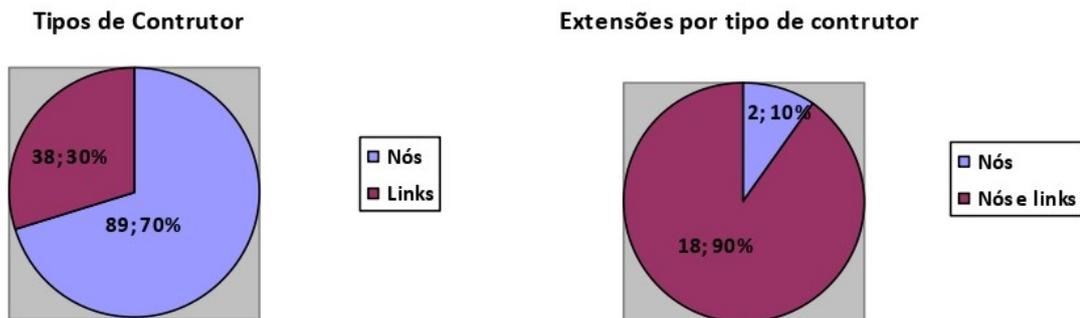
Tipo de extensão	Lista de artigos
Conservativa	(CHENG <i>et al.</i> , 2009), (GIL; ARAÚJO, 2009), (LEDRU <i>et al.</i> , 2011), (BARESI <i>et al.</i> , 2010), (FAVERI <i>et al.</i> , 2016), (HASSAN <i>et al.</i> , 2008), (NAGEL <i>et al.</i> , 2013), (HUANG <i>et al.</i> , 2009), (SEMMAK <i>et al.</i> , 2009b), (ISLAM, 2011), (AHMAD <i>et al.</i> , 2012) e (BRUNET <i>et al.</i> , 2008)
Não conservativa	(LAMDDI, 2017)

Fonte: elaborado pela autora.

5.7 QP6: Como a extensão da sintaxe concreta foi proposta?

Analisamos os artigos que estendem a KAOS na sintaxe concreta ou nas duas sintaxes (artigos que estendem apenas a sintaxe abstrata não foram considerados nessa avaliação), totalizando 20 artigos analisados. Esta análise tem o objetivo de identificar o tipo de construtores envolvidos nas extensões. Inicialmente, verificamos se o construtor da extensão foi aplicado a nós, links ou ambos. Foram identificadas 127 novos construtores. Concluímos que 70% (89 construtores) dos novos construtores são aplicados aos nós e 30% (38 construtores) são sobre links. Além disso, identificamos que 10% (2/20) das extensões envolvem a definição de apenas novos nós e 90% (18/20) das extensões envolvem a definição de novos nós e links. A Figura 17 mostra essa informação.

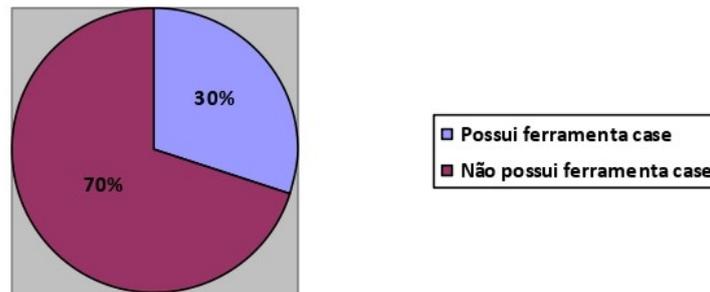
Figura 17 – Distribuição do tipo de construtores introduzidos pelas extensões identificadas.



Fonte: elaborada pela autora.

Além disso, analisamos se a nova representação introduzida na sintaxe concreta era suportada por uma ferramenta case. Apenas identificamos os trabalhos que chegaram ao estágio de implementação de uma ferramenta. Os resultados apontam 6/20 (30%) trabalhos selecionados possuem suporte de ferramentas e 14/20 (70%) não são suportados por uma ferramenta Case. A Figura 18 mostra o resultado dessa análise.

Figura 18 – Suporte de ferramenta case.



Fonte: elaborada pela autora.

O Quadro 15 lista para cada extensão a quantidade de construtores identificados e se possui suporte de uma ferramenta case ou não.

Quadro 15 – Quantidade de construtores identificados e ferramenta case.

