



DISSERTAÇÃO DE MESTRADO PROFISSIONAL

**Proposta de um modelo para documentar os requisitos
conversacionais de Chatbots**

Larissa Pereira Gonçalves

Programa de Pós-Graduação Profissional em Engenharia Elétrica

DEPARTAMENTO DE ENGENHARIA ELÉTRICA
FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

DISSERTAÇÃO DE MESTRADO PROFISSIONAL

**Proposta de um modelo para documentar os requisitos
conversacionais de Chatbots**

Larissa Pereira Gonçalves

*Dissertação de Mestrado Profissional submetida ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Mestre em Engenharia Elétrica*

Banca Examinadora

Edna Dias Canedo

Orientadora

Prof. André Luiz Marques Serrano, Ph.D,

EPR/FT/UnB

Examinador Interno

Professora Ana Paula Bernardi da Silva, Ph.D, Uni-

versidade Católica de Brasília (UCB)

Examinador externo

FICHA CATALOGRÁFICA

GONÇALVES, LARISSA PEREIRA

Proposta de um modelo para documentar os requisitos conversacionais de Chatbots [Distrito Federal] 2024. xvi, 102 p., 210 x 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2024).

Dissertação de Mestrado Profissional - Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Chatbots
3. Especificação
I. ENE/FT/UnB

2. Requisitos
4. Modelo Conversacional
II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

GONÇALVES, L. (2024). *Proposta de um modelo para documentar os requisitos conversacionais de Chatbots*. Dissertação de Mestrado Profissional, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 102 p.

CESSÃO DE DIREITOS

AUTOR: Larissa Pereira Gonçalves

TÍTULO: Proposta de um modelo para documentar os requisitos conversacionais de Chatbots.

GRAU: Mestre em Engenharia Elétrica ANO: 2024

É concedida à Universidade de Brasília permissão para reproduzir cópias desta Dissertação de Mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Do mesmo modo, a Universidade de Brasília tem permissão para divulgar este documento em biblioteca virtual, em formato que permita o acesso via redes de comunicação e a reprodução de cópias, desde que protegida a integridade do conteúdo dessas cópias e proibido o acesso a partes isoladas desse conteúdo. O autor reserva outros direitos de publicação e nenhuma parte deste documento pode ser reproduzida sem a autorização por escrito do autor.

Larissa Pereira Gonçalves

Depto. de Engenharia Elétrica (ENE) - FT

Universidade de Brasília (UnB)

Campus Darcy Ribeiro

CEP 70919-970 - Brasília - DF - Brasil

DEDICATÓRIA

Dedico o presente trabalho à Deus, que me permitiu chegar até aqui e me deu todos os recursos materiais e imateriais para cumprir meu propósito nesta vida. Dedico também este trabalho aos meus familiares, em especial meus pais, minha irmã, meu cunhado, meu sobrinho e meu namorado. Vocês demonstram dia após dia que nossa família é a base de tudo, e que aconteça o que acontecer, sempre seremos uns pelos outros. Dedico também aos meus bons amigos, colegas de trabalho e de vivência acadêmica, que me incentivaram a não desistir, sofrem as dores e celebram cada conquista junto comigo.

AGRADECIMENTOS

Agradeço primeiramente a Deus, que zela tanto por mim e de quem sinto um carinho infindável vindo das pessoas que Ele coloca em minha vida. Em seguida, agradeço aos meus pais, pessoas batalhadoras que me ensinaram a dignidade do trabalho, a humildade de ser um eterno aprendiz e a proatividade que me fez chegar até aqui. Agradeço pela paciência que exigi todas as vezes em que fiquei acordada até tarde fazendo trabalho, cheguei tarde em casa do trabalho ou fins de semana que passei reclusa por conta de alguma entrega. Infelizmente os momentos em família não aproveitados não voltarão, mas tudo tem um propósito embaixo dos céus. Chegará o dia em que eu poderei retribuir um milésimo de tudo que vocês fizeram por mim. Agradeço a minha irmã, cunhado e sobrinho, que nunca duvidaram do meu potencial, mesmo quando eu sempre duvidei. A alegria de vocês é contagiante e foi um dos meus combustíveis para chegar até aqui.

Agradeço também aos meus amigos de colégio, faculdade e da vida: Fernanda, Jordanna, Daniel, Júlia, Lucas, Rodrigo, Amanda, Bernardo, Luana e muitos outros mais, que não caberiam aqui. Poder dividir minha vida com vocês é muito divertido, mesmo quando o momento é de caos. O bom humor de vocês e as situações semelhantes que nos unem fazem eu me sentir em casa. Ao Marcos Cason, colega de mestrado, meus mais sinceros agradecimentos por todas as risadas, e por me transmitir seus conhecimentos em redes e ataques cibernéticos.

Estendo os agradecimentos com muito carinho a todos os meus colegas de trabalho da Confederação Nacional do Transporte, principalmente ao Luiz (melhor chefe!), Diego, Igor, à toda CTI, Bianca e Gustavo. Vocês são tão divertidos que eu sinto saudade todos os dias quando volto pra casa. Também agradeço imensamente ao meu psicólogo Eduardo, que me ajudou a organizar os pensamentos até chegar aqui.

Não poderia deixar de agradecer ao Jhony, meu namorado e melhor amigo. Muito obrigada por sempre me inspirar, me apoiar em todas as decisões, literalmente salvar a minha vida e torná-la mais leve e mais divertida a cada dia. Tem sido um prazer.

Por fim, tenho muito a agradecer à minha professora orientadora Edna Dias Canedo pela paciência, entusiasmo, disciplina, dedicação ímpar, convicção do potencial, e contribuição tão vasta e significativa neste trabalho.

Agradeço também às instituições relacionadas a este trabalho, principalmente à Universidade de Brasília e à instituição onde trabalho, pela oportunidade de investigar uma temática interessante, atual e necessária. A todos estes, meu muito obrigado!

RESUMO

Contexto: A interação de um usuário com um chatbot é considerada uma das mais semelhantes à interação humana entre os softwares. No entanto, essa antropomorfização torna a elicitación e documentação de requisitos de software mais desafiadora, uma vez que o comportamento e a percepção humana são marcados por nuances e subjetividades. Atualmente, não há padrões ou diretrizes claros para registrar e testar requisitos subjetivos, o que torna ainda mais difícil a obtenção, documentação e validação dos chamados requisitos conversacionais. **Objetivo:** O objetivo deste trabalho é identificar características diferenciais entre requisitos e especificações de chatbots em relação a outros softwares. Para isso, é necessária a análise de recursos e elementos presentes nos artefatos existentes a fim de sintetizar os padrões e elementos mais adequados em um modelo de especificação de requisitos conversacionais para chatbots, abrangendo em detalhes os requisitos conversacionais e emocionais. **Método:** Conduzimos 2 revisões sistemáticas para comparar o estado atual da documentação de requisitos para os diversos tipos de softwares e para chatbots, bem como as práticas conversacionais adotadas. A investigação destas práticas e formatos foi utilizada para a elaboração do Mapa Conversacional Integrado, um modelo de artefato proposto para documentar requisitos conversacionais de chatbots. Em seguida, o modelo foi submetido à validação por meio da simulação de documentação com artefatos normais e o artefato proposto, com a intenção de identificar oportunidades e melhorias em sua usabilidade. **Resultados:** Na avaliação do mapa conversacional os participantes relataram facilidade de uso e fácil aprendizado. Três tipos de instâncias do modelo foram disponibilizadas (tabela, diagrama e protótipo), e o *template* de tabela foi o mais escolhido pelos participantes da segunda fase. A disponibilização do *template* na internet já conta com mais de 100 utilizações orgânicas. **Conclusão e trabalhos futuros:** A proposta do modelo inicial foi submetida à validação e ainda possui espaço para evoluções. Como trabalho futuros, será realizado a aplicação de pesquisas adicionais com grupos mais abrangentes de participantes, a saber, pesquisas com usuários finais e stakeholders, para avaliar a compreensão do artefato.

ABSTRACT

Context: The interaction between a user and a chatbot is often considered one of the most human-like among software interactions. However, this anthropomorphism introduces challenges in eliciting and documenting software requirements, as human behavior and perception are nuanced and subjective. Currently, there are no clear standards or guidelines for recording and testing subjective requirements, making the process of obtaining, documenting, and validating conversational requirements particularly difficult. **Objectives:** This work aims to identify the distinctive characteristics of chatbot's requirements and specifications compared to other softwares. To achieve this, we analyzed existing artifacts to synthesize the most appropriate standards and elements for a conversational requirements specification model tailored to chatbots. This model focuses on detailing both conversational and emotional requirements. **Methods:** We conducted two systematic reviews to compare the current state of requirements specification for different types of software, including chatbots, as well as the conversational practices currently in use. The insights

gained from these reviews were the basis for the creation of the Integrated Conversational Map, our proposed artifact model for documenting chatbot conversational requirements. The model was then validated through simulations that compared traditional artifacts with the proposed model, aiming to identify opportunities for improvement in its usability. **Results:**In the evaluation of the conversational map, participants reported that it was easy to use and learn. Three types of model instances were provided (table, diagram, and prototype), with the table template being the most preferred by participants in the second phase. The template has already been made available online and has seen over 100 organic uses. **Conclusion and Future Work:** The initial model proposal has undergone validation and shows promise but also has room for further development. Future research will involve broader participant groups, including end users and stakeholders, to assess the artifact's comprehensibility and effectiveness.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	PROBLEMA DE PESQUISA	3
1.2	JUSTIFICATIVA	4
1.3	OBJETIVOS	5
1.3.1	OBJETIVO GERAL	5
1.3.2	OBJETIVOS ESPECÍFICOS	5
1.4	RESULTADOS ESPERADOS	5
1.5	METODOLOGIA DE PESQUISA	5
1.5.1	LEVANTAMENTO DAS TÉCNICAS DE DOCUMENTAÇÃO DE REQUISITOS	5
1.5.2	LEVANTAMENTO DE CARACTERÍSTICAS CONVERSACIONAIS DE CHATBOTS	6
1.5.3	PROPOSIÇÃO DO MODELO PARA DOCUMENTAÇÃO DE REQUISITOS CONVERSACIONAIS PARA CHATBOTS	6
1.5.4	VALIDAÇÃO	7
1.6	PUBLICAÇÕES RESULTANTES DESTA PESQUISA	7
1.7	ESTRUTURA DA DISSERTAÇÃO	7
2	REFERENCIAL TEÓRICO	9
2.1	CHATBOTS: CARACTERÍSTICAS E FUNCIONALIDADES	9
2.1.1	EVOLUÇÃO DOS CHATBOTS	9
2.1.2	CLASSIFICAÇÃO	11
2.1.3	FORMA E FLUXOS CONVERSACIONAIS	17
2.1.4	ARQUITETURA DOS CHATBOTS	17
2.2	ENGENHARIA DE REQUISITOS	19
2.2.1	TIPOS DE REQUISITOS DE SOFTWARE	20
2.2.2	DOCUMENTAÇÃO DE REQUISITOS	26
2.3	TRABALHOS CORRELATOS	28
3	REVISÃO SISTEMÁTICA - ESPECIFICAÇÃO E DOCUMENTAÇÃO DE REQUISITOS	29
3.1	QUESTÕES DE PESQUISA	29
3.2	SEQUÊNCIA TEXTUAL DE BUSCA	30
3.3	CRITÉRIOS DE SELEÇÃO	32
3.3.1	CRITÉRIOS DE INCLUSÃO	32
3.3.2	CRITÉRIOS DE EXCLUSÃO	33
3.4	AVALIAÇÃO DE QUALIDADE	33
3.5	EXTRAÇÃO DE DADOS	34
3.6	CONDUÇÃO DA PESQUISA	34
3.7	RESULTADOS DA SLR	35

3.7.1	QP.1 - QUAIS SÃO OS ARTEFATOS UTILIZADOS NA LITERATURA PARA A DOCUMENTAÇÃO DE REQUISITOS EM LINGUAGEM NATURAL?	35
3.7.2	QP.2 - COMO DIFERENTES ARTEFATOS DE DOCUMENTAÇÃO SÃO UTILIZADOS PARA ABORDAR REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS EM PROJETOS DE SOFTWARE?	38
3.7.3	QP.3 - QUAIS SÃO AS DIRETRIZES, ESTRUTURAS OU FORMATOS UTILIZADOS NOS ARTEFATOS DE DOCUMENTAÇÃO DOS REQUISITOS EM LINGUAGEM NATURAL?	42
3.8	SUMÁRIO DO CAPÍTULO	49
4	REVISÃO DE CARACTERÍSTICAS CONVERSACIONAIS EM CHATBOTS	50
4.1	QUESTÃO DE PESQUISA	50
4.2	SEQUÊNCIA TEXTUAL DE BUSCA	50
4.3	CRITÉRIOS DE SELEÇÃO	53
4.3.1	CRITÉRIOS DE INCLUSÃO	53
4.3.2	CRITÉRIOS DE EXCLUSÃO	53
4.4	AVALIAÇÃO DE QUALIDADE	54
4.5	EXTRAÇÃO DE DADOS	55
4.6	CONDUÇÃO DA PESQUISA	55
4.7	RESULTADOS DA SLR	56
4.7.1	QP 1. COMO SÃO DOCUMENTADOS OS REQUISITOS CONVERSACIONAIS DE CHATBOTS, E QUAIS ASPECTOS CONVERSACIONAIS SE FAZEM NECESSÁRIOS NA DOCUMENTAÇÃO DE REQUISITOS?	56
4.7.2	QUAIS ASPECTOS CONVERSACIONAIS ESTÃO PRESENTES NA DOCUMENTAÇÃO DE REQUISITOS?	59
4.8	SUMÁRIO DO CAPÍTULO	61
5	PROPOSTA E VALIDAÇÃO DE DOCUMENTAÇÃO	62
5.1	PROPOSTA DE META-MODELO DE ARTEFATO CONVERSACIONAL	62
5.1.1	REPRESENTAÇÕES DO MAPA CONVERSACIONAL INTEGRADO	63
5.2	VALIDAÇÃO DO MAPA CONVERSACIONAL INTEGRADO	67
5.3	RESULTADOS DA VALIDAÇÃO DO MODELO DE ARTEFATO CONVERSACIONAL	68
5.3.1	<i>Survey</i>	68
5.3.2	UTILIZAÇÃO ORGÂNICA	70
5.4	DISCUSSÃO	71
5.5	MELHORIAS A SEREM IMPLEMENTADAS NA VERSÃO 2.0	72
5.6	LIMITAÇÕES E AMEAÇAS À VALIDADE	72
5.6.1	METAMODELO PROPOSTO	73
5.6.2	PESQUISA	73
5.7	SUMÁRIO DO CAPÍTULO	74
6	DISCUSSÃO E CONCLUSÃO	75
	REFERÊNCIAS BIBLIOGRÁFICAS	77

