Diretrizes para a Extração Sistemática de Requisitos de Tolerância a Falhas de Sistemas-de-Sistemas a partir de Processos-de-Processos de Negócios

Sidny Almeida Molina[1]*, **Débora Maria Barroso Paiva**[1] **e Maria Istela Cagnin**[1]

[1] Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)
Cidade Universitária, Campo Grande, MS - CEP 79070-900

sidny.molina@gmail.com, {debora.paiva,istela.machado}@ufms.br

Abstract. *This paper presents a set of guidelines for the systematic extraction and specification of fault tolerance requirements in Systems-of-Systems (SoS) based on Processes-of-Business Processes (PoP) models in BPMN. The extracted requirements aim to reach reliability during the SoS interoperability, which automate the corresponding PoP, and the alignment between the technical and business levels. The guidelines were evaluated through a study case. The results indicate that the guidelines support the systematic extraction of fault tolerance requirements from essential information of the business level.*

Resumo. *Este artigo apresenta um conjunto de diretrizes para a extração e especificação sistemática de requisitos de tolerância a falhas em Sistemas-de-Sistemas (SoS), com base em modelos de Processos-de-Processos de Negócios (PoP) em BPMN. Os requisitos extraídos visam alcançar confiabilidade durante a interoperabilidade do SoS, que automatiza o PoP correspondente, e o alinhamento entre os níveis técnico e de negócio. As diretrizes foram avaliadas por meio de um estudo de caso. Os resultados apontam que as diretrizes apoiam a extração sistemática dos requisitos de tolerância a falhas a partir de informações relevantes do nível de negócio.*

1. Introdução

Engenharia de Requisitos (RE) é o processo de descobrir, documentar e manter os requisitos de um sistema de software, garantindo que ele seja compreendido corretamente antes de sua implementação [Jeremy Dick 2017]. RE é fundamental para o sucesso de um projeto de software pois tem como intuito manter o alinhamento das necessidades e expectativas dos usuários e stakeholders, e definir os requisitos de software de maneira adequada, a fim de reduzir ambiguidades e inconsistências. Neste sentido, RE pode desempenhar um papel ainda mais crítico quando diversos sistemas de software individuais (denominados constituintes), geralmente provenientes de organizações distintas, interoperaram entre eles [Ncube and Lim 2018] obtendo funcionalidades mais amplas que não seriam possíveis a partir de cada sistema isoladamente [Maier 1998]. Nesse contexto, esses constituintes formam Sistemas-de-Sistemas (SoS).

SoS é um sistema de software complexo, caracterizado principalmente pelo seu comportamento emergente [Maier 1998], resultante da comunicação entre os seus

*Este trabalho contou com o apoio financeiro da UFMS e da CAPES (código de financiamento 001).

sistemas constitutintes. Por sua vez, os constituintes podem pertencer a diferentes organizações que fazem parte de alianças de organização (resultantes de fusões, parcerias ou aquisições), possuem operacionalização e gestão independentes, geralmente operam de forma distribuída (geograficamente ou virtualmente) e evoluem independentemente do SoS. Portanto, SoS automatiza os processos de negócio de alianças de organizações que juntos compõem processos de negócio complexos e dinâmicos, denominados Processos-de-Processos de Negócio (PoP) [Cagnin and Nakagawa 2021], para alcançar os objetivos estratégicos de negócio dessas alianças. Ressalta-se que quando os constituintes possuem características de Sistemas de Informação (SI) intensivos em software, que se comunicam dinamicamente para coletar, armazenar e processar dados, fornecendo informações e conhecimento que apoiam tomadas de decisões organizacionais, têm-se uma categoria de SoS denominada SoIS (*Systems-of-Information Systems*) [Graciano Neto et al. 2021].

Todavia, durante a comunicação entre os constituintes de um SoS ou SoIS, podem ocorrer falhas, como atrasos na comunicação, perdas de pacotes de dados, indisponibilidade de serviços, ou durante a reorganização da arquitetura dinâmica do SoS quando constituintes são adicionados, alterados ou removidos durante a sua execução, comprometendo a confiabilidade do SoS [Ferreira et al. 2021] e, consequentemente, o funcionamento do negócio da aliança de organizações envolvida [Pereira 2024]. Assim, os requisitos de tolerância a falhas durante a interoperabilidade devem ser tratados ao longo do ciclo de vida do SoS, tornando-se um desafio para a Engenharia de Requisitos. Tolerância a falhas é uma das subcaracterísticas de confiabilidade de software e se refere ao grau em que um sistema opera conforme planejado, apesar da presença de falhas de hardware ou software [ISO/IEC 25010 2011].