Lista de Artigos	Quantidade de construtores	Ferramenta case
(CHENG <i>et al.</i> , 2009)	10	Não possui ferramenta case
(DÍAZ <i>et al.</i> , 2006)	7	Possui ferramenta case
(CAILLIAU; LAMSWEERDE, 2013)	6	Não possui ferramenta case
(SEMMAK <i>et al.</i> , 2009a)	4	Não possui ferramenta case
(SEMMAK <i>et al.</i> , 2008)	4	Não possui ferramenta case
(PONSARD; TOUZANI, 2017)	2	Não possui ferramenta case
(LEDRU <i>et al.</i> , 2011)	9	Possui ferramenta case
(BARESI <i>et al.</i> , 2010)	6	Não possui ferramenta case
(FAVERI <i>et al.</i> , 2016)	16	Não possui ferramenta case
(ARENAS <i>et al.</i> , 2015)	1	Não possui ferramenta case
(BROWN <i>et al.</i> , 2006)	5	Não possui ferramenta case
(HASSAN <i>et al.</i> , 2008)	4	Não possui ferramenta case
(NAGEL <i>et al.</i> , 2013)	4	Possui ferramenta case
(SUTCLIFFE; SAWYER, 2013)	6	Não possui ferramenta case
(HUANG <i>et al.</i> , 2009)	7	Não possui ferramenta case
(SEMMAK <i>et al.</i> , 2009b)	5	Possui ferramenta case
(DIAS <i>et al.</i> , 2009)	13	Possui ferramenta case
(ISLAM, 2011)	9	Não possui ferramenta case
(AHMAD <i>et al.</i> , 2012)	5	Possui ferramenta case
(BRUNET <i>et al.</i> , 2008)	4	Não possui ferramenta case

Fonte: elaborado pela autora.

5.8 QP7: Foram definidos mecanismos de extensão para KAOS?

Nos artigos selecionados, não encontramos uma descrição dos mecanismos de extensão padrão da KAOS. Todos os trabalhos selecionados apresentam uma extensão para KAOS e introduziram novos construtores no metamodelo. Esses documentos selecionados usam um conjunto de representações de metamodelo, como novas representações gráficas, novas propriedades, novos compartimentos e representações textuais para aumentar a sintaxe textual da KAOS. No entanto, não foi baseado em mecanismos de extensão.

5.9 Análises e Discussão

Apresentamos uma visão geral das extensões da KAOS. O estudo destacou principais autores, conferências e periódicos, principais áreas de aplicação envolvidas nas extensões e tipo de validação. Além disso, respondemos 7 questões de pesquisa relacionadas a linguagem KAOS.

Descrevemos a seguir um conjunto de questões em aberto relacionadas as extensões da KAOS e que podem ser utilizadas como direcionamentos para trabalhos futuros.

Na Seção 5.2 listamos 11 áreas de aplicação para as quais as extensões aqui identificadas foram propostas. Não foram identificadas extensões nesta RSL para as áreas de aplicação como agentes, conflitos, sistemas colaborativos, contexto, banco de dados, avaliação, sistemas robóticos, conformidade, confiabilidade, requisitos mensuráveis e normas. Assim, a criação de novas extensões de KAOS para estas áreas de aplicação seriam importantes para apoiar a modelagem de requisitos destas de forma adequada.

Análise de como as extensões da KAOS vem sendo propostas e o que pode ser feito para melhorá-las. A análise desta RSL ajuda a se ter uma melhor compreensão sobre as extensões que foram identificadas. Uma análise mais profunda pode contribuir de forma a entender como se propõe essas extensões e o que pode ser feito para melhorá-las. Esta questão em aberto pode ser explorada com um estudo qualitativo com os autores de extensões da KAOS assim como foi apresentado por Goncalves *et al.* (2018c) para iStar.

Criação de um processo para conduzir as próximas extensões para KAOS assim como um processo foi proposto para conduzir as extensões do iStar em Goncalves *et al.* (2020). Como não há uma forma padrão para estender a KAOS e as extensões para KAOS têm sido propostas para um formato de fim específico, podem ocorrer inconsistências e conflitos entre os novos construtores criados. Para isso, é importante se criar processos que orientem os extensores

na criação de extensões da linguagem de modelagem KAOS;

Criação de mecanismos de extensão para KAOS. A grande variedade de formas para estender a KAOS talvez seja causada pela falta de uma maneira padronizada de estender a linguagem. Logo, se torna importante criar mecanismos de extensão padrão. Uma vez que estes mecanismos sejam parte da linguagem, eles podem ser incluídos em ferramentas case e facilitar a modelagem de extensões para KAOS assim como acontece com outras linguagens como UML e iStar (Goncalves *et al.* (2018)).

Criação de um catálogo de extensões da KAOS assim como foi proposto o *A Catalogue of iStar Extensions* (CATIE) por Goncalves *et al.* (2018b). Ao se criar um catálogo se torna mais fácil ter acesso as informações das extensões. Um catálogo vem sendo criado para manter as extensões da KAOS identificadas nesta RSL. Ele foi projetado para ser atualizado com novas extensões. O repositório está disponível em: <http://kaosextensions.quixada.ufc.br/>.