LISTA DE FIGURAS

1.1	Etapas da engenharia de requisitos e suas interações.	2
1.2	Metodologia de trabalho e etapas de entrega.....	6
2.1	Linha do tempo dos chatbots. Adaptado de Khan & Das, 2018 (1).....	9
2.2	Requisitos funcionais, não funcionais e de chatbots.	21
3.1	Artigos restantes após cada etapa da SLR.	35
3.2	Número de menções de cada artefato de documentação nos artigos selecionados.....	36
4.1	Artigos restantes após cada etapa de condução da SLR.	55
4.2	Distribuição dos anos de publicação dos estudos da SLR.	57
4.3	Contagem de artefatos encontrados nos estudos.	57
4.4	Presença de requisitos conversacionais em cada tipo de artefato.	60
4.5	Requisitos conversacionais encontrados na literatura.....	61
5.1	Representação do mapa conversacional integrado em diagrama UML.....	65
5.2	Representação do mapa conversacional integrado em formato de protótipo.	66
5.3	Respostas por questão na escala <i>likert</i>	70
5.4	Desempenho do mapa conversacional integrado na comunidade do Figma.....	71

LISTA DE TABELAS

2.1	Classificação de chatbots, adaptado de Adamopoulou e Moussiades (2) e Braun e Matthes (3).	12
2.2	Descrição dos tipos de requisito de software. Fonte: Adaptado de Kurtanović et al, 2017 (4).	22
3.1	Questões de pesquisa	30
3.2	Termos PICOC traduzidos em português para construção de sequência de busca.....	31
3.3	Adaptação de sequência textual às diferentes bases bibliográficas.	32
3.4	Artefatos e tipos de documentação encontrados. FR = requisito funcional; NFR: requisito não-funcional.	40
3.5	Padrões em documentação de software.	45
4.1	Termos PICOC - em português - utilizados para construção de sequência de busca.....	51
4.2	Adaptação de sequência textual às diferentes bases bibliográficas.	53
4.3	Estudos restantes após aplicação de todas as etapas da SLR.....	56
4.4	Número de chatbots por classificação agrupada.....	58
5.1	Composição do metamodelo de artefato para documentação de requisitos conversacionais. ..	63
5.2	Mapa conversacional integrado na visualização de tabela.	64
5.3	Comparação entre mapa conversacional integrado e artefatos para especificação de chatbots. ..	67
5.4	Descrição dos questionários	69
5.5	Descrição anonimizada dos participantes.....	70
5.6	Sugestões extras dos participantes para o modelo.	71

1 INTRODUÇÃO

Os chatbots são softwares projetados para simulação de diálogo, mimetizando o comportamento de um ser humano por meio da linguagem natural, e também por meio da realização de tarefas como pesquisas em bancos de dados ou agendamento de consultas em sistemas (2). Houve um crescimento em seu uso durante a pandemia de COVID-9, devido à necessidade inesperada de isolamento social. A restrição imposta das possibilidades de interação presencial forçou muitas organizações a se adequar para o relacionamento digital com seus colaboradores e usuários (5).

O conceito de chatbots remonta a várias décadas, com o exemplo mais antigo conhecido sendo ELIZA. Esta usou técnicas de processamento de linguagem natural para simular a conversa humana e foi projetada para imitar um psicoterapeuta (6). Com o aumento do uso de mídias sociais, dispositivos portáteis, aplicativos de mensagens e o advento da pandemia(7), os chatbots tornaram-se renovado objeto de interesse e investimento para plataformas de tecnologia, permitindo que os desenvolvedores criassem chatbots para suas respectivas plataformas. Este interesse ocasionou a popularização de chatbots em uma variedade de setores, incluindo atendimento ao cliente, comércio eletrônico e assistência médica. A demanda do público também mudou, sendo marcada atualmente por maior autonomia em tarefas online como “faça você mesmo” e autodiagnóstico (8).

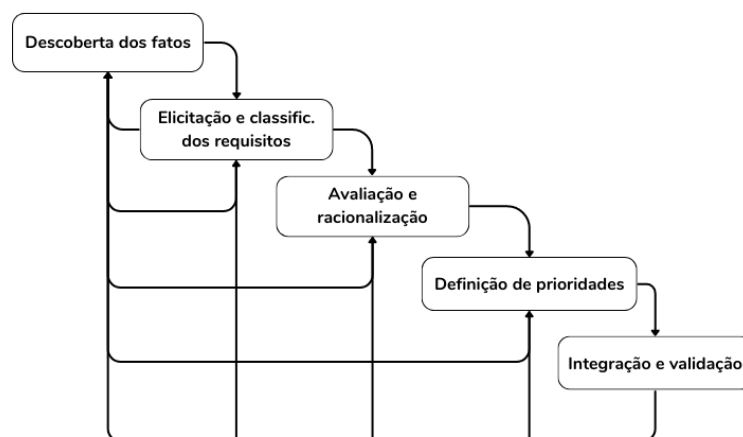
Além da disponibilidade, o crescente interesse pelos chatbots também pode ser explicado por outras vantagens trazidas por eles, como: 1) Interface de usuário conversacional, mimetizando a interação humana (9); 2) Personalização, que permite customização para usuários ou necessidades específicas; 3) Disponibilidade sem restrições, pois funcionam sempre que um usuário entra em um prompt, e não apenas em horário comercial; 4) Automação, reduzindo a necessidade de intervenção humana (10); 5) Integração, pois podem ser integrados a outros serviços e plataformas, como mídias sociais, aplicativos de mensagens e sites de comércio eletrônico. 6) Aprendizado de Máquina e Inteligência Artificial (AI), há uma melhora em sua capacidade de entender, prever as necessidades do usuário, fornecer respostas específicas e até mesmo se comportar de uma determinada maneira (11).

Para garantir o cumprimento dos objetivos estabelecidos para o chatbot, é essencial realizar a identificação das informações pertinentes ao comportamento desejado: identificação e autenticação de usuários, persistência de perfis de usuário, contexto conversacional volátil, generalização, perguntas e respostas, mapeamento personalizado de intenção e ação, ações customizadas, ações espaçadas, proatividade e contextualização de respostas (12).

A identificação das expectativas envolve o processo de levantamento de requisitos, o qual configura-se como fase imprescindível para elucidar o que devem ser atendido até a obtenção do resultado final (13). Esse procedimento engloba a identificação, análise e documentação das necessidades e expectativas dos usuários, e define o que será desenvolvido (14).

De acordo com Christel e Kang (15), Kotonya e Sommerville (16), e Berenbach et al. (17), a Engenharia de Requisitos compreende 5 atividades principais: descoberta dos fatos, elicitação (ou levantamento) e classificação dos requisitos, avaliação e racionalização (análise), definição de prioridades, integração e

validação de requisitos, conforme apresentado na Figura 1.1.



Adaptado de Christel e Kang, 1992 (18).

Figura 1.1: Etapas da engenharia de requisitos e suas interações.

Todas as etapas da engenharia de requisitos estão intrinsecamente ligadas à garantia de que o software seja capaz de atender às necessidades reais dos usuários e às expectativas do cliente (19). Por meio da definição de escopo, prazo e metodologia, a aplicação das etapas previnem retrabalho, reduzem custos desnecessários, auxiliam na definição de prioridades, no planejamento do projeto e na comunicação efetiva entre a equipe de desenvolvimento e os stakeholders (20).

A primeira e a segunda etapa - descoberta dos fatos e elicitação de requisitos – podem ser conduzidas diretamente ou indiretamente, utilizando diversas abordagens, tais como observação do cotidiano, criação de personas, entrevistas, questionários, sessões de brainstorming, workshops conjuntos com usuários e desenvolvedores (JAD), prototipagem, análise de documentações anteriores, mapeamento de processo, entre outras (19). Cada técnica de descoberta e elicitação das principais necessidades apresenta vantagens e desvantagens (21), que devem ser consideradas levando em conta o tipo de software desejado, a natureza do negócio em questão e a disponibilidade dos stakeholders ou usuários (22).

A parte mais difícil do desenvolvimento de um software é determinar e documentar precisamente o que deve ser desenvolvido (23), e, portanto, a documentação do processo de desenvolvimento, quando abrangendo registros de requisitos, especificações técnicas, planos de teste e outros documentos correlatos, desempenha um papel fundamental para assegurar a compreensão, a manutenção e a evolução do sistema (24). Além de facilitar a comunicação entre os membros da equipe, a documentação adequada contribui para evitar retrabalho e promover a transferência de conhecimento, principalmente em contextos de alta rotatividade (23).

O método de elicitação e especificação varia de acordo com o modelo acordado entre a equipe de desenvolvimento e os stakeholders, e a maturidade de software da organização (24). Em abordagens mais tradicionais, é comum a criação de um documento central que consolida todos os requisitos do software, proporcionando uma base para sua evolução futura (25). Por outro lado, em metodologias ágeis, a elicitação é geralmente realizada de forma descentralizada, através de user stories (US) ou protótipos, podendo comportar requisitos funcionais, não funcionais, inclusive de usabilidade (26). Essa abordagem permite alterações mais rápidas, uma vez que a documentação é feita de forma concisa e específica para cada

funcionalidade (27). No entanto, a documentação descentralizada pode dificultar a referência futura, pela dispersão de regras e informações por funcionalidade. Na prática, a documentação em metodologias ágeis tende a ser escassa ou incompleta (28).

No contexto dos chatbots, a ênfase recai principalmente na conversação em si, em contraste com outras aplicações que enfatizam o uso de recursos visuais. Consequentemente, surge a necessidade de levantar requisitos específicos para o domínio conversacional (29), que determinam a forma como o diálogo é iniciado, quais recursos estão disponíveis para o usuário, como o chatbot deve responder e quando encerrar uma conversa. Diante desse contexto, este trabalho tem por objetivo investigar/propor um modelo para apoiar na documentação dos requisitos de chatbots, visando fornecer aos profissionais da área de Tecnologia da Informação e Comunicação (TIC) um template de organização, agrupamento e construção da documentação, com ênfase na conversação, visando mitigar a lacuna documental existente (30, 31).

1.1 PROBLEMA DE PESQUISA

A complexidade dos requisitos de chatbot cresce a cada dia, acompanhando a tendência de ascendente popularidade e adoção de chatbots em diversos setores (32). A interação com uma quantidade cada vez maior de seres humanos, com diferenças culturais, sociais, políticas e filosóficas implica na necessidade de especificar e documentar de maneira precisa os requisitos conversacionais desses sistemas, que diferem substancialmente dos demais tipos por simular uma pessoa humana (33). Ao criar artefatos que capturam os fluxos de conversa e os conteúdos obrigatórios, é possível aproximar a interação final da inicialmente desejada entre o chatbot e os diversos tipos de usuários (34).

Além disso, a compreensão dos requisitos conversacionais dos chatbots contribui para a transparência e a confiança na interação homem-máquina (35). A definição assertiva do modo como os fluxos de conversa devem fluir e quais conteúdos são obrigatórios contribuem para que os usuários tenham uma expectativa clara do que esperar do chatbot (36). Isso é especialmente relevante em setores sensíveis, como saúde e finanças, onde a precisão e a segurança das informações são fundamentais (37). Entretanto, a literatura apresenta lacunas em relação a estudos comparativos de especificação para chatbots. Sendo assim, este trabalho se propõe a responder 3 questões de pesquisa:

- **QP.1. Quais são as características que diferenciam os chatbots de outros softwares?**

Nessa questão temos como objetivo entender as características diferenciais de chatbots em relação a outros softwares e elencar estas características encontradas.

- **QP.2. Como os requisitos de software são documentados pelos profissionais da engenharia de requisitos de forma a fornecer uma compreensão comum sobre o software e permitir a colaboração de diferentes atores?**

Esta questão investiga os formatos relatados na literatura para a documentação de requisitos de software. Neste tópico, serão investigados os artefatos em linguagem natural, uma vez que esta é comum ao time de software e demais atores envolvidos.

Para melhor detalhamento, esta questão foi dividida em 3 subquestões:

QP.2.1 - Quais são os artefatos utilizados na literatura para a documentação de requisitos em linguagem natural?

QP.2.2 - Como diferentes artefatos de documentação são utilizados para abordar requisitos funcionais e não funcionais em projetos de software?

QP.2.3 - Quais são as diretrizes, estruturas ou formatos utilizados nos artefatos de documentação dos requisitos em linguagem natural?

- **QP.3. Como são documentados os requisitos conversacionais de chatbots, e quais aspectos conversacionais se fazem presentes na documentação de requisitos?**

Por fim, esta questão tem por objetivo elucidar como é registrada e mantida a especificação de requisitos no contexto de chatbots, observando as características diferenciais em relação a outros tipos de software.

1.2 JUSTIFICATIVA

A falta de padronização, disponibilidade e compartilhamento de documentação não funcional adequada por parte das empresas resulta em um ciclo vicioso de aprendizado e falta de modelos básicos para orientar o desenvolvimento de sistemas, com documentos inconsistentes (38). No caso de chatbots, tendo em vista que a conversa em si é o principal fator de levantamento de requisitos, pode-se perder as premissas básicas do projeto (29).