Devido às características peculiares de SoS e SoIS, a Engenharia de Requisitos enfrenta desafios na extração, especificação e gestão dos requisitos, por exemplo, múltiplas interações entre diversos sistemas constituintes que têm necessidades de uso diferentes em razão de suas particularidades [Ncube and Lim 2018]. Assim, a identificação adequada de requisitos de tolerância a falhas durante a interoperabilidade de SoS é essencial para garantir sua estabilidade e alinhamento com os objetivos estratégicos da aliança de organizações envolvida. No entanto, de acordo com a literatura, esses requisitos são tratados tardiamente, ou seja, na fase *design* arquitetural, sendo negligenciados nas etapas iniciais da Engenharia de Requisitos. Além disso, esses requisitos não estão em conformidade com o nível de negócio [Molina et al. 2022], colaborando para comprometer a estabilidade e a confiabilidade do SoS.

Alguns trabalhos recentes propõem abordagens para o contexto de requisitos em SoS, como extração e integração de requisitos de confiabilidade e segurança por meio de indicadores de desempenho (*Key Performance Indicators - KPIs*) e de registro de dados de sensores para otimizar a gestão de ativos em um SoS ferroviário [Kumari et al. 2024], co-criação de requisitos emergentes de um SoS de energia inteligente por meio da colaboração e alinhamento de metas coletivas de diversos stakeholders [Chitchyan 2024], e extração de requisitos de segurança a partir da observação de comportamentos de carros autônomos por drones, em cenários operacionais perigosos, para estabelecer uma arquitetura de controle de um SoS em um domínio crítico [Al-Shareefy et al. 2023]. Porém, nenhum dos trabalhos possui foco em tolerância a falhas. Embora esses trabalhos apontam avanços na Engenharia de Requisitos de SoS, a adoção do nível de negócio como

fonte de informação fundamental para extrair requisitos de tolerância a falhas durante a interoperabilidade de SoS não é explorada, constatado também em [Molina et al. 2022]. Para preencher essa lacuna, este trabalho propõe diretrizes para extrair e especificar sistematicamente requisitos de tolerância a falhas durante a interoperabilidade de SoS a partir de modelos de PoP, em particular, Modelo Detalhado de Missão do PoP[1]. Acredita-se que essas diretrizes contribuem para a interoperabilidade plena, que é um dos Grandes Desafios de Pesquisa em SI no Brasil (2016-2026) [Maciel et al. 2017].

2. Método de Pesquisa

O método de pesquisa adotado neste trabalho é qualitativo, exploratório e baseado em *Design Science Research* (DSR), que foca em resolver problemas práticos em contextos específicos por meio de artefatos, gerando novos conhecimentos científicos [Wieringa 2014]. Este trabalho seguiu o ciclo iterativo da DSR, que incluiu o levantamento do referencial teórico e publicações dos resultados (Ciclo de Rigor), a identificação das lacunas e definição dos objetivos para contribuição (Ciclo de Relevância) e o desenvolvimento do artefato e sua avaliação por meio de uma prova de conceito (Ciclo de *Design*). O artefato resultante desta pesquisa é o conjunto de diretrizes de extração de requisitos de tolerância a falhas durante a interoperabilidade de SoS. Esses requisitos são extraídos a partir de elementos específicos da notação BPMN utilizada para representar tratamento de exceções durante a interoperabilidade em modelos de PoP, tomando como base cenários abstratos definidos em [Molina et al. 2023]. Para a construção dessas diretrizes foram executadas quatro etapas. Na primeira etapa, foi conduzida uma análise detalhada para levantar os elementos BPMN representados nos cenários abstratos [Molina et al. 2023] e que são relevantes para identificar e especificar requisitos de tolerância a falhas de SoS. Na segunda etapa, foram definidos os campos necessários para compor a especificação desses requisitos, enquanto a terceira etapa define templates textuais para especificá-los. A quarta etapa define um algoritmo para extrair sistematicamente os requisitos de interesse a partir do Modelo Detalhado de Missão do PoP e especificá-los utilizando os templates definidos. Por fim, a quinta etapa apresenta uma avaliação das diretrizes definidas por meio de um estudo de caso. Cada etapa é descrita sucintamente na próxima seção. Detalhes são obtidos em [Pereira 2024][2].