6 CONCLUSÕES E TRABALHOS FUTUROS

KAOS é uma linguagem de modelagem de requisitos orientada por objetivos que vem sendo amplamente utilizada e estendida desde sua criação em meados dos anos 90. Neste trabalho, apresentamos os resultados de uma Revisão Sistemática da Literatura para identificar e analisar as extensões da KAOS. Esta informação pode ser útil aos analistas de requisitos que queiram utilizar uma extensão existente ou criar uma nova extensão para KAOS e também aos usuários que queiram utilizar as extensões da KAOS. Nosso objetivo com este trabalho é melhorar o entendimento de como a KAOS vem sendo estendida através de evidências na literatura. Buscas automáticas foram utilizadas. Nós identificamos 22 artigos que estendem a KAOS, essas abordagens foram agrupadas em categorias e 8 questões de pesquisa foram respondidas com base em evidências encontradas nos trabalhos selecionados.

Inicialmente, apresentamos a evolução de criação de extensões para KAOS por ano. Também analisamos os autores que propuseram mais extensões, bem como as conferências/periódicos. Exibimos as formas de validação usadas em cada estudo selecionado. Identificamos a área de aplicação das extensões (agrupando-as em 5 categorias). Investigamos se as extensões descobertas estendiam de outras extensões e como a apresentação dos conceitos envolvido nas extensões foi feito. Também esmiuçamos o nível de sintaxe abordado pelas extensões, a classificação das extensões, como a sintaxe abstrata é definida, definição de construtores (incluindo análise sobre definição de ferramenta), se há mecanismos de extensão para estender a KAOS e questões em aberto relacionadas à linguagem.

As descobertas mais importantes são as seguintes: Não há mecanismos de extensão padrão para KAOS, como em UML, por exemplo. Além disto, as extensões vêm sendo feitas de forma ad-hoc. Portanto, acreditamos que o primeiro passo para sistematizar as extensões da KAOS é criar um processo para orientar essa atividade. Necessidade de suporte à ferramenta case. As ferramentas de modelagem são importantes para permitir o uso de extensões da KAOS pela academia e pelo setor de profissionais. No entanto, os resultados apontam que 70% das extensões não possuem uma ferramenta de modelagem para suportar a modelagem da KAOS com os novos construtores introduzidos pela extensão. Acreditamos que as extensões da KAOS sem uma ferramenta para dar suporte podem desencorajar o engenheiro de requisitos a usar a proposta.

Os resultados desta RSL podem contribuir para uma melhor compreensão das extensões já feitas e para criar uma base consistente para pesquisas adicionais abordando a extensibili-

dade da KAOS. Ademais, a metodologia de pesquisa utilizada nesta RSL pode servir como base para futuras revisões.

Como trabalho futuro, será realizado um processo de snowballing com a intenção de encontrar qualquer trabalho que não foi considerado durante esta RSL. Tal processo se dará com base em trabalhos já avaliados como relevantes. O snowballing será realizado com os autores dos trabalhos selecionados. Além disso, está sendo desenvolvido um catálogo baseado nos resultados desta RSL que está hospedado na Universidade Federal do Ceará - Campus Quixadá. Nesse catálogo, é possível listar as extensões indexadas, listar seus construtores e pesquisar por autor, título e área de aplicação. Para ter acesso ao repositório do catálogo disponível se faz necessário acessar o seguinte link: <http://kaosextensions.quixada.ufc.br/>