Essa lacuna se torna evidente quando é necessário recriar uma aplicação similar à anterior. A ausência de documentação dos requisitos funcionais e não funcionais em softwares consome um tempo considerável, pois os desenvolvedores precisam gastar esforços extras para compreender o funcionamento do software existente, principalmente quando não existem exemplos para contextualização das regras implementadas (39). Isso pode levar a implicações negativas, como penalidades financeiras, perda de confiança dos clientes e danos à reputação da empresa (40). Portanto, a documentação adequada dos requisitos não funcionais é essencial para evitar a recursão de aprendizado, facilitar a reutilização de modelos básicos, garantir a rastreabilidade em sistemas complexos, atender às exigências de auditoria e mitigar erros ou inconsistências em atualizações futuras (41).

Em particular, os chatbots são aplicações que possuem de forma inerente uma vasta gama de requisitos não visuais, e portanto, precisam de atenção especial com relação às interações esperadas, respostas pré-definidas e requisitos de privacidade, segurança, desempenho e ética (37). Desta forma, o presente trabalho visa preencher uma lacuna na literatura referente a modelos de documentação específicos para chatbots.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

O objetivo deste trabalho é propor um modelo para documentar requisitos conversacionais de chatbots, abrangendo os aspectos associados ao desenvolvimento de software, para apoiar os engenheiros/analistas de requisitos na elaboração e documentação de soluções em chatbots.

1.3.2 OBJETIVOS ESPECÍFICOS

Para atingir o objetivo geral desse trabalho, os seguintes objetivos específicos foram definidos:

- Realizar uma revisão da literatura terciária sobre formas de documentação dos requisitos de software;
- Realizar uma revisão da literatura sobre os tipos de requisitos conversacionais, com o intuito de identificar os aspectos conversacionais usados no desenvolvimento de chatbots;
- Propor um modelo para documentar requisitos conversacionais de chatbot;
- Validar o modelo proposto com profissionais da área;
- Realizar ajustes no modelo proposto, a partir das sugestões de evolução.

1.4 RESULTADOS ESPERADOS

Geração de um modelo para documentar requisitos conversacionais de chatbots, visando apoiar os profissionais da engenharia de requisitos e desenvolvimento de software a registrar, manter e padronizar os requisitos elicitados em relação aos fluxos conversacionais desejados.

1.5 METODOLOGIA DE PESQUISA

1.5.1 Levantamento das técnicas de documentação de requisitos

Esta pesquisa tem caráter teórico e investigativo, realizada através da análise de práticas de documentação de software encontradas na literatura e utilizadas por desenvolvedores e analistas na indústria de software. Esta análise será realizada por meio de revisão sistemática da literatura. Uma revisão sistemática da literatura é um método de pesquisa que envolve a coleta, seleção e análise sistemática de todos os estudos relevantes disponíveis sobre um determinado tópico ou questão de pesquisa, seguindo um protocolo definido com o objetivo de minimizar o viés (42).

Para gerar a primeira parte do resultado esperado, será realizada a revisão terciária de literatura sobre técnicas de especificação e documentação de requisitos em repositórios já existentes e a partir de guias

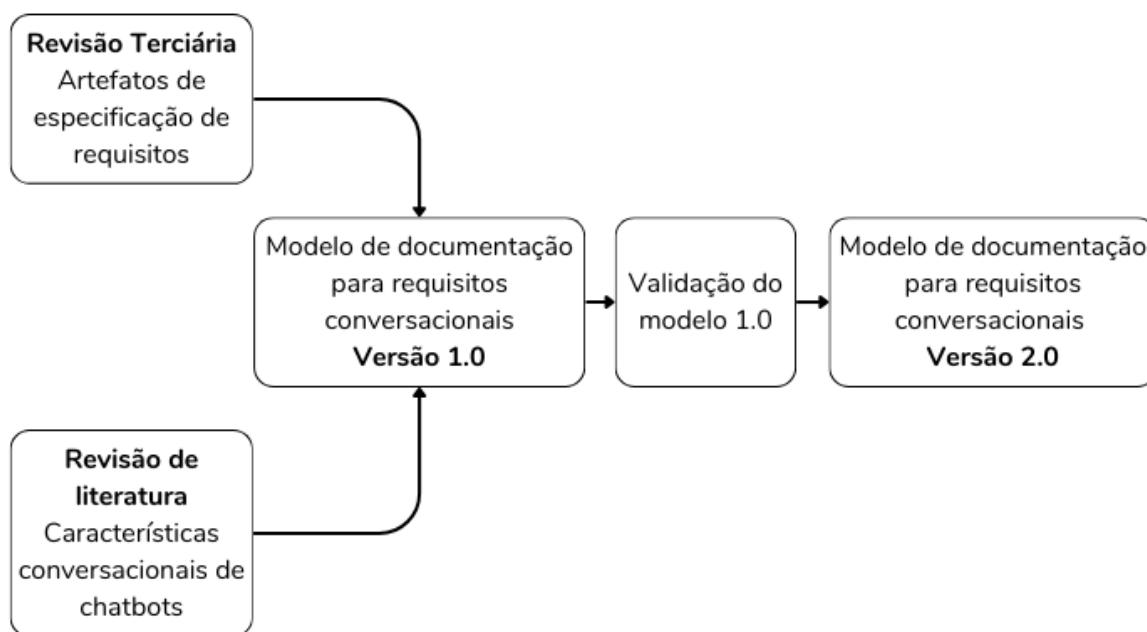
disponibilizados publicamente. A categorização das técnicas, metodologia de busca e sua descrição em detalhes são apresentadas no Capítulo 3.

1.5.2 Levantamento de características conversacionais de chatbots

O fluxo conversacional presente em chatbots é um diferencial em relação a outros softwares, que possuem outras formas de interação com o usuário. Portanto, esta etapa do trabalho tem por objetivo levantar quais características o chatbot precisa apresentar em relação à sua forma de receber entrada e fornecer resposta ao usuário (43). A categorização das documentações existentes, a metodologia de busca e sua descrição em detalhes são apresentadas no Capítulo 4.

1.5.3 Proposição do modelo para documentação de requisitos conversacionais para chatbots

Com base nos tipos de documentação existente e tendo em vista os requisitos conversacionais para um chatbot, foram identificadas na literatura lacunas e pontos de melhoria, que serviram de subsídio na proposta de um modelo de documentação para requisitos subjetivos, principalmente conversacionais.



Fonte: Autoria própria.

Figura 1.2: Metodologia de trabalho e etapas de entrega.

1.5.4 Validação

Para a validação do modelo, os participantes serão convidados a realizar a documentação de um chatbot, utilizando um contexto real, primeiramente de forma livre, e após, utilizando o modelo proposto. Em seguida, responderão ao questionário, cujas perguntas terão foco maior em conhecer a percepção deles acerca da facilidade e usabilidade do modelo proposto.

1.6 PUBLICAÇÕES RESULTANTES DESTA PESQUISA

Essa dissertação gerou as publicações:

1. Gonçalves, L.P; Silva, G. R. S.; Canedo, E. D. (2024) Documentation artifacts for conversation-related requirements specification in chatbots. *27th Workshop on Requirements Engineering*. DOI: 10.29327/1407529.27-19.
2. Gonçalves, L.P; Canedo, E. D.; Da Silva, D.A.; Veiga, C. E. L.; Sousa Júnior, R. T.; Mendonça, F.L.L.; (2024) Requirements documentation containing natural language: A Systematic Tertiary Literature Review. *27th Workshop on Requirements Engineering*. DOI: 10.29327/1407529.27-20.
3. Gonçalves, L.P; Canedo, E. D.; Santos, G. (2024) A Meta-Model for Documenting Conversational Requirements in Chatbots. In: Bertolino, A., Pascoal Faria, J., Lago, P., Semini, L. (eds) *Quality of Information and Communications Technology. QUATIC 2024. Communications in Computer and Information Science*, vol 2178. Springer, Cham. DOI: 10.1007/978-3-031-70245-7_5.

1.7 ESTRUTURA DA DISSERTAÇÃO

Este trabalho está organizado em seis capítulos além deste.

Capítulo 2 - Referencial Teórico: No Capítulo 2 serão apresentados os conceitos teóricos relacionados a chatbots, engenharia de requisitos, classificação e documentação de requisitos. Além disso, serão apresentados os trabalhos correlatos a essa pesquisa e as principais diferenças em relação a ela. Este capítulo responde à questão de pesquisa 1.

Capítulo 3 Revisão Sistemática Terciária da Literatura: São realizadas buscas estruturadas na literatura, em forma de revisão terciária, com o objetivo de responder à segunda questão de pesquisa e suas respectivas subquestões, sumarizando a evolução e situação atual da documentação de requisitos, com ênfase em artefatos que utilizem linguagem natural.

Capítulo 4 Revisão de artefatos que documentem características conversacionais em chatbots: Investigação e listagem dos diferentes elementos presentes em chatbots para composição do fluxo conversacional, respondendo à terceira questão de pesquisa.

Capítulo 5 - Proposta e Validação de Documentação: Este capítulo complementa a terceira questão

de pesquisa, por meio de proposta do modelo de documentação e conduzindo a validação do modelo proposto para analistas de requisitos em projetos de chatbots.

Capítulo 6 - Discussão e Conclusão: Apresenta uma análise geral do que foi pesquisado neste trabalho em cada tema e apresentação do resultado detalhado, e finaliza apresentando as conclusões do estudo.

2 REFERENCIAL TEÓRICO

2.1 CHATBOTS: CARACTERÍSTICAS E FUNCIONALIDADES

2.1.1 Evolução dos chatbots

Historicamente, a capacidade de gerar respostas semelhantes às de um humano tem sido alcançada pela implementação incremental de inteligência de máquina em chatbots (44). Para medir experimentalmente a inteligência de uma máquina, o Teste de Turing foi proposto por Alan Turing em 1950. No teste, um humano interage com uma entidade oculta através de um canal de comunicação de texto. Se o humano não consegue distinguir se está interagindo com uma máquina ou outro humano, então a máquina é considerada como tendo passado no teste, demonstrando um nível de inteligência equivalente ao humano (45).

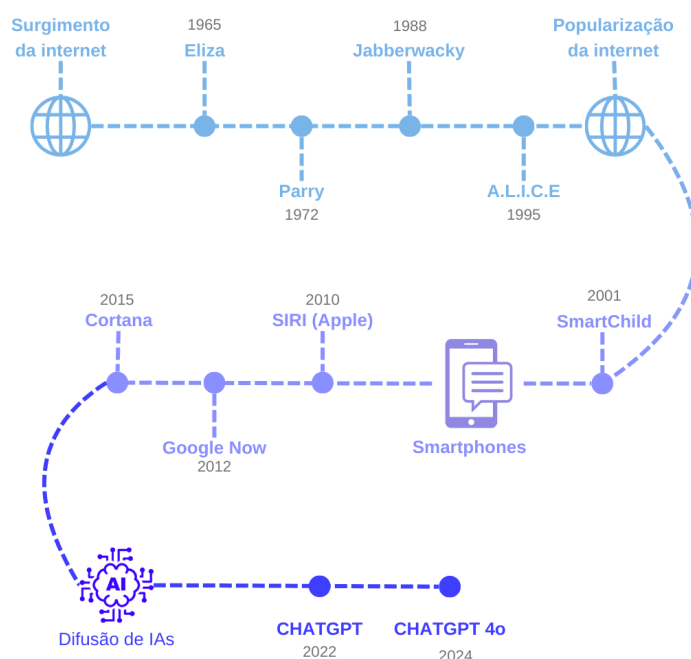


Figura 2.1: Linha do tempo dos chatbots. Adaptado de Khan & Das, 2018 (1).

Os chatbots iniciais surgiram em meados dos anos 60, a partir de programas como ELIZA. Ela foi desenvolvida em 1966 no laboratório de inteligência artificial do MIT para simular uma psicóloga rogeriana, que formula um questionamento a partir das respostas do paciente (46). A estratégia técnica escolhida para mimetizar o comportamento humano foi o uso da metodologia de reconhecimento chamada “*pattern matching*”, que consiste na identificação de palavras-chave, associação com um contexto e modificação dos questionamentos recebidos em novos questionamentos devolvidos ao paciente. Apesar de distante, a abordagem projeta a sensação de continuidade na conversa, tornando-a mais humana (6).

Em 1981 surgiu outro famoso chatbot com viés psicológico, cujo objetivo era mimetizar o comporta-

mento paranoico associado à esquizofrenia - o PARRY (47). Desenvolvido em Stanford, promoveu avanço devido à sua capacidade de simular respostas emocionais e reativas, ao invés de lógicas e racionais, pela implementação de premissas, e personalidade consistente (47).

Após Parry, outras características humanas tiveram sua implementação almejada. Em 1988, Rollo Carpenter criou o Jabberwacky, cujo objetivo era simular uma conversação humana natural, com foco no bom humor e cujo aprendizado se desse através de retroalimentação, armazenando o input do utilizador (48). Já em 1994, a chatbot denominada JULIA conseguia lembrar (49) e aprender (50) sobre o usuário a partir da interação com este ou de informações registradas sobre as ações do mesmo, dando uma maior sensação de inteligência.

Em 1995, Richard Wallace desenvolveu o chatbot A.L.I.C.E. (Artificial Linguistic Internet Computer Entity), que utiliza processamento de linguagem natural (NLP) e linguagem de marcação AIML (Artificial Intelligence Markup Language) (51). O uso de AIML permitiu que o projeto se tornasse colaborativo e aberto. A.L.I.C.E tornou-se conhecida por ganhar três vezes a *Loebner Prize*, uma competição que consiste em um teste de Turing, no qual os chatbots tentam convencer os juízes de que são humanos.