3. Diretrizes de Extração de Requisitos de Tolerância Falhas de SoS

Inicialmente, na **primeira etapa**, foram identificados os elementos BPMN mais apropriados para representar falhas comuns na interoperabilidade em PoP dirigidos[3] e seus respectivos tratamentos de exceções. Para isso, foi conduzida uma análise dos elementos BPMN utilizados nos cenários abstratos de tratamento de exceções no envio e no recebimento de retorno de mensagens [Molina et al. 2023], que representam o tratamento de exceções durante a interoperabilidade entre os processos constituintes de PoP. Em seguida, foram definidas variáveis para subsidiar a extração de informações valiosas de PoP para a identificação dos requisitos de tolerância a falhas de SoS durante a interoperabilidade. A Tabela 1 mostra um subconjunto das variáveis identificadas, associadas aos elementos BPMN correspondentes. Por exemplo, a variável *Constituinte_Origem* corres-

[1]Esse modelo representa detalhadamente todos os processos constituintes necessários para atingir uma determinada missão do PoP, bem como a interoperabilidade entre eles [Cagnin and Nakagawa 2022].

[2]Disponível em: `https://repositorio.ufms.br/handle/123456789/8784`

[3]Quando os processos constituintes que compõem o PoP são controlados por uma autoridade central, como um processo dominante, visando alcançar as missões do PoP e, consequentemente, do SoS.

ponde à piscina de origem de uma mensagem, representada no modelo PoP pelo elemento BPMN `participant`. Todas as variáveis definidas são apresentadas no Apêndice A[4].

Tabela 1. Variáveis para extração de requisitos de tolerância a falhas de SoS [Pereira 2024]

Variáveis	Elementos BPMN	Fonte de informação
Constituinte_Origem	(Participante/Lane) `participant: id, name, processRef`	Piscina que dá origem ao envio da mensagem
Constituinte_Destino	(Participante/Lane) `participant: id, name, processRef`	Piscina que dá destino da mensagem
Tarefa_envio_msg_com_ev_erro	(Tarefa de envio) `sendTask: id, name` + `boundaryEvent: id, attachedToRef` (refere-se ao ID da sendTask) + `errorEventDefinition: id, outgoing`	Tarefa que realiza o envio de mensagem
Ev_interm_envio_msg	(Evento intermediário de envio) `intermediateThrowEvent: id, name, incoming, outgoing` + `messageEventDefinition: id`	Evento intermediário que realiza o envio de mensagem
Realizar_tratamento_de_exceção_de_envio	(Subprocesso) `subProcess: id, name, incoming, outgoing`	Tarefa que irá realizar o tratamento de exceção da tarefa de envio de mensagem que falhou
Momento_Falha_Envio	Composta pela tarefa de envio (`sendTask = id, name`) com o evento intermediário de erro (`errorEventDefinition`) anexo em sua borda (`boundaryEvent`)	Observada pela tarefa de envio com evento de erro anexo na borda da tarefa
Solução_falha_envio	Composta por evento intermediário de tempo (`intermediateCatchEvent = id, name`), tarefa de envio (`sendTask = id, name`), tarefa de recebimento (`receiveTask = id, name`), tarefa de serviço (`serviceTask = id, name`), tarefa manual (`manualTask: id`), desvio exclusivo (`exclusiveGateway: id`)	composta por todos os elementos identificados após cada falha até o final do segundo desvio exclusivo final (encerramento) do tratamento de exceções no envio
...	...	...