REFERÊNCIAS

- AHMAD, M.; BRUEL, J.-M.; LALEAU, R.; GNAHO, C. Using relax, sysml and kaos for ambient systems requirements modeling. **Procedia Computer Science**, Elsevier, v. 10, p. 474–481, 2012.
- ARENAS, A. E.; MASSONET, P.; PONSARD, C.; AZIZ, B. Goal-oriented requirement engineering support for business continuity planning. In: JEUSFELD, M. A.; KARLAPALEM, K. (Ed.). **Advances in Conceptual Modeling**. Cham: Springer International Publishing, 2015. p. 259–269.
- BA, K.; CHARTERS, S. **Guidelines for performing systematic literature reviews in software engineering**. v. 2, 01 2007.
- BARESI, L.; PASQUALE, L.; SPOLETINI, P. Fuzzy goals for requirements-driven adaptation. In: IEEE. **2010 18th IEEE International Requirements Engineering Conference**. [S.l.], 2010. p. 125–134.
- BRAMBILLA, M.; CABOT, J.; WIMMER, M. Model-driven software engineering in practice. **Synthesis Lectures on Software Engineering**, Morgan & Claypool Publishers, v. 1, n. 1, p. 1–182, 2012.
- BROWN, G.; CHENG, B. H.; GOLDSBY, H.; ZHANG, J. Goal-oriented specification of adaptation requirements engineering in adaptive systems. In: ACM. **Proceedings of the 2006 international workshop on Self-adaptation and self-managing systems**. [S.l.], 2006. p. 23–29.
- BRUNET, J.; SEMMAK, F.; LALEAU, R.; GNAHO, C. Using variants in kaos goal modelling. In: **ICEIS (3-2)**. [S.l.: s.n.], 2008. p. 339–344.
- CAILLIAU, A.; LAMSWEERDE, A. van. Assessing requirements-related risks through probabilistic goals and obstacles. **Requirements Engineering**, Springer, v. 18, n. 2, p. 129–146, 2013.
- CHENG, B. H.; SAWYER, P.; BENCOMO, N.; WHITTLE, J. A goal-based modeling approach to develop requirements of an adaptive system with environmental uncertainty. In: SPRINGER. **International Conference on Model Driven Engineering Languages and Systems**. [S.l.], 2009. p. 468–483.
- DARDENNE, A.; LAMSWEERDE, A. V.; FICKAS, S. Goal-directed requirements acquisition. **Science of computer programming**, Elsevier, v. 20, n. 1-2, p. 3–50, 1993.
- DERMEVAL, D.; VILELA, J.; BITTENCOURT, I. I.; CASTRO, J.; ISOTANI, S.; BRITO, P.; SILVA, A. Applications of ontologies in requirements engineering: a systematic review of the literature. **Requirements Engineering**, Springer, v. 21, n. 4, p. 405–437, 2016.
- DIAS, A.; AMARAL, V.; ARAUJO, J. Towards a domain specific language for a goal-oriented approach based on kaos. In: IEEE. **2009 Third International Conference on Research Challenges in Information Science**. [S.l.], 2009. p. 409–420.
- DÍAZ, G.; CAMBRONERO, M.-E.; TOBARRA, M. L.; VALERO, V.; CUARTERO, F. Analysis and verification of time requirements applied to the web services composition. In: SPRINGER. **International Workshop on Web Services and Formal Methods**. [S.l.], 2006. p. 178–192.

- DING, W.; LIANG, P.; TANG, A.; VLIET, H. V. Knowledge-based approaches in software documentation: a systematic literature review. **Information and Software Technology**, Elsevier, v. 56, n. 6, p. 545–567, 2014.
- DYBÅ, T.; DINGSØYR, T. Empirical studies of agile software development: a systematic review. **Information and software technology**, Elsevier, v. 50, n. 9-10, p. 833–859, 2008.
- FATIMA, M.; FATIMA, M. Kaos: A goal oriented requirement engineering approach. **International Journal for Innovative Research in Science & Technology**, v. 1, n. 10, p. 133–135, 2015.
- FAVERI, C. D.; MOREIRA, A.; AMARAL, V. Goal-driven deception tactics design. In: IEEE. **2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)**. [S.l.], 2016. p. 264–275.
- FRANCE, R.; RUMPE, B. Model-driven development of complex software: a research roadmap. In: IEEE COMPUTER SOCIETY. **2007 Future of Software Engineering**. [S.l.], 2007. p. 37–54.
- GIL, A.; ARAÚJO, J. Aspectkaos: integrating early-aspects into kaos. In: ACM ELECTRONIC LIBRARY. **Early Aspects**. [S.l.]: ACM Electronic Library, 2009.
- GONCALVES, E.; ARAUJO, J.; CASTRO, J. Towards extension mechanisms in istar 2.0. **International i* Workshop co-located with the 30th International Conference on Advanced Information Systems Engineering (CAiSE 2018)**, 2018.
- GONCALVES, E.; ARAUJO, J.; CASTRO, J. A process to support the creation of istar extensions. **35th ACM/SIGAPP Symposium On Applied Computing - Requirements Engineering Track**, 2020.
- GONCALVES, E.; CASTRO, J.; ARAUJO, J.; HEINECK, T. A systematic literature review of istar extensions. **Journal of Systems and Software**, Elsevier, v. 137, p. 1–33, 2018.
- GONCALVES, E.; HEINECK, T.; CASTRO, J. Catie: a catalogue of istar extensions. **CADERNOS DO IME. SÉRIE INFORMÁTICA**, v. 48, p. 23–37, 2018.
- GONCALVES, E.; OLIVEIRA, M. D.; MONTEIRO, I.; CASTRO, J.; ARAÚJO, J. Understanding what is important in istar extension proposals: the viewpoint of researchers. **Requirements Engineering**, 07 2018.
- HASSAN, R.; BOHNER, S.; EL-KASSAS, S.; ELTOWEISSY, M. Goal-oriented, b-based formal derivation of security design specifications from security requirements. In: IEEE. **2008 Third International Conference on Availability, Reliability and Security**. [S.l.], 2008. p. 1443–1450.
- HERNANDES, E.; ZAMBONI, A.; FABBRI, S.; THOMMAZO, A. D. Using gqm and tam to evaluate start-a tool that supports systematic review. **CLEI Electronic Journal**, Centro Latinoamericano de Estudios en Informática, v. 15, n. 1, p. 3–3, 2012.
- HORKOFF, J.; AYDEMIR, F. B.; CARDOSO, E.; LI, T.; MATÉ, A.; PAJA, E.; SALNITRI, M.; MYLOPOULOS, J.; GIORGINI, P. Goal-oriented requirements engineering: a systematic literature map. In: IEEE. **2016 IEEE 24th International Requirements Engineering Conference (RE)**. [S.l.], 2016. p. 106–115.