Os chatbots não haviam sido adotados em massa pelo público, até 2001, quando o *SmarterChild* foi lançado no *AOL Instant Messenger* e *MSN Messenger*, grandes canais de comunicação. Ele possuía capacidade para realizar várias funções, como reportar o tempo, notícias, buscar informações na web e até jogar. O sucesso do SmarterChild sinalizou um ponto de inflexão na adoção de chatbots pelo público em geral (52).

Mesmo assim, houve um aumento expressivo do uso de chatbots na década de 2010, quando empresas como Facebook, Microsoft e Google começaram a incorporar chatbots em suas plataformas de mensagens (53). Os avanços na aprendizagem de máquinas e o surgimento de novos paradigmas como *Deep Learning* também contribuíram para este aumento (54).

Diferentemente das mensagens de texto (SMS), que são tarifadas pelas operadoras de telefonia móvel, os aplicativos de mensagens instantâneas operam via internet e podem ser gratuitos se usados em uma rede Wi-Fi (55). Esta facilidade, aliada à ausência de limites de caracteres, incentivou a migração dos serviços. Exemplos desses aplicativos de mensagens instantâneas incluem WhatsApp, Facebook Messenger, Viber e WeChat, e explicam porque chatbots comerciais estão cada vez mais presentes dentro de aplicativos de mensagens, poupando o usuário de telefonar, enviar um e-mail ou visitar o website da empresa (56).

O advento do GPT (Generative Pretrained Transformer), particularmente a versão mais avançada, GPT-4o, e sua variante de chat, ChatGPT, marcaram um ponto de inflexão significativo na história dos chatbots. Desenvolvidos pela OpenAI, esses modelos de linguagem natural representam um salto quantitativo e qualitativo no desenvolvimento de chatbots e em sua capacidade de gerar texto coerente e relevante (57), contando inclusive com a capacidade de geração de imagens, leitura e interpretação de texto por mais formatos de arquivo recebidos (texto, imagem, áudio, documento).

Os chatbots baseados em GPT, como o ChatGPT, foram revolucionários em vários aspectos. Primeiro, eles foram treinados com uma enorme quantidade de dados, dando-lhes uma vasta base de conhecimento para gerar respostas. Além disso, a arquitetura de rede neural *transformer* usada pelo GPT permitiu que os modelos aprendessem a prever a próxima palavra em uma sentença, conferindo-lhes uma capacidade sem

precedentes de gerar respostas similares às humanas (58).

A capacidade do ChatGPT de entender o contexto e gerar respostas coerentes e relevantes para uma ampla gama de consultas foi um salto significativo em relação aos chatbots anteriores, que dependiam em grande parte de scripts pré-definidos e não podiam lidar efetivamente com perguntas fora do seu conjunto de dados de treinamento. Além disso, o ChatGPT, como outros modelos GPT, é capaz de aprender e se adaptar com o tempo, melhorando continuamente sua capacidade de fornecer respostas úteis e precisas pela aprendizagem por reforço (59).

2.1.2 Classificação

Os chatbots podem ser classificados de várias maneiras, e normalmente possuem mais de uma classificação ao mesmo tempo (2). De acordo com Adamopoulou e Moussiades (2), as categorias são: abrangência de domínio, serviço prestado, finalidade, método de geração da resposta, suporte humano, permissões e canais de comunicação. Braun (3) também apresenta as classificações sincronia temporal (*timing*), plataforma, compreensão e canais de comunicação, chamados "entrada/saída"(3), conforme apresentado na Tabela 2.1.

2.1.2.1 Abrangência de domínio:

A abrangência de domínio refere-se à amplitude ou especificidade do contexto ou tópico que o sistema é projetado para lidar (3). Chatbots com domínio aberto não são restritos a um conjunto específico de tópicos ou áreas; Caso o chatbot seja restrito a um ou alguns domínios do saber, é classificado como domínio fechado (2). As áreas mais comuns de atuação dos chatbots são a comercial, a área de entretenimento, produtividade, saúde e educação.

Comercial: Os chatbots de atendimento ao cliente são ferramentas de interação com o usuário final, projetadas por empresas para facilitar o suporte ao cliente e otimizar as operações de negócios. Esses chatbots podem operar 24 horas por dia, todos os dias, em diferentes fusos horários. Além disso, eles podem lidar com um volume maior de consultas simultaneamente, o que pode melhorar a eficiência operacional e reduzir os custos de mão de obra (60). Entretanto, chatbots que possuem falhas de comunicação, erros ou necessitam de mais respostas do consumidor para resolver um problema tendem a diminuir a adoção do usuário (61, 62, 63).

A personalização de tratamento é bem aceita pelos usuários, mas é importante ressaltar que existe um paradoxo de privacidade, pois a personalização da experiência depende da utilização de alguns dados, e muitas vezes não fica claro como estes serão tratados ou se podem ser retirados da base de dados de treinamento caso o cliente deseje (40).

Entretenimento: Ao contrário dos chatbots de atendimento ao cliente com objetivos específicos, os chatbots de conversação podem participar de discussões abertas e diversificadas. Por imitar a maneira como os humanos conversam, são muito utilizados para entretenimento (64). Entretanto, a antropomorfização

Categoria	Classificação
Abrangência de domínio	Domínio aberto Domínio fechado
Serviço prestado	Interpessoal Intrapessoal Inter-agente
Finalidade	Informativo Baseado em conversa / conversacional Baseado em tarefas
Método de geração da resposta	Baseado em regras Baseado em busca/recuperação Generativo
Suporte humano	Auxiliados Não auxiliados
Permissões	Código aberto Código proprietário
Entrada/saída	Texto Voz Imagem
Sincronia temporal	Síncrono Assíncrono
Plataforma	Web Mídias sociais Independente
Compreensão	Notificações Palavras-chave Contextual Personalizado Autônomo

Tabela 2.1: Classificação de chatbots, adaptado de Adamopoulou e Moussiades (2) e Braun e Matthes (3).

apresenta desafios, pois quanto mais similar a um ser humano, maior a expectativa dos utilizadores em relação ao teor comportamental da conversa (65).

Produtividade: Os chatbots de produtividade, também conhecidos como Assistentes Pessoais Inteligentes, são ferramentas digitais projetadas para ajudar os usuários a realizar tarefas de forma mais eficiente e organizada. Siri da Apple, IBM Watson e Alexa da Amazon são alguns exemplos de agentes que aumentam a produtividade por meio de funcionalidades como agendar reuniões, enviar lembretes, organizar tarefas, rastrear o tempo ou até mesmo integrar-se a outras ferramentas de produtividade para automatizar fluxos de trabalho (66).

Saúde: Os chamados chatbots terapêuticos, como o Woebot, são desenvolvidos a partir de terapias e técnicas da psicologia. O Woebot foi desenvolvido utilizando a terapia cognitivo-comportamental como base (67), e auxilia na promoção da saúde mental por meio de conversas diárias e rastreamento de humor do usuário. Alguns chatbots são usados para fornecer conselhos médicos a partir de bibliotecas médicas, como o *One remission* para pacientes de câncer (68). Outro chatbot, o Florence, ainda atua como uma enfermeira particular, tanto provendo informações quanto lembrando horários de remédios e rastreando a saúde do usuário (69). No entanto, este nicho lida com dados pessoais altamente confidenciais, como histórico de doenças e prontuário do paciente (70). Inclusive, podem até mesmo entrar em contextos de ocorrência policial, como o exemplo de um chatbot para saúde mental voltado para crianças que falhou em reconhecer uma situação de abuso sexual (71).

Educação: Chatbots podem ser usados como tutores virtuais, fornecendo aos alunos assistência personalizada 24 horas por dia, de forma complementar aos professores (72). Também é possível que atuem no aumento de interatividade com alunos, permitindo um aprendizado mais profundo (73, 74) e confidencial (75). Outro detalhe é que em casos de chatbots generativos (74, 76), o aluno pode pedir para que este explique de forma diferente, múltiplas vezes. Assim como os chatbots terapêuticos, na educação podem ser usados para fornecer suporte à saúde mental e ao bem-estar dos alunos (76). Por fim, atualmente existe uma utilização expressiva de chatbots na busca por informações, de forma tão ou mais eficiente que os mecanismos de pesquisa existentes (77). Entretanto, é importante notar que a resposta de chatbots a um determinado prompt depende dos dados utilizados em seu modelo, e, portanto, podem oferecer informações incorretas ou inventadas algumas vezes, se houverem vieses durante o treinamento (78, 79).

2.1.2.2 Serviço prestado:

A categoria de serviço prestado, segundo Adamopoulou e Moussiades (2) é referente à proximidade sentimental do chatbot em relação ao usuário. Chatbots interpessoais obtêm informações e conversam com os usuários, mas não são seus companheiros, no sentido de conexão emocional (80). Já os chatbots intrapessoais são companheiros, geralmente possuem personalidade e proveem suporte emocional, como seres humanos (80). Por fim, os chatbots classificados como inter-agentes são os agentes que podem conversar com outros chatbots, a exemplo da integração Alexa-Cortana (80).

2.1.2.3 Finalidade:

A classificação baseada na finalidade considera o objetivo e atuação principal do chatbot. Chatbots informativos são projetados para fornecer informações ao usuário, de forma humanizada ou não, com a possibilidade de acesso a informações de domínio aberto ou domínio fechado. Chatbots conversacionais conversam com o usuário como outro ser humano, e seu objetivo é responder corretamente à frase que lhes foi dada, aplicando inteligência emocional, caso tenha sido projetado para isto (80). Chatbots baseados em tarefas desempenham uma tarefa específica, como reservar um restaurante ou ajudar um usuário (81). Em geral, a finalidade costuma determinar a abrangência de domínio, para cumprir o objetivo a ser alcançado.

2.1.2.4 Método de geração da resposta:

Baseados em regras: Os chatbots baseados em regras formulam suas respostas com base em um conjunto predefinido de regras ou scripts (82). Sua operação é mais direta e menos complexa em comparação com os chatbots baseados em IA, o que os torna muito eficazes em lidar com consultas ou tarefas simples, padronizadas e bem definidas. No entanto, por serem limitados ao seu conjunto de regras, podem apresentar dificuldade em processar variações linguísticas ou responder a perguntas fora de seu escopo programado (82).

Baseados em busca/recuperação: Estes chatbots diferenciam-se um pouco dos agentes baseados em regras, pois consultam e analisam recursos disponíveis usando APIs (83), aplicando a seleção de candidatos a respostas de um índice antes de aplicar a abordagem de correspondência na seleção da resposta (2).

Generativos: Os chatbots generativos - ou seja, baseados em inteligência artificial (IA) - utilizam algoritmos avançados de aprendizado de máquina e processamento de linguagem natural (79). Diferentemente dos chatbots baseados em regras, que são limitados a respostas predefinidas, os chatbots generativos têm a capacidade de aprender com interações anteriores, e seu poder de geração tem aumentado consideravelmente (58). Eles são capazes de lidar com uma variedade mais ampla de entradas dos usuários, incluindo variações linguísticas e erros de digitação (84). Além disso, eles podem gerar respostas mais naturais e conversacionais, proporcionando uma experiência mais antropomórfica ao usuário (85), principalmente em casos nos quais há uma limitação ou falta de interesse em relacionamento interpessoal, como fobia social (86).

2.1.2.5 Suporte humano:

Outra classificação considera a quantidade de suporte (ou auxílio) humano. Chatbots auxiliados por humanos utilizam cálculo humano em pelo menos um elemento para preencher as lacunas causadas pelas limitações dos chatbots (2). Já os chatbots não auxiliados realizam todas as etapas de forma automática. Embora o cálculo humano ofereça mais flexibilidade e robustez, o processamento da informação não é tão rápido quanto uma máquina, o que afeta a escalabilidade da solução (81).

2.1.2.6 Permissões:

Chatbots também podem ser classificados de acordo com as permissões fornecidas por sua plataforma de desenvolvimento (2). Plataformas de código aberto permitem ao criador do chatbot a capacidade de intervir na maioria dos aspectos da implementação. Plataformas de código proprietário, como o Google (87), permitem menor personalização, o que pode comprometer o projeto a depender da necessidade, ou beneficiar o criador do chatbot com dados que essas empresas de código proprietário coletam (2).

2.1.2.7 Entrada/saída:

Texto: Chatbots de texto são chatbots que recebem as entradas em texto e também respondem de forma escrita em linguagem natural (3). São considerados mais flexíveis, e se popularizaram em uma variedade de aplicações (88). Chatbots de texto não têm acesso a expressões humanas, como a fala e expressões corporais e visuais, entretanto, isto torna mais difícil para um usuário de distinguir se é uma máquina ou um humano (89), enquanto chatbots de voz são mais fáceis de reconhecer (90).

Voz: Os chatbots assistentes de voz, também conhecidos como assistentes virtuais pessoais, são ferramentas que interagem com os usuários através do reconhecimento e síntese de voz (91). Eles incluem dispositivos como Amazon Alexa, Google Assistant e Apple Siri. O que os distingue é a capacidade de interagir através do discurso em vez de texto escrito, o que permite um modo de interação mais natural e acessível para muitos usuários. A partir da integração proporcionada pelas tecnologias de internet das coisas (IoT), podem controlar dispositivos domésticos inteligentes, bem como serem incorporados em uma variedade de dispositivos, desde smartphones e alto-falantes inteligentes até carros e eletrodomésticos (92).

Imagem: Chatbots que geram imagens a partir de texto, como MidJourney e Dall-E, foram lançados em 2022 (93). Entretanto, apesar das capacidades sem precedentes destes chatbots, seu desenvolvimento e uso introduzem novos tipos de risco como vieses culturais e raciais, violação de direitos autorais, invasão de privacidade (94) e outros 19 pontos, abordados por Bird et al. (94).

2.1.2.8 Sincronia temporal:

O tempo entre a entrada e a geração de resposta também é fonte de variação entre chatbots. Em geral, assistente de voz ou chat são síncronos, ou seja, recebem um entrada e fornecem uma resposta "imediate". Agentes assíncronos, como bots de redes sociais não necessariamente precisam responder logo após poucos segundos, podendo ser considerados assíncronos. A sincronicidade geralmente impõe desafios em relação ao desempenho de um sistema (3).