A **segunda etapa** se refere à definição de campos para compor a especificação dos requisitos. Para isso, os campos presentes em templates existentes de especificação de requisitos de SoS [Cagnin and Nakagawa 2024] e que estão em conformidade com a sintaxe para especificação de requisitos de software [ISO/IEC 29148 2018], foram analisados a fim de identificar aqueles que poderiam ser reutilizados ou adaptados. Ademais, foram criados campos especificamente para o contexto de requisitos de tolerância a falhas durante a interoperabilidade. Os campos identificados nesta etapa são exibidos no Apêndice B[4]. Na **terceira etapa**, foram elaborados templates textuais para especificar os requisitos relacionados a tolerância a falhas no envio de mensagens (Tabela 2) e os requisitos relacionados a tolerância a falhas no recebimento de mensagens entre PoP (Apêndice C[4]). Os templates indicam os campos relevantes para documentar os requisitos (definidos na segunda etapa) e as variáveis correspondentes ao conteúdo dos campos (identificadas na primeira etapa). Na **quarta etapa**, foi elaborado um algoritmo para guiar a extração e especificação sistemática dos requisitos a partir do Modelo Detalhado de cada Missão do PoP de interesse. Antes de utilizar o algoritmo, é necessário garantir que cada Modelo Detalhado de Missão do PoP esteja em conformidade com os cenários abstratos [Molina et al. 2023]. Sucintamente, o algoritmo percorre cada Modelo Detalhado de Missão e elicita os requisitos a partir de informação dos elementos BPMN envolvidos em cada troca de mensagem entre dois processos constituintes que possuem tarefa de envio ou recebimento com evento de erro na aborda e com subprocesso associado, com base nas variáveis definidas. Paralelamente, os conteúdos dos campos dos templates são registrados conforme são identificados. Durante a extração manual realizada por analistas

[4]`https://doi.org/10.6084/m9.figshare.28451789`.

ou engenheiro de requisitos de SoS, é essencial atribuir um identificador único ao campo "ID" do requisito e registrar o nome do SoS no campo "Sujeito".

Tabela 2. Template para especificação detalhada dos requisitos de tolerância a falhas no envio [Pereira 2024]

Campo	Conteúdo
ID	ID da tarefa de envio com falha
Classe	Tolerância a falhas
Sujeito	Sujeito do requisito, no caso é o próprio SoS
Constituinte de origem	*constituinte_origem*
Constituinte de destino	*constituinte_destino*
Momento para ocorrência da falha durante o envio da mensagem	*momento_falha_envio* é observado pela *tarefa_envio_msg_com_ev_erro*
Quais falhas que ocorrem durante o envio da mensagem	Cada *falha_envio* é observada por cada *rotulo_fluxo_sequencia* que sai do *desvio_exclusivo* do subprocesso cujo rótulo é "Realizar_Tratamento_de_Exceção_de_Envio"
Como resolver as falhas durante o envio da mensagem	Cada *solução_falha_envio* é observada por cada *ev_interm_temporal*, por cada *ev_interm_envio_msg* e por cada task após cada *falha_envio*
Ação	**Durante o envio de mensagem do** *constituinte_origem* **para o** *constituinte_destino*, **quando** *momento_falha_envio* *[**ao ocorrer** *falha_envio*, **então** *solução_falha_envio*]
Rastreabilidade	Participant *constituinte_origem* para Participant *constituinte_destino* na SendTask *tarefa_envio_msg_com_ev_erro*

Na **quinta etapa**, foi conduzido um estudo de caso para observar a aplicação das diretrizes de extração definidas com respeito ao apoio na extração sistemática de requisitos de tolerância a falhas no contexto de um SoS real a partir de um Modelo Detalhado da Missão "Adicionar turmas e os respectivos alunos matriculados no Ambiente Virtual de Aprendizagem (AVA)" (Figura 1)[5] de um PoP dirigido (no caso, PoP Educacional da UFMS) sob o ponto de vista de um especialista em BPMN. Durante o estudo de caso, foi seguido o algoritmo definido na etapa anterior para extrair os requisitos de tolerância a falhas do SoS Educacional a partir do PoP correspondente. Nesse PoP, há um total de sete tarefas de envio e de recebimento de retorno de mensagem que podem apresentar falhas durante a interoperabilidade entre os constituintes envolvidos. Neste estudo de caso, a extração manual dos requisitos de tolerância a falhas é baseada nessas tarefas. Devido à limitação de espaço, a Tabela 3 apresenta a especificação de um dos requisitos extraídos (cuja fonte de informação é a tarefa destacada em caixa na cor laranja). Detalhes dos demais requisitos extraídos e especificados durante o estudo de caso podem ser obtidos no Apêndice D[4]. As especificações de todos os requisitos de tolerância a falhas extraídos foram entregues como contrapartida para os analistas de TI da Agência de Tecnologia da Informação e Comunicação (AGETIC) da UFMS. Com base nesses artefatos, os analistas poderão planejar a evolução do SoS Educacional caso não esteja atendendo algum dos requisitos de tolerância a falhas identificados e que são importantes tanto para o negócio quanto para manter a estabilidade do SoS.