HORKOFF, J.; LI, T.; LI, F.-L.; SALNITRI, M.; CARDOSO, E.; GIORGINI, P.; MYLOPOULOS, J. Using goal models downstream: a systematic roadmap and literature review. **International Journal of Information System Modeling and Design (IJISMD)**, IGI Global, v. 6, n. 2, p. 1–42, 2015.

HUANG, C.; SUN, J.; WANG, X.; SI, Y. Role engineering with skaos for systems employing rbac. **Networking and Digital Society, International Conference on**, v. 2, p. 56–60, 05 2009.

ISLAM, S. Towards a framework for offshore outsource software development risk management model. **Journal of Software**, v. 6, n. 1, p. 38–47, 2011.

KELLY, S.; TOLVANEN, J.-P. **Domain-specific modeling: enabling full code generation**. [S.l.]: John Wiley & Sons, 2008.

KITCHENHAM, B.; BRERETON, P. A systematic review of systematic review process research in software engineering. **Information and software technology**, Elsevier, v. 55, n. 12, p. 2049–2075, 2013.

LAMDDI, M. A. Developing dependability requirements engineering for secure and safe information systems with knowledge acquisition for automated specification. **Journal of Software Engineering and Applications**, Scientific Research Publishing, v. 10, n. 02, p. 211, 2017.

LAPES. **Start-state of the art through systematic review tool**. 2010. Disponível em: <http://lapes.dc.ufscar.br/>, Acesso em: 15 abr. 2019.

LEDRU, Y.; RICHIER, J.-L.; IDANI, A.; LABIADH, M.-A. From kaos to rbac: a case study in designing access control rules from a requirements analysis. **2011 Conference on Network and Information Systems Security, SAR-SSI 2011, Proceedings**, 05 2011.

MATULEVIČIUS, R.; HEYMANS, P. Visually effective goal models using kaos. In: SPRINGER. **International Conference on Conceptual Modeling**. [S.l.], 2007. p. 265–275.

MILES, R.; HAMILTON, K. **Learning UML 2.0**. [S.l.: s.n.], 2006. ISBN 0596009828.

MUSSBACHER, G.; AMYOT, D.; BREU, R.; BRUEL, J.-M.; CHENG, B.; COLLET, P.; COMBEMALE, B.; FRANCE, R.; HELDAL, R.; HILL, J.; KIENZLE, J.; SCHÖTTLE, M.; STEIMANN, F.; STIKKOLORUM, D.; WHITTLE, J. The relevance of nodel-driven engineering thirty years from now. In: **MoDELS**. [S.l.: s.n.], 2014.

NAGEL, B.; GERTH, C.; POST, J.; ENGELS, G. Kaos4soa - extending kaos models with temporal and logical dependencies. In: **CAiSE Forum**. [S.l.: s.n.], 2013.

PONSARD, C.; TOUZANI, M. Extending land administration domain models with a goal perspective. In: **GISTAM**. [S.l.: s.n.], 2017.

RESPECT - IT. **A KAOS tutorial**: Objectiver. Parc de l'Alliance, France, 2007.

SEMMAK, F.; GNAHO, C.; LALEAU, R. Extended kaos to support variability for goal oriented requirements reuse. In: **MoDISE-EUS**. [S.l.: s.n.], 2008. p. 22–33.

SEMMAK, F.; GNAHO, C.; LALEAU, R. Extended kaos method to model variability in requirements. In: **Evaluation of Novel Approaches to Software Engineering**. [S.l.]: Springer, 2009. p. 193–205.

SEMMAK, F.; LALEAU, R.; GNAHO, C. Supporting variability in goal-based requirements. In: IEEE. **2009 Third International Conference on Research Challenges in Information Science**. [S.l.], 2009. p. 237–246.

SUTCLIFFE, A.; SAWYER, P. Modeling personalized adaptive systems. In: SPRINGER. **International Conference on Advanced Information Systems Engineering**. [S.l.], 2013. p. 178–192.