2.1.2.9 Plataforma:

Web: Os chatbots de web são interfaces de conversação incorporadas diretamente a um site para melhorar a experiência do usuário, fornecendo suporte imediato e interativo (95). Eles são comumente usados para funções de atendimento ao cliente, como responder a perguntas frequentes, orientar os usuários na navegação do site ou ajudar em processos de compra. É possível que sejam programados para interagir proativamente com os visitantes do site, fornecendo informações relevantes, fazendo perguntas para entender as necessidades do usuário ou incentivando ações específicas, como inscrever-se em uma newsletter ou iniciar um processo de checkout. Além disso, podem ser integrados à navegação do usuário com plugins e permanecerem em contato com o cliente, mesmo depois de este sair do site em questão (96).

Redes sociais: Os chatbots de redes sociais são projetados ou adaptados para operar dentro de plataformas de mídia social como Facebook, Twitter, Instagram, e podem automatizar uma variedade de interações sociais. De acordo com Stieglitz et al. (97), os chatbots sociais podem ser classificados por intenção e imitação do comportamento humano. Em relação à intenção, podem ser benignos, neutros ou maliciosos; Já em relação à habilidade de imitação do comportamento humano, podem apresentar alto ou baixo nível (97).

Chatbots maliciosos com alto nível de imitação, por exemplo, podem ser utilizados para infiltração em organizações e disseminação de notícias falsas (97). Em contrapartida, existem chatbots benignos para detecção de fake news. Como exemplo, o Tribunal Superior Eleitoral (TSE) fez uma parceria com a plataforma Telegram e criou um canal para divulgar informações oficiais sobre as eleições, cujo robô possuía capacidade de tirar dúvidas dos usuários (98). Em 2020, a World Health Organization (WHO) também fez uma parceria com o Facebook para combater a disseminação de informações falsas referentes a pandemia de COVID-19 (99). Mesmo assim, os chatbots de intenção benigna ou neutra precisam ser cuidadosamente gerenciados para evitar interações inadequadas ou spam, que podem levar a uma percepção negativa da marca (100).

Independentes: Chatbots de plataforma independente se referem a aplicações que existem em dispositivos separados. Atualmente, esta classificação se aplica em sua maior parte a assistentes de voz como Alexa da Amazon, que são embarcados em dispositivos próprios (3).

Uma das principais diferenças entre os chatbots assistentes de voz e outros tipos de chatbots é a necessidade de uma palavra-chave específica para iniciar a conversação, geralmente, o nome ou apelido do próprio chatbot (91). Além disso, questões de privacidade e segurança podem ser especialmente importantes, uma vez que esses chatbots podem ter acesso a informações pessoais sensíveis e são frequentemente usados em ambientes privados (101). A voz é um sinal sonoro que revela informações pessoais da pessoa que fala, como estado emocional, estresse, idade e gênero. Apesar de já terem sido propostas alternativas para mitigar o reconhecimento de informações pessoais por voz (102), ainda é uma discussão recente.

2.1.2.10 Compreensão:

O nível de compreensão em um chatbot é percebido como a complexidade de processamento. Os sistemas menos complexos não processam nenhuma entrada, são apenas um sistema de notificação que envia mensagens (3). Sistemas mais complexos dependem da detecção de palavras-chave, e os ainda mais sofisticados levam em consideração o contexto de uma conversa. O próximo nível de compreensão personaliza a experiência do usuário por levar em conta informações sobre o usuário de conversas anteriores ou outras fontes (103). Além disso, existem chatbots que são totalmente autônomos e não apenas se comunicam com humanos, mas também com outros bots(80).

2.1.3 Forma e Fluxos Conversacionais

A base de qualquer chatbot é a conversação com o usuário, a partir de uma forma e um fluxo conversacional (43). A forma conversacional refere-se à apresentação do chatbot em si, estilo, tom e formato das interações. O fluxo conversacional é definido como o grau em que uma conversa é experienciada como suave, eficiente e mutuamente envolvente (104). A experiência subjetiva do fluxo conversacional pode ser produzida por vários aspectos, como qualidade das alternâncias de turno, curtas latências de resposta e poucas interrupções (105). Assim, a forma conversacional refere-se às características objetivas da conversa, enquanto o fluxo refere-se à experiência subjetiva dela (106).

O fluxo conversacional realiza um papel importante no estabelecimento de relações sociais e percepções da realidade (106). A própria validação de opiniões também se dá pela troca de informação com outras pessoas (107). A solidariedade emerge não somente pelo conteúdo da interação social, mas também é fortemente influenciada pela forma de interação (104). Até mesmo na comunicação com animais, o conteúdo e a forma complementam-se de forma relevante para propagar a intenção e comando específico do adestrador (108).

A naturalidade, ou antropomorfismo da interação também pode ser aplicado para superar preconceitos e percepções negativas que os usuários possam ter em relação aos chatbots (109). De acordo com Moore e Arar (110), o design de conversação envolve observação e interação com os usuários, definição de personas e objetivos dos usuários, molde de conversas, definição de persona para o chatbot, antecipação a mensagens do usuário e do agente, prototipação e testes (110).

É de se esperar então que a parte de experiência do usuário na conversação tenha algumas melhores práticas a serem seguidas, a saber: (1) Introdução clara e assertiva, para que o usuário esteja ciente das capacidades do chatbot desde o início; (2) Divulgação progressiva, para dividir as informações em partes gerenciáveis e fáceis de assimilar; (3) O uso de artefatos, como imagens, que podem transmitir instruções complicadas melhor do que palavras (111); O uso de feedback, incluindo indicadores de digitação para mostrar ao usuário que uma resposta está sendo elaborada (111); A capacidade de retornar ao fluxo principal da conversa de maneira elegante (111); A criação de uma personalidade apropriada e consistente ao longo da conversa (110).

2.1.4 Arquitetura dos Chatbots

A maior diferença dos chatbots em relação aos outros software é a sua capacidade de interagir com os usuários através de linguagem natural. Além disso, eles são projetados para aprender e melhorar suas respostas com o tempo, um aspecto que os distingue de softwares tradicionais que não apresentam características de aprendizagem (112).

Os chatbots modernos são construídos utilizando uma combinação de técnicas de processamento de linguagem natural (PLN), aprendizagem de máquinas e representações distribuídas (113). A arquitetura geral de um chatbot envolve três componentes principais: A Natural Language Understanding (NLU) - A Compreensão de Linguagem Natural -, que compreende o significado das declarações e as converte ao formato representacional, o Dialog Manager(DM) - Gerenciador de Diálogo -, que recebe e processa o

formato representacional do NLU e envia ao terceiro componente principal, a Natural Language Generator (NLG), responsável pela geração da resposta (114).

Antes de introduzir os núcleos principais da arquitetura, é importante mencionar o papel da interface do usuário, que é a forma pela qual os usuários interagem e se comunicam com o chatbot. A principal característica da interface de um chatbot é a conversação em linguagem natural, o que difere de outros softwares cujo design de experiência do usuário tem maior predominância de elementos visuais (111). A interface depende do local em que o chatbot é utilizado. Como exemplo, em redes sociais ou aplicativos de mensagens instantâneas, é mais comum que a forma de apresentação seja majoritariamente em mensagens de texto, ou no máximo com a inclusão de alguns botões. Já em assistentes virtuais baseados em comando de voz, a interface do usuário é a própria comunicação ouvida e falada, cujos feedbacks podem ser visuais e sonoros. Alguns chatbots, geralmente implementados em websites, podem possuir interfaces gráficas mais complexas, com botões, menus suspensos, listas ou até mesmo gráficos e imagens (115).

Após interagir com a interface, o usuário fornece uma entrada ao chatbot, que pode ser escrita ou falada. A *Natural Language Understanding* (116), popularmente conhecida como NLU, é o componente que interpreta esta entrada e transforma em uma forma que o chatbot possa entender, por meio de 7 passos:

- **Pré-processamento:** A entrada do usuário é submetida a limpeza de ruídos, normalização de pontuação, conversão de letras maiúsculas em minúsculas, dentre outras formas de limpezas de dados.
- **Tokenização:** A entrada pré-processada é dividida em unidades menores - tokens. Esses tokens podem ser palavras individuais ou até mesmo partes menores, como n-gramas (117).
- **Análise de token:** Cada token separado na fase anterior é analisado para determinar sua forma básica - lematização - e suas características gramaticais, como gênero, número e tempo verbal(117).
- **Reconhecimento de entidades:** A partir daí, é realizado o reconhecimento e separação de entidades nomeadas na entrada do usuário. Uma entidade é uma representação de um objeto do mundo real, podendo descrever pessoas, locais, datas, organizações, entre outros. Para cada tipo de entidade, são atribuídas ações e um tratamento. Por exemplo, em uma viagem, a pessoa é quem viaja, e a data marca em que momento a viagem ocorre. Uma data não pode viajar, e uma pessoa não representa um ponto no tempo (118).
- **Análise sintática:** Com as entidades reconhecidas, é analisada a estrutura gramatical da frase para entender as relações entre as palavras. Isso inclui a identificação das partes da fala (substantivos, verbos, adjetivos, etc.) e a construção de uma árvore de análise sintática que representa a estrutura gramatical da frase (119). A estrutura gramatical fornece subsídio para identificar a intenção do usuário com base na entrada fornecida (119).
- **Extração de intenção e informações adicionais:** Além da intenção, são extraídas as informações adicionais, chamadas *slots*. Os slots são informações específicas que são relevantes para a intenção identificada. Por exemplo, para a intenção "fazer uma reserva de hotel", os slots podem ser "data de check-in", "data de check-out" e "número de quartos"(119).
- Uma vez concluída a etapa de NLU, as informações extraídas, como a intenção e os slots, são passadas para o próximo estágio do processamento do chatbot, que pode ser um gerenciador de

diálogo ou diretamente para a geração de resposta. Algumas plataformas de NLU, como Watson IBM (120) e DialogFlow (87) permitem a criação de chatbots por meio de interfaces gráficas na nuvem sem código e geram logs para rastreamento e correção de erros e exceções (121).

O gerenciador de diálogo é o componente adicional que determina a resposta adequada, responsável por controlar e coordenar a interação entre o chatbot e o usuário, e tomar decisões sobre a próxima ação com base na entrada do usuário e no estado atual da conversa. Ele analisa a intenção do usuário, os slots extraídos e outros contextos relevantes para determinar a resposta mais apropriada a ser fornecida (122). Algumas abordagens e técnicas para implementar um gerenciador de diálogo incluem a definição de um conjunto de regras para mapear as intenções e slots identificados para as respostas correspondentes. Outras abordagens, baseadas em modelos de aprendizagem de máquina, abarcam treinamento dos modelos com dados rotulados para prever a próxima ação com base na entrada do usuário e no histórico da conversa (123).

Após o gerenciador de diálogo, a geração de resposta, também conhecida como *Natural Language Generation* (NLG), é o processo em que um chatbot ou sistema de diálogo cria uma resposta em linguagem natural a partir de texto, dados e até mesmo imagens (124). É a etapa em que o chatbot converte informações internas ou contextos para a saída apropriada (texto ou voz) ao ser humano, por meio de estágios que podem variar de acordo com a complexidade do sistema (124). O sistema determina a estrutura básica da resposta com base no tipo de pergunta ou solicitação do usuário, e quais informações ou dados são relevantes. Pode envolver a recuperação de informações armazenadas em bancos de dados, integração com sistemas externos ou uso de algoritmos para selecionar o conteúdo ou a forma de apresentação mais adequada com base no assunto tratado ou em padrões pré-estabelecidos (124). Além disso, é definido o estilo e o tom da resposta com base na personalidade do chatbot, personalização direcionada ao usuário por meio de conversas anteriores ou com base nas diretrizes da marca. Após a personalização, são aplicadas técnicas de geração de texto para criar a resposta em linguagem natural, área cujas inovações permanecem dinâmicas (103).

A personalização da experiência acontece de forma mais eficiente pelo uso de informações já disponibilizadas pelo usuário no histórico de conversas e armazenadas na base de conhecimento (125), a qual desempenha um papel fundamental no aprendizado de máquina e na inteligência artificial, pela retroalimentação que mimetiza um aprendizado contínuo do chatbot (126).

2.2 ENGENHARIA DE REQUISITOS

A engenharia de requisitos se originou no início do período da informática, embora sua formalização e reconhecimento como um campo específico de estudo tenha ocorrido mais tarde (127). O início do desenvolvimento de software ocorreu durante a década de 1960, e não havia um entendimento uniforme da importância dos requisitos ou de como eles deveriam ser gerenciados. À medida que os sistemas de software tornaram-se mais complexos, reconheceu-se a necessidade de abordagens mais sistemáticas e disciplinadas para o desenvolvimento de software.

No entanto, a engenharia de requisitos só foi reconhecida como um campo distinto na década de 80,

quando os engenheiros observaram que o desenvolvimento eficiente e eficaz de sistemas complexos exigia uma compreensão clara dos requisitos do sistema desde o início do projeto (128). Atualmente, a engenharia de requisitos é parte indispensável de qualquer projeto de desenvolvimento de software ou sistema, devido à crescente complexidade dos sistemas(128), expectativas distintas entre stakeholders, mudanças solicitadas ao longo do projeto e priorização de requisitos quando necessário.

A ausência de compreensão clara e completa dos requisitos dificulta a entrega de um produto de software que satisfaça as necessidades dos usuários e dos *stakeholders*, devidamente encaixado no prazo estipulado e com o valor previamente estabelecido (129). Por isso, a engenharia de requisitos é um processo contínuo, cujos requisitos devem ser constantemente revistos e atualizados à medida que o projeto avança e as circunstâncias mudam, originando a subdisciplina de gestão da mudança de requisitos (130).