De maneira geral, os requisitos extraídos e especificados são considerados úteis, claros e alinhados ao negócio, com potencial para aprimorar o SoS Educacional. Salienta-se que as diretrizes são específicas para SoS direcionados. Além disso, elas estão restritas apenas aos elementos BPMN identificados em [Molina et al. 2023]. As principais ameaças à validade do estudo de caso estão descritas no Apêndice E[4]. Ressalta-se que as diretrizes foram implementadas em uma ferramenta [Costa et al. 2024], que foi avaliada por quatro participantes selecionados por conveniência. Todos os participantes concordaram com a facilidade de uso da ferramenta e a maioria reconheceu a sua utilidade.

[5]Para melhor visualização, a Figura 1 também está disponível no apêndice digital[4].

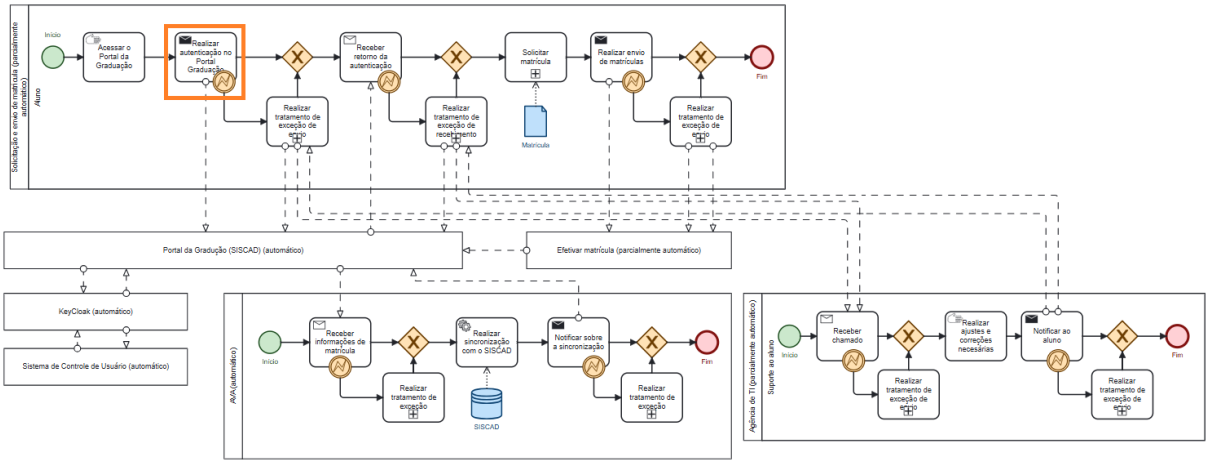

Figura 1. Modelo Detalhado de Missão "Adicionar turmas e os respectivos alunos matriculados no Ambiente Virtual de Aprendizagem (AVA)" [Pereira 2024]

Tabela 3. Requisito de tolerância a falha extraído automaticamente da tarefa de envio "Realizar autenticação no Portal Graduação" [Pereira 2024]

Campo	**Conteúdo**
ID	Activity_0t3g3fe
Classe	Tolerância a falhas
Sujeito	SoS Educacional
Constituinte de origem	Solicitação de matrícula (parcialmente automático)
Constituinte de destino	Sistema Acadêmico de Graduação (SISCAD) (automático)
Momento para ocorrência da falha durante o envio da mensagem	Ao "Realizar autenticação no Portal Graduação".
Quais falhas que ocorrem durante o envio da mensagem	"Falha ao realizar autenticação" e "Portal indisponível".
Como resolver as falhas durante o envio da mensagem	Para "Falha ao realizar autenticação", então "Aguardar intervalo de tempo" e "Realizar autenticação novamente". Para "Portal indisponível", então "Acionar Agência de TI para verificação", "Aguardar intervalo de tempo" e "Receber retorno da Agência de TI".
Ação	**Durante o envio de mensagem** do "Solicitação de matrícula (parcialmente automático)" **para o** "Sistema Acadêmico de Graduação (SISCAD) (automático)", **quando** "Realizar autenticação no Portal Graduação" **ao ocorrer** "Falha ao realizar autenticação", **então** "Aguardar intervalo de tempo" e "Realizar autenticação novamente"; **ao ocorrer** "Portal indisponível", **então** "Acionar AGETIC para verificação", "Aguardar intervalo de tempo" e "Receber retorno da AGETIC".
Rastreabilidade	Participant "Solicitação de matrícula (parcialmente automático)" para Participant "Sistema Acadêmico de Graduação (SISCAD) (automático)" na SendTask "Realizar autenticação no Portal Graduação".