YU, E. **Modelling strategic relationships for process reengineering**. [S.l.]: Thesis (Ph.D.)—University of Toronto, 1995. (Canadian theses). Disponível em: <https://books.google.com.br/books?id=n1yqSgAACAAJ>. Acesso em: 5 jun. 2019. ISBN 9780612028876.

YU, E. S. **Conceptual Modeling**: foundations and applications. In: BORGIDA, A. T.; CHAUDHRI, V. K.; GIORGINI, P.; YU, E. S. (Ed.). Berlin, Heidelberg: Springer-Verlag, 2009. cap. Social modeling and i*, p. 99–121. ISBN 978-3-642-02462-7. Disponível em: http://dx.doi.org/10.1007/978-3-642-02463-4_7. Acesso em: 25 abr. 2019.

APÊNDICE A – STRING DE BUSCA PARA CADA BASE DE DADOS

Quadro 16 – String de busca para cada base de dados.

Base de Dados	String de Busca
ACM	acmdlTitle:(((KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))) OR keywords.author.keyword:(((KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))) OR recordAbstract:(((KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process)))
El Compedex	(KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))
IEEE Xplore	(KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))
Science Direct	tak(((KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND (goal modeling OR goal modelling OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process)))
SCOPUS	TITLE-ABS-KEY ((KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND ("goal modeling"OR "goal modelling"OR goal-oriented) AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process)))
Springer	(KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND ("goal modeling"OR "goal modelling"OR "goal-oriented") AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))
Scholar	(KAOS OR "Knowledge Acquisition in Automated Specification") AND Requirements AND ("goal modeling"OR "goal modelling"OR "goal-oriented") AND ((extension OR extends OR extended OR extensibility) OR (pattern OR profile OR approach OR process))

Fonte: elaborado pela autora.

APÊNDICE B – PROCEDIMENTOS PARA A BUSCA NAS BASES DE DADOS

Quadro 17 – Procedimentos para a busca nas bases de dados.

Base de Dados	Etapas
ACM	<ol style="list-style-type: none"> 1. Na página inicial, acessar a <i>advanced search</i>; 2. Selecionar <i>show query syntax</i>; 3. No campo <i>edit query</i> colar o texto da string; 4. Clicar no botão <i>search</i>.
El Compedex	<ol style="list-style-type: none"> 1. Na página inicial, selecionar <i>search</i> (no menu) e em seguida <i>expert search</i>; 2. Colar a string no campo de pesquisa, em seguida clicar em <i>date</i> e refinar a pesquisa por ano de publicação alterando o campo definindo a busca para artigos de <i>1990-2019</i>; 3. Clicar no botão de pesquisa; 4. Na página de resultados, refinar a pesquisa por idioma (selecionando inglês).
IEEE Xplore	<ol style="list-style-type: none"> 1. Na página inicial, clicar em <i>advanced search</i>; 2. Clicar em <i>command search</i>; 3. Colar a string de busca no campo de busca e clicar em <i>search</i>. 4. Selecionar <i>conference e journal</i>.
Science Direct	<ol style="list-style-type: none"> 1. Na página inicial, clique em <i>advanced search</i>; 2. Colar a string no campo de busca por termos e adicionar 1990-2019 no campo de busca por ano e clicar no botão de buscar.
SCOPUS	<ol style="list-style-type: none"> 1. Na página inicial, selecionar <i>advanced</i>; 2. Colar a string de busca na <i>query string</i>; 3. Na página de resultados refinar a busca por idioma (selecionando inglês).
Springer	<ol style="list-style-type: none"> 1. Na página inicial, colar a string de busca no campo de pesquisa e clicar no botão de pesquisa; 2. Clicar em <i>data published</i> e no campo adicionar 1990-2019 para refinar a busca; 3. Refinar a pesquisa por idioma (selecionando inglês).
Scholar	<ol style="list-style-type: none"> 1. Na página inicial, colar a string de busca no campo de pesquisa e clicar no botão de pesquisa; 2. Clicar em <i>período específico</i> e no campo adicione 1990-2007 para refinar a busca; 3. Repetir o procedimento do passo 1 e adicionar 2008-2011 no em <i>período específico</i>; 4. Repetir o procedimento do passo 1 e adicionar 2012-2014 no em <i>período específico</i>; 5. Repetir o procedimento do passo 1 e adicionar 2015-2019 no em <i>período específico</i>.

Fonte: elaborado pela autora.

APÊNDICE C – FORMULÁRIO DE EXTRAÇÃO

Quadro 18 – Formulário de extração.