De acordo com o *Chaos Report* realizado pelo *Standish Group* em 2020, dentre os projetos de software realizados, apenas 39% foram finalizados com sucesso (131). Um projeto bem sucedido é aquele que cumpriu a tríplice restrição: cronograma, custo e escopo; Os projetos desafiadores são aqueles que cumprem 2 das 3 restrições; e considera-se que o projeto falhou quando foi cancelado antes de ser completado ou foi completado mas não é utilizado.

O mesmo relatório destaca a contínua modernização e uma falha de 19 % dos projetos de forma geral, em decorrência do ambiente de trabalho, da maturidade técnica do time e da relação com os patrocinadores do projeto (121). É visto então que estes três fatores afetam a obtenção de requisitos eficazes e precisos, o que por sua vez reflete na entrega final do produto e na satisfação dos usuários. Para mitigar essa discrepância, é necessário que os analistas e desenvolvedores estejam atentos desde o início à especificação e priorização dos requisitos de software.

2.2.1 Tipos de requisitos de software

Requisitos de software são a descrição de funções que precisam ser desenhadas e desenvolvidas para um determinado sistema ou aplicação (132) e podem ser categorizados de várias maneiras, dependendo da abordagem específica e do contexto. No entanto, uma divisão comum é entre requisitos funcionais e não funcionais (133, 134, 135, 136). Os requisitos funcionais se referem às funções que o software deve realizar e os dados que ele deve processar, ou seja, descrevem as funcionalidades de um sistema (137). Já os requisitos não funcionais especificam como o sistema deve se comportar e impõem restrições sobre o que o sistema funcional pode fazer, ou seja, comportam-se como padrões de qualidade (135).

2.2.1.1 Requisitos funcionais

Os requisitos funcionais são declarações diretas de quais serviços o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações (figura 2.2). Eles descrevem o que o sistema deve fazer, e como funcionalidades diversas devem se relacionar entre si (138).

É importante notar que a forma como os requisitos funcionais são documentados pode variar dependendo da metodologia de desenvolvimento de software utilizada. Por exemplo, em abordagens tradicionais

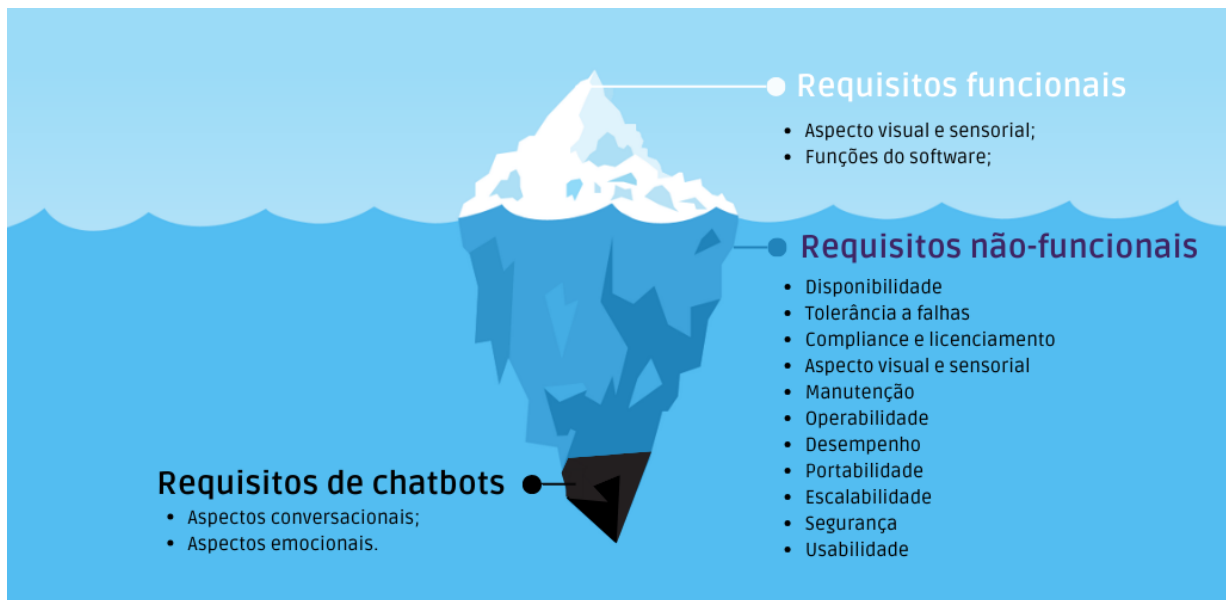


Figura 2.2: Requisitos funcionais, não funcionais e de chatbots.

de desenvolvimento de software, como o modelo em cascata, os requisitos funcionais são geralmente documentados em um Documento de Requisitos de Software. Já em abordagens ágeis, os requisitos funcionais podem ser capturados em histórias de usuário, casos de uso ou backlogs de produto. Apesar da documentação adotada, os requisitos funcionais idealmente devem ser escritos de forma a serem independentes de design ou solução tecnológica (139).

2.2.1.2 Requisitos não funcionais

Os requisitos não funcionais, também conhecidos como requisitos de qualidades, se referem às restrições que o sistema deve respeitar ou qualidades específicas que deve apresentar (135). Eles não estão diretamente relacionados à funcionalidade específica, mas sim à qualidade do serviço, como desempenho, segurança, usabilidade e confiabilidade (134). Os requisitos não funcionais podem ser ainda subdivididos em várias categorias, incluindo requisitos de desempenho, requisitos de segurança, requisitos de confiabilidade, requisitos de usabilidade, requisitos de compatibilidade, requisitos de manutenibilidade e requisitos legais, entre outros (140).

Além destes requisitos presentes na maioria dos softwares, existem requisitos emergentes, como os “conversacionais” e “emocionais”, que surgiram e tornaram-se mais evidentes com o desenvolvimento de novos chatbots (43). Não obstante serem categorias relativamente novas e não tradicionalmente incluídas nas classificações padrão, tendem a ser enquadrados como requisitos não funcionais, pois geralmente se referem a como o sistema deve operar, em vez do que o sistema deve fazer. Por conta disto, também serão incluídos na presente análise como requisitos não funcionais, conforme apresentado na figura 2.2 e tabela 2.2.

Disponibilidade: É um requisito não funcional crítico para muitos sistemas de software que necessitam de alta disponibilidade, ou seja: sistemas bancários online, infraestrutura de rede, aplicativos de comércio

Tipo	Específico de chatbot	Requisito	Descrição
NF	Não	Disponibilidade	Descreve a probabilidade de o sistema estar acessível para um usuário em um determinado momento.
NF	Não	Tolerância a falhas	Grau em que um sistema, produto ou componente opera conforme pretendido, apesar da presença de falhas de hardware ou software.
NF	Não	Compliance e Licenciamento	Certificados ou licenças que o sistema deve possuir, bem como legislações a qual está sujeito.
F	Não	Aspecto visual e sensorial	Descreve o estilo da aparência do produto.
NF	Não	Manutenção	Grau de eficácia e eficiência com o qual um produto ou sistema pode ser modificado pelos mantenedores pretendidos.
NF	Não	Operabilidade	Grau em que um produto ou sistema possui atributos que o tornam fácil de operar e controlar.
NF	Não	Desempenho	Desempenho relativo à quantidade de recursos utilizados nas condições estabelecidas.
NF	Não	Portabilidade	Grau de eficácia e eficiência com que um sistema, produto ou componente pode ser transferido de um hardware, software ou outro ambiente operacional ou de uso para outro.
NF	Não	Escalabilidade	Grau em que um produto ou sistema pode ser eficaz e eficientemente adaptado para hardware, software ou outros ambientes operacionais ou de uso diferentes ou em evolução.
NF	Não	Segurança	Grau em que um produto ou sistema protege informações e dados, de modo que pessoas ou outros produtos ou sistemas tenham o grau de acesso aos dados apropriado para seus tipos e níveis de autorização.
NF	Não	Usabilidade	Grau em que um produto ou sistema pode ser usado por usuários específicos para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto específico de uso.
NF	Sim	Aspectos conversacionais	Majoritariamente aplicado a chatbots, refere-se à capacidade de um software de comunicar-se em linguagem natural de forma educada, fluida e objetiva.
NF	Sim	Aspectos emocionais	Majoritariamente aplicado a chatbots, descreve a habilidade do software em reconhecer emoções, intenções, e respondê-las de forma considerada educada e respeitosa pelos usuários.

Tabela 2.2: Descrição dos tipos de requisito de software. Fonte: Adaptado de Kurtanović et al, 2017 (4).

eletrônico e softwares militares (141). Estes sistemas possuem alta demanda por parte dos usuários, e o tempo de inatividade no geral tem consequências significativas, como perda de receita, confiança do cliente ou até mesmo risco à capacidade efetiva de operações militares (141). No contexto da engenharia de software, a disponibilidade é geralmente definida como a proporção de tempo que um sistema está operacional e acessível para uso. Por exemplo, uma disponibilidade de 99,9 %, comumente referida como "três noves", implica que o sistema estará indisponível por não mais que 8,76 horas em um ano. Alcançar alta disponibilidade geralmente envolve a utilização de técnicas como redundância de hardware e software, balanceamento de carga, monitoramento constante do sistema, estratégia de recuperação de desastres (142) e mitigação ou retirada de interferências (143).

Tolerância a falhas: A complexidade embutida e a flexibilidade inerentes ao desenvolvimento de software o tornam frágeis a mudanças (144). Isso implicava até então projetar sistemas com redundância (diversos meios para se realizar uma determinada ação) (145) e/ou constante manutenção de backups e snapshots (146). Entretanto, alcançar este requisito tornou-se um novo desafio após a popularização da computação em nuvem, a qual trouxe a necessidade de tolerância maior, uma vez que o ambiente apresenta maior volatilidade (147) e espera-se que os arquivos criados e alterações realizadas sejam salvos instantaneamente.

Compliance e licenciamento: É necessário que o software esteja em conformidade com as leis, regulamentos, normas e diretrizes, bem como a aquisição de licenças ou certificações necessárias, no seu país de origem e nos países em que atua. Em termos legais, o software deve aderir a todas as leis aplicáveis no contexto de seu uso, as quais podem variar desde leis de privacidade e proteção de dados, como o Regulamento Geral de Proteção de Dados da União Europeia - GDPR (148), a Lei Geral de Proteção de Dados no Brasil (149) até normas específicas da indústria, como a Lei de Portabilidade e Responsabilidade de Seguros de Saúde dos Estados Unidos (HIPAA) para software médico (150). Quanto ao licenciamento, pode envolver a obtenção de permissões para o uso de tecnologias patenteadas, direitos autorais, ou certificações de conformidade de entidades regulatórias ou de normas. Ficar aquém nesses aspectos pode resultar em sanções legais, perda de confiança do cliente e danos à reputação (151).

Aspecto visual e sensorial: A apresentação visual do software tem um impacto significativo na usabilidade e na satisfação do usuário, incluindo elementos como cores, formas, tamanho e disposição dos componentes da interface. A padronização de elementos em tela faz com que o usuário reconheça, ao invés de aprender, e reduz a carga cognitiva deste (152). Já o aspecto sensorial se refere à interatividade do software e como os usuários percebem a experiência de usá-lo, incluindo elementos como a resposta tátil (em mobile e outros contextos cabíveis), a fluidez da interface, a previsibilidade da interação do usuário e mudanças de estado (115, 153).

Manutenção: Refere-se à facilidade com que um sistema de software pode ser alterado para corrigir falhas, melhorar desempenho ou funcionalidades, ou adaptar o sistema a um ambiente alterado ou em evolução (154). É um requisito abrangente que engloba a modularidade, a reutilização, a analisabilidade, a modificabilidade e a capacidade de teste. É importante para minimizar o esforço e o custo das alterações no

software ao longo de sua vida útil, que muitas vezes são significativamente maiores do que o custo inicial de desenvolvimento. Fatores que podem afetar a manutenção incluem a qualidade da documentação do software, a estrutura e a complexidade do código, a aderência a padrões de codificação e a utilização de tecnologias e plataformas amplamente suportadas (155).

Operabilidade: Este requisito pode ser definido como o grau em que um sistema ou produto possui atributos que o tornam fácil de operar e controlar (154), ou seja, a facilidade com que os usuários podem operar o sistema e o controle que têm sobre ele. A operabilidade eficiente é aquela que permite aos usuários realizar suas tarefas de maneira intuitiva e fácil, aumentando a eficiência e a produtividade. A operabilidade envolve considerações sobre a consistência da interface do usuário, a adequação dos comandos, a facilidade de uso das funções e o controle que os usuários têm sobre o sistema. Portanto, a operabilidade é um componente-chave na concepção de sistemas que são mais úteis e eficazes para os usuários finais (156, 157).

Desempenho: Bastante utilizado na maioria dos softwares, os requisitos de desempenho definem as expectativas para a velocidade, eficiência e eficácia da aplicação (158), tanto em software quanto em hardware (159). O desempenho de um sistema é avaliado em termos de taxa de transferência, tempo de resposta ou o número de tarefas concluídas em um período específico (158). Estes requisitos podem ser atendidos de diversas formas (160) e são fundamentais para assegurar que o sistema não só funcione como esperado, mas também que o faça de forma eficiente, mantendo uma experiência do usuário agradável mesmo sob condições de alta demanda ou de utilização intensiva de recursos (158).

Portabilidade: É a capacidade de um sistema de ser transferido eficientemente de um ambiente para outro, sendo projetado de forma a conseguir operar em diferentes ambientes de hardware e software sem exigir mudanças significativas. Isso inclui portabilidade de sistema operacional, portabilidade de hardware, portabilidade de dados e portabilidade de software (161). A portabilidade do software tem implicações significativas em termos de custos de desenvolvimento, suporte e manutenção e pode afetar a aceitação do sistema pelos usuários, dependendo do ambiente de destino. Em dispositivos portáteis utilizados em grande escala por muitos usuários, como os celulares, os serviços de portabilidade (normalmente associados à troca de aparelho físico com a manutenção do mesmo número de telefone) são comumente fornecidos por sistemas e bancos de dados centralizados que dependem de terceiros (162).