4. Considerações Finais

As diretrizes apresentadas neste trabalho representam um potencial para avançar a Engenharia de Requisitos em SoS e SoIS, especialmente no processo de extração e especificação de requisitos de tolerância a falhas de SoS durante a interoperabilidade. Além de contribuir para a estabilidade e confiabilidade do SoS ou SoIS, os requisitos obtidos por essas diretrizes promovem um alinhamento ao negócio ao longo do ciclo de vida desses sistemas de software complexos. Este trabalho não apenas preenche lacunas existentes na literatura, mas também abre novas perspectivas para pesquisas futuras e aplicações práticas, especialmente em cenários dinâmicos e heterogêneos de interoperabilidade. Para mitigar limitações do trabalho, planeja-se expandir e adaptar as diretrizes para outros tipos de SoS e SoIS (como reconhecido, colaborativo e virtual), além de conduzir estudos de caso em diferentes domínios para observar a sua flexibilidade.

Referências

Al-Shareefy, H., Butler, M., and Hoang, T. S. (2023). AIC approach for intelligent systems requirements elicitation. In *7th ICSRS*, Bologna, Italy.

Cagnin, M. and Nakagawa, E. (2024). Processes-of-business processes: A novel information source of systems-of-systems requirements. *Requirements Eng.* Under review.

Cagnin, M. I. and Nakagawa, E. Y. (2021). Towards dynamic processes-of-business processes: a new understanding. *Business Process Management Journal*.

Cagnin, M. I. and Nakagawa, E. Y. (2022). M-PoP: leveraging the systematic modeling of processes-of-business processes. *Business Process Management Journal*.

Chitchyan, R. (2024). What can requirements engineering do for emerging system of systems? case of smart local energy. In *46th ICSE-SEIS*, Lisbon, Portugal.

Costa, M., Molina, S., Paiva, D., and Cagnin, M. (2024). PoP-ARE: a tool for extracting systems-of-systems non-functional requirements from processes-of-business processes. In *XXXVIII SBES*.

Ferreira, F. H., Nakagawa, E. Y., and dos Santos, R. P. (2021). Reliability in software-intensive systems: Challenges, solutions, and future perspectives. In *47th SEAA*.

Graciano Neto, V. V., Araújo Lebtag, B. G., Teixeira, P. G., et al. (2021). Expanding frontiers: Settling an understanding of systems-of-information systems. *arXiv e-prints*.

ISO/IEC 25010 (2011). ISO/IEC 25010:2011 - Systems and soft. eng. — Systems and soft. Quality Req. and Evaluation (SQuaRE) — System and soft. quality models.

ISO/IEC 29148 (2018). ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes – Requirements Engineering.

Jeremy Dick, Elizabeth Hull, K. J. (2017). *Requirements engineering*. Springer Cham.

Kumari, J., Karim, R., Dersin, P., and Thaduri, A. (2024). A performance-driven framework with a system-of-systems approach for augmented asset management of railway system. *International Journal of System Assurance Eng. and Manag.*

Maciel, R. S., David, J. M., Claro, D., and Braga, R. (2017). Full Interoperability: Challenges and Opportunities for Future Information Systems. In *I GranDSI-BR*. SBC.

Maier, M. (1998). Architecting Principles for Systems-of-Systems. *Systems Engineering*.

Molina, S., Costa, M., Nazário, A., Paiva, D., and Cagnin, M. (2023). Cenários abstratos de tratamento de exceções na interoperabilidade de processos-de-processos de negócios. In *V MSSiS*, Campo Grande-MS.

Molina, S., Paiva, D., and Cagnin, M. I. (2022). Tratamento de Requisitos de Confiabilidade de Sistemas-de-Sistemas: Um Mapeamento Sistemático da Literatura. In *XXV CIbSE*, pages 315–329, Córdoba, Argentina.

Ncube, C. and Lim, S. L. (2018). On systems of systems engineering: A requirements engineering perspective and research agenda. In *26th RE*, Banff, AB, Canada.

Pereira, S. d. A. M. (2024). Abordagem baseada em Cenários para Extrair Requisitos de Tolerância a Falhas de Sistemas-de-Sistemas a partir de Processos-de-Processos de Negócio. Master's thesis, Facom, UFMS.

Wieringa, R. (2014). *Design Science Methodology for Information Systems and Software Engineering*. Springer.