Dados do estudo	Valores	Questão de pesquisa relacionada
Identificação do estudo	Referência para cada estudo	Visão geral
Título, autores, ano, nome da conferência/periódico	Informação dos artigos	Visão geral
Tipo de publicação	Capítulo de livro, conferência ou periódico	Visão geral
Tipo de validação	Estudo de caso, experimento, exemplo de uso, entrevista, questionário, sem avaliação	Visão geral
Áreas de aplicação	Sistemas adaptativos, sistemas autônômicos, segurança/privacidade/vulnerabilidade, processos organizacional/negócio, riscos, sistemas de ambiente, aspectos, continuidade de negócios, proteção, escalabilidade e serviços WEB	QP1
Possui extensão base	Sim ou não	QP2
Definição de conceitos	Apresenta definição, não apresenta definição, apresenta definição parcialmente	QP3
Nível de sintaxe da extensão	Abstrata, concreta, ambas	QP4
Tipo de extensão	Conservativa, não conservativa	QP5
Completo do metamodelo	Completo, nós ausentes, links ausentes, nós e links ausentes	QP5
Tipo de construção	Nó, link, nó e link	QP6
Definição de ferramenta	Possui ferramenta case, não possui ferramenta case	QP6

Fonte: elaborado pela autora.

APÊNDICE D – AVALIAÇÃO DE QUALIDADE

Quadro 19 – Avaliação de qualidade.

Artigo	CAQ1	CAQ2	CAQ3	CAQ4	CAQ5	CAQ6	CAQ7	CAQ8
(CHENG <i>et al.</i> , 2009)	SC	SP						
(DÍAZ <i>et al.</i> , 2006)	SC	SC	SC	SC	SP	SC	NS	NS
(GIL; ARAÚJO, 2009)	SC	SC	SC	SC	SC	NS	SC	SC
(CAILLIAU; LAMSWEERDE, 2013)	SC	SP						
(LAMDDI, 2017)	SC	SC	SP	SC	SC	SC	SC	NS
(SEMMAK <i>et al.</i> , 2009a)	SC	SC	SC	SC	SC	NS	NS	NS
(SEMMAK <i>et al.</i> , 2008)	SC	SC	SC	SC	SC	NS	NS	NS
(PONARD; TOU- ZANI, 2017)	SC	SC	SC	SC	SC	NS	SP	SC
(LEDRU <i>et al.</i> , 2011)	SC	SC	SP	SC	SC	SC	SC	SC
(BARESI <i>et al.</i> , 2010)	SC	SC	SC	SC	NS	SC	NS	SP
(FAVERI <i>et al.</i> , 2016)	SC							
(ARENAS <i>et al.</i> , 2015)	SC	SC	SC	SC	SC	NS	SP	SC
(BROWN <i>et al.</i> , 2006)	SC	SC	SC	SC	SP	NS	SC	NS
(HASSAN <i>et al.</i> , 2008)	SC	SC	SP	SC	NS	SC	NS	SC
(NAGEL <i>et al.</i> , 2013)	SC	SC	SC	SC	SC	NS	NS	NS
(SUTCLIFFE; SAWYER, 2013)	SC	SC	SC	SC	SC	SC	SP	SC
(HUANG <i>et al.</i> , 2009)	SC	SC	SC	SC	SC	SC	NS	NS
(SEMMAK <i>et al.</i> , 2009b)	SC	SC	SC	SC	SC	SC	NS	NS
(DIAS <i>et al.</i> , 2009)	SC	SC	SC	SC	SC	SC	NS	NS
(ISLAM, 2011)	SC	SC	SC	NS	SC	SC	SC	NS
(AHMAD <i>et al.</i> , 2012)	SC	SC	SC	SP	SC	SC	SC	NS
(BRUNET <i>et al.</i> , 2008)	SC	SC	SC	SC	SC	SC	NS	SC

Fonte: elaborado pela autora.

APÊNDICE E – ESTUDOS SELECIONADOS

Tabela 1 – Estudos selecionados.

Referência	Título	Autores	Ano	Local da Publicação
(CHENG <i>et al.</i> , 2009)	A Goal-Based Modeling Approach to Develop Requirements of an Adaptive System with Environmental Uncertainty	Betty H.C. Cheng, Pete Sawyer, Nelly Bencomo e Jon Whittle	2009	Model Driven Engineering Languages and Systems
(DÍAZ <i>et al.</i> , 2006)	Analysis and Verification of Time Requirements Applied to the Web Services Composition	Gregorio Díaz, María-Emilia Cambroner, M. Llanos Tobarra, Valentín Valero e Fernando Cuartero	2006	Web Services and Formal Methods
(GIL; ARAÚJO, 2009)	AspectKAOS- Integrating Early-Aspects into KAOS	André Gil e João Araújo	2009	EA '09 Proceedings of the 15th workshop on Early aspects
(CAILLIAU; LAMSWEERDE, 2013)	Assessing requirements-related risks through probabilistic goals and obstacles	Antoine Cailliau e Axel van Lamswerde	2013	Requirements Engineering, 2013
(LAMDDI, 2017)	Developing Dependability Requirements Engineering for Secure and Safe Information Systems with Knowledge Acquisition for Automated Specification	Mohammed Abu Lamddi	2017	Journal of Software Engineering and Applications, 2017