Escalabilidade: Refere-se à habilidade de um sistema se adaptar e gerenciar o aumento da demanda. Um sistema altamente escalável é capaz de lidar com o crescimento do volume de trabalho ou de usuários sem degradar o desempenho ou a funcionalidade. De acordo com Herbst et al. (163), a escalabilidade não é apenas importante para garantir a continuidade do serviço sob alta demanda, mas também para a otimização do uso de recursos e custos operacionais (163, 164), visto que um sistema que visa a escalabilidade pressupõe um melhor gerenciamento de recursos por parte do time responsável. Em metodologias ágeis, a elicitação de requisitos de escalabilidade costuma ser escassa, pelo desnível de conhecimento sobre escalabilidade em equipes de software e pelo fato de as entregas serem menores e contínuas (165, 166). Entretanto, estudos recentes estão sendo conduzidos para criação e implementação de modelos de elicitação

e documentação de requisitos de escalabilidade (167, 166).

Segurança: Uma das preocupações mais críticas no desenvolvimento de sistemas de software é a segurança, caracterizada pela proteção de informações e sistemas contra ataques e acessos não autorizados, garantindo a confidencialidade, integridade e disponibilidade de dados, além de estabelecer qual o grau de segurança desejado e a tolerância a falhas (168). São requisitos que atualizam-se constantemente em função do surgimento de novas tecnologias, como os novos serviços de 5G e 6G, recentemente (169). A confidencialidade e anonimização envolve a proteção de informações pessoais ou sensíveis, e é baseada em leis de proteção de dados como a GDPR na Europa (148) e LGPD no Brasil (149). Já o esforço em garantir a integridade se refere à salvaguarda da exatidão e consistência dos dados, e a disponibilidade se preocupa com a garantia de que o sistema e suas informações estejam sempre acessíveis quando necessário (170).

Além disso, os requisitos de segurança muitas vezes vão além da proteção de dados e estendem-se para abordar problemas de privacidade e conformidade com regulamentos, que são especialmente relevantes para sistemas que lidam com informações pessoais identificáveis (148, 149). A implementação eficaz de requisitos de segurança em um sistema de software implica uma abordagem multifacetada, atuando em várias camadas da aplicação (171, 172, 173). É importante notar que a segurança de software inicia-se desde o mapeamento do processo de negócio, quando procura-se primariamente evitar que erros ocorram (174).

Usabilidade: o termo “usabilidade” na engenharia de software se refere ao grau em que um produto ou sistema pode ser usado por usuários específicos para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso específico (175). Ela lida principalmente com a facilidade de aprendizagem e uso do sistema para novos usuários e a satisfação geral do usuário. Uma boa usabilidade pode aumentar a produtividade, reduzir erros e melhorar a satisfação geral, diminuindo a necessidade de o usuário utilizar documentação adicional, e pode ser avaliada antes mesmo de esperar feedbacks de usuários (176, 177, 178).

Além disso, o design de usabilidade também considera a acessibilidade, que é a facilidade com que diferentes usuários, incluindo aqueles com deficiências ou quaisquer dificuldades, podem usar o produto (179, 180). A acessibilidade é cada vez mais reconhecida como um aspecto essencial da usabilidade, especialmente em um mundo onde a tecnologia da informação desempenha um papel importante na vida cotidiana (181). As próprias bibliotecas mais recentes de website já vêm com componentes pré-definidos que atendem a vários requisitos de acessibilidade, como a responsividade em diferentes dispositivos. Em chatbots, a implementação de maior tolerância do tempo de resposta, flexibilidade de formulários e são algumas das questões relacionadas à acessibilidade que são passíveis de implementação (182). Entretanto, há discussões e propostas de melhoria quanto à eficácia dos padrões de acessibilidade para conteúdo na web - W3C's Web Content Accessibility Guidelines (WCAG), devido a condições especiais que não necessariamente são ligadas a deficiências, como dificuldades de aprendizado (181).

Aspectos Conversacionais: Requisitos conversacionais estão emergindo como um campo importante na engenharia de requisitos, especialmente com o advento de sistemas interativos baseados em linguagem,

como os chatbots (43). Essencialmente, os requisitos de conversação dizem respeito à eficácia com que um sistema interativo, como um chatbot, pode envolver os usuários em diálogos naturais semelhantes aos humanos (53, 85). Esses requisitos desempenham um papel fundamental na determinação de quão bem um sistema pode entender e responder às entradas do usuário, manter o contexto da conversa, gerenciar fluxos de diálogo e exibir comportamentos de conversa semelhantes aos humanos, principalmente para chatbots (183).

A utilização de inteligência artificial fez avanços significativos na melhoria das habilidades de chatbots (184), mas a complexidade da linguagem humana e a diversidade dos contextos de conversação continuam a apresentar desafios. Uma consideração importante na definição dos requisitos de conversação é a compreensão e a geração de respostas contextualmente apropriadas, que podem ser influenciadas por muitos fatores, como a intenção do usuário, o sentimento e o histórico da conversa (185).

À medida que o campo cresce, a própria reação dos usuários tende a moldar os requisitos emergentes. Um exemplo é a mudança entre abordagem humanizada e abordagem clássica de máquina a depender da situação. Por exemplo, para fins educativos, um chatbot humanizado é melhor aceito pelos usuários por fornecer um ambiente de aprendizagem personalizado e orientado a resultados (186). Entretanto, para situações que envolvem o compartilhamento de dados sensíveis, em um contexto financeiro, por exemplo, os usuários geralmente confiam mais em máquinas do que humanos, sugerindo uma abordagem mais tradicional para o chatbot (37).

Aspectos emocionais: Requisitos emocionais são cada vez mais relevantes no contexto de design de interação, e assim como os requisitos conversacionais, especialmente em chatbots, onde a experiência do usuário vai além da funcionalidade e usabilidade e abrange como o sistema faz com que o usuário se sinta (187), (188). Além da execução de tarefas, estes sistemas são projetados para reconhecer e responder a estados emocionais do usuário e suas variações, visando proporcionar uma experiência mais envolvente e satisfatória (189).

A modelagem de emoções em sistemas interativos é uma tarefa complexa que requer uma compreensão das teorias emocionais e seu mapeamento para fluxos conversacionais antropomórficos (190). Os sistemas devem ser capazes de interpretar sinais emocionais dos usuários (que podem ser expressos por meio de texto, voz, expressões faciais, etc.) e reagir de uma maneira equilibrada, que não seja extremamente autônoma de forma a excluir o usuário, nem totalmente dependente para não sobrecarregá-lo cognitivamente (191).

2.2.2 Documentação de requisitos

O termo "documentação de software" se refere ao conjunto de informações que descrevem as funcionalidades, a arquitetura e o uso de um sistema de software. Em abordagens tradicionais, toda a documentação ou a maior parte dela concentra-se em um único documento, enquanto em metodologias ágeis a documentação é formada por uma coletânea de artefatos menores, a qual facilita a leitura dos requisitos (192). Entretanto, neste processo alguns requisitos (principalmente não funcionais) podem ser subespecificados ou não documentados, e nestes casos, a gestão ineficiente das mudanças de requisitos pode resultar na falha

do projeto ou no desvio de cronograma e escopo (193).

De acordo com Habib et. al (194) existem seis (6) componentes de um processo de documentação de software que compõem uma documentação considerada leve e ao mesmo tempo suficiente, sendo eles:

- Documento de design: Descreve a operação do software, funcionamento, módulos e procedimento de desenvolvimento.
- Visão Geral do Projeto: Usado para identificar os resultados do software.
- Documento de requisitos - Elabora os requisitos pelo cliente e desenvolvedor. Refere-se à etapa de especificação.
- O documento de suporte - Útil no processo de desenvolvimento dos módulos do software e sua sincronização.
- Documento de operação - Esclarece a instalação, armazenamento e outros aspectos gerais perspectivas do software.
- Documentação do produto: tipo de documentação referente ao tempo e custo estimado do software.
- A documentação do sistema é a documentação geral do software com base em contrato como instalação e treinamento, e o modelo de contrato é um documento de software e acordo entre o cliente e o desenvolvedor (194). Os artefatos produzidos durante o desenvolvimento podem ou não ter mais de um dono, e geralmente adotam controle de versão para comportar as mudanças que surgem ao longo do projeto (195).

Em geral, a forma dos documentos de requisitos é afetada pelo contexto, o qual envolve a categoria do produto, tamanho e a experiência dos desenvolvedores e stakeholders (193). No entanto, todo documento deve seguir uma estrutura que facilite a conversa entre o time de desenvolvimento e os stakeholders, já que a documentação estabelece um “acordo” acerca do software pretendido. Primeiramente, o documento de requisitos do sistema deve ser composto por sentenças em linguagem natural, apresentando e exemplificando o que o sistema deve fazer; Em seguida, os requisitos devem estar organizados logicamente, seja por módulo, seja por ordem cronológica (ex: entrada, processamento e saída). É importante que também estejam classificados em requisitos funcionais e não funcionais. Além disso, cada requisito deve ter um identificador único, para posterior referência. Entretanto, é comum utilizar documentação incompleta ou obsoleta, mas ainda assim eficiente, sugerindo um tradeoff entre o escopo e nível de detalhes da documentação, e seu custo de elaboração e manutenção (196).

Em sistemas não tradicionais, ou seja, que seguem metodologias ágeis, é comum que exista um débito documental, ou seja, a não criação ou criação incompleta de documentação do software (28). Adicionalmente, não há um consenso quanto a requisitos emergentes, principalmente para softwares de inteligência artificial, como chatbots (197). Embora os chatbots de IA tenham a conversação como denominador comum, podem tornar-se bastante especializados de acordo com sua base de treinamento (125). Usualmente, a maioria das documentações são pensadas de requisitos para funcionalidades, e portanto ainda existe uma lacuna em relação a boas práticas para a documentação de requisitos para chatbots, especialmente aqueles relacionados ao uso e tratamento dos dados de treinamento do modelo, e ao fluxo de conversação (43).

2.3 TRABALHOS CORRELATOS

Tizard et al. (197) exploraram a extração e classificação de requisitos em diferentes fóruns online, que assim como as redes sociais, são uma robusta fonte de feedback dos usuários finais em relação aos produtos. Entretanto, as redes sociais muitas vezes têm limite de caracteres para a escrita, enquanto nos fóruns o contexto e a situação costumam ser melhor descritas, o que auxilia na coleta de requisitos. Os autores encontraram várias evidências de requisitos relacionados a desempenho, disponibilidade e usabilidade, bem como erros diversos, a dimensão (em escala) de certos problemas na visão dos usuários (pelo número de reclamações e confirmações de outros usuários) e solicitações de recursos adicionais desejados (198).

Em relação à documentação, Castle-Green et al. (199) destacaram o formato de árvores de decisão. Este destaque advém de trabalhos anteriores, como a criação de fluxos de trabalho ou histórias para mapear as etapas de interação, a inclusão de mapeamento de fluxo nas diretrizes do Google e a utilização do termo “árvores de decisão” pela Amazon como referência a rotas mais curtas e lógica do sistema. O artigo aponta que existem 3 “partes móveis” deste tipo de documentação, que modificam e são modificadas pela árvore de decisão lógica: Plataformas comoditizadas, que se referem à inserção do chatbot em plataformas já existentes; times de design e desenvolvimento, que incorporam a árvore de decisão ao seu trabalho; e a experiência do usuário, que pode ser afinada e restritiva ou mais flexível a depender do objetivo e método de desenvolvimento do chatbot (199).

No ano posterior, Borsting e Hesenius (200) propuseram uma adaptação do método “Quarto de interação” (Interaction Room) para o levantamento e documentação de requisitos em chatbots. O método original consiste em reunir membros de um mesmo projeto porém de diferentes áreas para encontrar soluções em conjunto, que são escritas pelo moderador no momento da interação e depois transferidas para documentos que são posteriormente compartilhados com todos os participantes. Nesta adaptação, os artefatos produzidos contém requisitos das propriedades consideradas relevantes neste cenário - declarações, intenções e entidades - e são divididos em Canvas (telas) de intenção, diálogo e entidades (200).

Em seguida, Silva e Canedo (43) apontaram que a maior dificuldade na elicitação de requisitos é que o produto supra as necessidades dos stakeholders, e que este é um desafio ainda maior em chatbots devido aos requisitos emergentes associados a estes - proatividade, consciência, comunicabilidade e inteligência emocional. O estudo também aponta que, no geral, as técnicas utilizadas para levantamento de requisitos são as mesmas utilizadas para outros softwares, entretanto, existem várias diferenças de priorização de requisitos em diferentes times de desenvolvimento, e portanto, a adaptabilidade de requisitos precisa ser levada em consideração, principalmente em chatbots, onde características almejadas por stakeholders são subjetivas, difíceis de testar e validar (43).

Embora cada uma dessas abordagens contribua para a nossa compreensão sobre a documentação de requisitos para chatbots, nenhuma delas fornece um modelo que contemple as particularidades de requisitos subjetivos durante os ciclos de elicitação e especificação de requisitos. Neste sentido, esta pesquisa busca sintetizar essas abordagens distintas em um único modelo de especificação de requisitos conversacionais para chatbots, abrangendo em detalhes os requisitos considerados “emergentes” e até então específicos deste tipo de software: conversacionais e emocionais.