(SEMMAK <i>et al.</i> , 2009a)	Extended KAOS method to model variability in requirements	Farida Semmak, Christophe Gnaho e Régine Laleau	2010	Communications in Computer and Information Science
(SEMMAK <i>et al.</i> , 2008)	Extended Kaos to support variability for goal oriented requirements reuse	Farida Semmak, Christophe Gnaho e Régine Laleau	2008	International Workshop on Model Driven Information Systems Engineering: Enterprise, User and System Models (MoDISE-EUS'08) - CAISE 2013, 25th International Conference on Advanced Information Systems Engineering.
(PONSARD; TOUZANI, 2017)	Extending land administration domain models with a goal perspective	Christophe Ponsard e Mounir Touzani	2017	GISTAM 2017 - Proceedings of the 3rd International Conference on Geographical Information Systems Theory, Applications and Management
(LEDRU <i>et al.</i> , 2011)	From KAOS to RBAC A Case Study in Designing Access Control Rules from a Requirements Analysis	Yves Ledru, Jean-Luc Richier, Akram Idani e Mohamed-Amine Labiadh	2011	2011 Conference on Network and Information Systems Security
(BARESI <i>et al.</i> , 2010)	Fuzzy Goals for Requirements-Driven Adaptation	Luciano Baresi, Liliana Pasquale e Paola Spoletini	2010	2010 18th IEEE International Requirements Engineering Conference

(FAVERI <i>et al.</i> , 2016)	Goal-driven deception tactics design	Cristiano De Faveri, Ana Moreira e Vasco Amaral	2016	2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)
(ARENAS <i>et al.</i> , 2015)	Goal-Oriented Requirement Engineering Support for Business Continuity Planning	Alvaro E. Arenas, Philippe Massonet, Christophe Ponsard e Benjamin Aziz	2015	2nd International Workshop on Conceptual Modeling in Requirements and Business Analysis (MreBA)
(BROWN <i>et al.</i> , 2006)	Goal-oriented Specification of Adaptation Requirements Engineering in Adaptive Systems	Greg Brown, Betty H.C. Cheng, Heather Goldsby e Ji Zhang	2006	Proceedings of the 2006 International Workshop on Self-adaptation and Self-managing Systems
(HASSAN <i>et al.</i> , 2008)	Goal-oriented, B-based formal derivation of security design specifications from security requirements	Riham Hassan, Shawn Bohner, Sherif El-Kassas e Mohamed Eltoweissy	2008	Proceedings of the The Third International Conference on Availability, Reliability and Security, ARES 2008
(NAGEL <i>et al.</i> , 2013)	Kaos4SOA - Extending KAOS models with temporal and logical dependencies	Benjamin Nagel, Christian Gerth, Jennifer Post e Gregor Engels	2013	FORUM CAISE 2013, 25th International Conference on Advanced Information Systems Engineering.
(SUTCLIFFE; SAWYER, 2013)	Modeling Personalized Adaptive Systems	Alistair Sutcliffe e Pete Sawyer	2013	CAISE 2013, 25th International Conference on Advanced Information Systems Engineering.

(HUANG <i>et al.</i> , 2009)	Role Engineering with SKAOS for Systems Employing RBAC	Chao Huang, Jianling Sun, Xinyu Wang e Yuanjie Si	2009	2009 International Conference on Networking and Digital Society
(SEMMAK <i>et al.</i> , 2009b)	Supporting variability in goal-based requirements	Semmak Farida, Laleau Régine e Gnaho Christophe	2009	Proceedings of the 2009 3rd International Conference on Research Challenges in Information Science, RCIS 2009
(DIAS <i>et al.</i> , 2009)	Towards a domain specific language for a goal-oriented approach based on KAOS	Ana Dias, Vasco Amaral e Joao Araújo	2009	2009 Third International Conference on Research Challenges in Information Science
(ISLAM, 2011)	Towards a framework for offshore outsource software development risk management model	Shareeful Islam e Siv Hilde Houmb	2011	Journal of Software, 2011
(AHMAD <i>et al.</i> , 2012)	Using RELAX, SysML and KAOS for ambient systems requirements modeling	Manzoor Ahmad, Jean-Michel Bruel, Regine Laleau e Christophe Gnaho	2012	Procedia Computer Science
(BRUNET <i>et al.</i> , 2008)	Using variants in KAOS goal modelling	Joël Brunet, Farida Semmak, Régine Laleau, Christophe Gnaho	2008	Proceedings of the Tenth International Conference on Enterprise Information Systems – ISAS

Fonte: elaborada pela autora.