3 REVISÃO SISTEMÁTICA - ESPECIFICAÇÃO E DOCUMENTAÇÃO DE REQUISITOS

Este capítulo apresenta uma Revisão Sistemática Terciária da Literatura (RSTL) seguindo o protocolo de Kitchenham e Charters (42) para investigar os produtos da documentação de requisitos de software, de forma a responder a QP. 2 deste estudo: "*Como os requisitos de software são documentados pelos profissionais da engenharia de requisitos de forma a fornecer uma compreensão comum sobre o software e permitir a colaboração de diferentes atores?*". A RSTL é um tipo de estudo terciário que consiste em identificar, avaliar e interpretar todas as evidências relacionadas a estudos relevantes de uma área ou questão de pesquisa em foco, e foi escolhida pela necessidade de abranger a literatura já existente sobre o tema, e é composta pelas seguintes etapas (42):

- **1. Planejamento:** A primeira etapa visa identificar a necessidade de uma revisão pela observação de lacunas na literatura, estabelecendo objetivos e definindo o protocolo de revisão, que consiste em perguntas de pesquisa, palavras-chave, critérios de seleção do estudo, lista de dados a serem extraídos e avaliação de qualidade;
- **2. Condução:** Após o planejamento, o protocolo de revisão é iniciado, utilizando o artefatos produzidos na fase anterior para filtrar estudos e extrair informações;
- **3. Relatório:** Por fim, os resultados obtidos após a condução são descritos para gerar o artefato resultante, que, neste caso, é um trabalho de pesquisa.

A realização de uma revisão sistemática da literatura fornece uma compreensão abrangente e atualizada do estado da arte em uma determinada área de pesquisa, e auxilia na identificação de lacunas do conhecimento, avaliação e síntese de evidências, identificação de abordagens e metodologias utilizadas, fornecimento de embasamento teórico e na contribuição para o avanço do conhecimento, ao agregar e integrar os resultados de diferentes estudos, identificar direções futuras de pesquisa e propor novas abordagens ou perspectivas (201).

3.1 QUESTÕES DE PESQUISA

A motivação para conduzir uma Revisão Sistemática da Literatura (SLR) reside na síntese de técnicas, abordagens e artefatos utilizados na documentação de requisitos, e quais são as características diferenciais entre eles. Neste capítulo é abordada a questão de pesquisa 2 desta dissertação - "*QP.2. Como os requisitos de software são documentados pelos profissionais da engenharia de requisitos de forma a fornecer uma compreensão comum sobre o software e permitir a colaboração de diferentes atores?*". A questão de pesquisa é direcionada aos requisitos de software elicitados pelos analistas e engenheiros de requisitos,

QP	Questão de Pesquisa
QP.2.1	Quais são os artefatos utilizados na literatura para a documentação de requisitos em linguagem natural?
QP.2.2	Como diferentes artefatos de documentação são utilizados para abordar requisitos funcionais e não funcionais em projetos de software?
QP.2.3	Quais são as diretrizes, estruturas ou formatos utilizados nos artefatos de documentação dos requisitos em linguagem natural?

Tabela 3.1: Questões de pesquisa

com o envolvimento de stakeholders durante o processo (202). Esta questão foi dividida em subquestões, que são apresentadas na Tabela 3.1.

As questões de pesquisa buscam evidenciar os aspectos mais relevantes dos modelos de documentação de requisitos existentes, Sendo assim, a **QP.2.1** visa sumarizar os artefatos existentes para documentação de especificações. Já a **QP.2.2** busca elucidar a diferença de representação dos diferentes requisitos nos artefatos existentes. Por fim, entende-se que com o decorrer das práticas de engenharia de software, padrões foram propostos e/ou adotados e normas foram criadas para o registro de requisitos elicitados ao longo do desenvolvimento. Assim, a **QP.2.3** investiga quais são as diretrizes, estruturas ou formatos adotados nos artefatos de documentação.

3.2 SEQUÊNCIA TEXTUAL DE BUSCA

O *Framework* escolhido para a criação da sequência textual de busca foi o PICOC (em português: População, Intervenção, Comparação, Resultado, Contexto) (203). Neste *Framework*, a população se refere ao objeto de estudo; a intervenção é o meio utilizado pela população; a comparação é o que está sendo comparado com a intervenção; o resultado é o desfecho da intervenção; e o contexto é o foco do estudo, suas restrições e limitações. A Tabela 3.2 mostra a definição final dos termos PICOC para construir a cadeia de pesquisa genérica. A comparação não é aplicável porque a intervenção não é comparada.

Após algumas etapas de aplicação, a sequência final de busca foi decidida. A sequência textual de busca foi traduzida e aplicada em inglês, visto que a maior parte da literatura encontra-se nesta língua. As bases de dados digitais escolhidas para aplicar a sequência de palavras foram baseadas em sua relevância na pesquisa em engenharia de software (204), na abrangência na indexação de conferências e jornais e na possibilidade de aplicar a sequência de busca genérica diretamente no campo de pesquisa.

PICOC	Palavras-chave	Palavras relacionadas
População	Requisitos	
Intervenção	Especificação	Documentação, linguagem natural.
Comparação	Não aplicável	Não aplicável, pois a comparação não é necessária neste contexto.
Resultado	Artefato	Modelo, guia, formato.
Contexto	Revisão Sistemática da Literatura	revisão, "revisão sistemática", revisão, "mapeamento sistemático"

Tabela 3.2: Termos PICOC traduzidos em português para construção de sequência de busca.

Os termos foram utilizados em inglês para efetuar a busca de fato.

Sequência de busca:

("Requirements") AND ("specification"OR "documentation"OR "natural language") AND ("artifact"OR "model"OR "guidelines"OR "format") AND ("systematic review"OR "review"OR "systematic mapping").

Foram utilizadas 5 bases de dados para a pesquisa, seguindo a metodologia de Kitchenham, 2010 (205): ACM, IEEEExplore, SCOPUS, Springer Link e Web of Science. As bases CiteSeer e Google Scholar não foram incluídas no estudo por não fornecerem filtros suficientes que garantissem a reprodutibilidade. A sequência de busca inicial foi adaptada para cada uma das fontes e a pesquisa foi feita pelo título, palavras-chave e resumo (abstract) dos artigos.

Entretanto, a base de indexação ACM não disponibiliza a opção de busca pelos 3 de uma só vez, motivo pelo qual cada um dos 3 campos teve de ser pesquisado separadamente (Tabela 3.3). Alguns dos critérios de exclusão foram aplicados nos próprios filtros adicionais da consulta, como sugerido por Costal et al. (206). Por exemplo, na ACM, Springer Link e Web of Science, foi aplicado um filtro manual para “Tipo de conteúdo”, selecionando apenas “Artigos de Revisão” e adicionando um intervalo de data desde 2013 até o momento da pesquisa (novembro de 2023). Na base IEEEExplore, a aplicação do intervalo temporal foi manual. Na base SCOPUS, todos os filtros foram adicionados na própria sequência, como mostrado na tabela 3.3. As bases Web of Science e SCOPUS possuem a opção de pesquisa por "tópico", o qual abrange título, abstract e palavras-chave, portanto esta opção foi utilizada. Também fornecem os recursos de filtro por tipo de documento, ano de publicação e áreas de pesquisa, os quais foram utilizados respectivamente para selecionar apenas artigos de revisão publicados de 2013 a 2023, e que fossem da engenharia de software ou áreas de pesquisa correlatas (excluindo assim artigos de enfermagem, farmacologia e afins).

**Base de in- Sequência Textual
dexação**

ACM Digital Library	Title:(specification OR documentation OR "natural language") AND Title:(artifact OR model OR guideline OR format) AND Title:(review OR "systematic review"OR "systematic mapping") AND Abstract:(Requirements) AND Abstract:(specification OR documentation OR "natural language") AND Abstract:(artifact OR model OR guideline OR format) AND Abstract:((review OR "systematic review"OR "systematic mapping")) "filter": E-Publication Date: (01/01/2013 TO 12/31/2023),ACM Content: DL
IEEE Xplore	("Requirements") AND ("specification"OR "documentation"OR "natural language") AND ("artifact"OR "model"OR "guidelines"OR "format") AND ("systematic review"OR "review"OR "systematic mapping")
Scopus	(TITLE-ABS-KEY (requirements) AND (TITLE-ABS-KEY (specification OR documentation OR "natural language"))) AND (TITLE-ABS-KEY (artifact OR model OR guideline OR format)) AND TITLE-ABS-KEY (review OR "systematic review"OR "systematic mapping") AND DOCTYPE(re) AND PUBYEAR > 2013 AND PUBYEAR < 2023)
Springer Link*	("Requirements") AND ("specification"OR "documentation"OR "natural language") AND ("artifact"OR "model"OR "guideline"OR "format") AND ("review"OR "systematic review"OR "systematic mapping")
Web of Science	(TS="Requirements") AND (TS="specification"OR TS="documentation"OR TS="natural language") AND (TS="artifact"OR TS="model"OR TS="guideline"OR TS="format") AND (TS="review"OR TS="systematic review"OR TS="systematic mapping")

Tabela 3.3: Adaptação de sequência textual às diferentes bases bibliográficas.

3.3 CRITÉRIOS DE SELEÇÃO

3.3.1 Critérios de inclusão

Seguindo a proposta de Kitchenham (201), foram definidos critérios de inclusão e exclusão. Os critérios de inclusão foram baseados no tema principal dos artigos, em consonância com o tema do trabalho e cujo trabalho tenha gerado artefatos. Os critérios de inclusão são:

- (IC 1) Os artigos devem abordar artefatos, modelos, guias ou formatos relacionadas à documentação de requisitos em linguagem natural.
- (IC 2) Os artigos devem apresentar pelo menos um tipo de documentação de requisitos.

- (IC 3) Os artigos devem apresentar informações sobre artefatos de documentação existentes e suas aplicações para ilustrar requisitos funcionais e não funcionais.

3.3.2 Critérios de exclusão

Os critérios de exclusão têm como objetivo excluir trabalhos que atendam aos critérios de inclusão, mas não apresentem alguma metodologia, foco, aspecto ou abordagem. Os critérios de exclusão (EC) são mostrados abaixo:

- (EC 1) Artigos que não estejam relacionados exclusivamente à documentação de requisitos ou não abordem diretamente as questões de pesquisa mencionadas;
- (EC 2) Artigos que não sejam escritos em um idioma diferente do compreendido pelos autores (português, espanhol e inglês);
- (EC 3) Artigos que não são uma revisão de literatura (por exemplo, capítulos de livro, artigos de pesquisa, estudos de caso, dissertações, teses, trabalhos em andamento, documentos de posição, trabalhos duplicados);
- (EC 4) Publicado antes de 2013.
- (EC 5) O foco do artigo não é a análise dos artefatos de documentação em linguagem natural, mas outros procedimentos (i.e: automação, classificação com NLP, aplicação de machine learning, técnicas de elicitação, etc).
- (EC 6) O assunto principal do artigo não está relacionado à engenharia de software ou áreas correlatas, mas outras áreas de pesquisa;

3.4 AVALIAÇÃO DE QUALIDADE

Apesar do filtro prévio já realizado pelos critérios de exclusão, ainda é necessário verificar a qualidade dos artigos selecionados, para garantir que as revisões são válidas e foram feitas de acordo com práticas neste trabalho. Para isso, foi utilizada a seguinte lista de verificação:

- (Q1) As questões de pesquisa foram claramente definidas?
- (Q2) A metodologia é clara, adequada e replicável para a busca, seleção e análise dos estudos incluídos?
- (Q3) As limitações da revisão foram evidenciadas e explicadas?
- (Q4) Os artigos consideram a aplicabilidade prática dos resultados e suas implicações para a área de pesquisa, fornecendo recomendações úteis e relevantes?

- (Q5) Os artigos trazem contribuição original ao conhecimento da área, seja por identificar lacunas na literatura, sintetizar resultados conflitantes ou propor novas perspectivas ou direções de pesquisa?

Cada artigo pré-selecionado foi avaliado em relação às 5 questões, e somente os artigos que atendiam a todos os critérios propostos foram incluídos na análise.

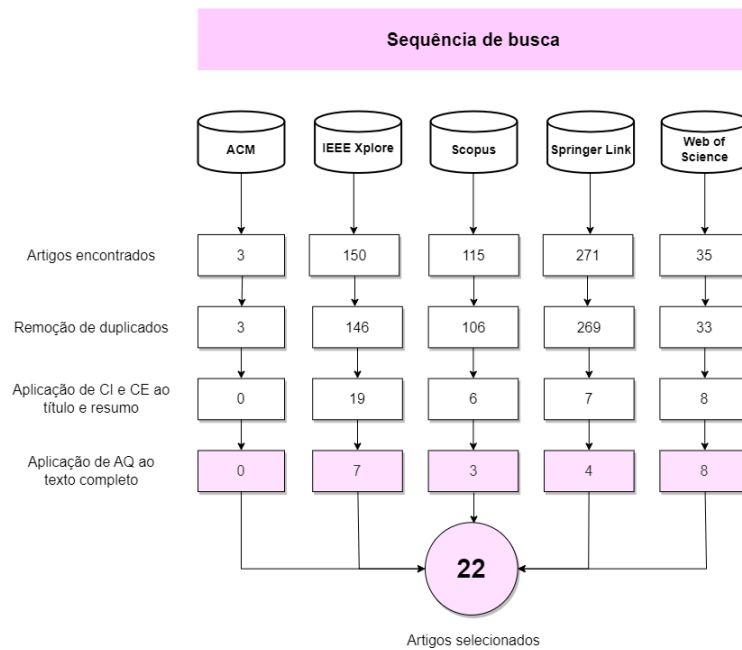
3.5 EXTRAÇÃO DE DADOS

Para responder às questões propostas, foram extraídos dados referentes à publicação (título, autores, ano de publicação, resumo, DOI, palavras-chave e tipo de revisão (revisão de literatura, revisão sistemática, meta-revisão ou revisão terciária). Além disso, foram extraídos dados referentes às questões de pesquisa em si, como artefatos de linguagem natural abordados em cada artigo, a categoria de requisitos abordados por eles (requisitos funcionais e não funcionais), contexto de uso do artefato, e recomendações ou observações do artigo, quando houvesse.

3.6 CONDUÇÃO DA PESQUISA

Para a realização desta revisão, foi utilizada a ferramenta Parsifal, uma plataforma web gratuita e de código aberto destinada ao apoio à Revisão Sistemática da Literatura (SLR). A escolha da plataforma deu-se devido à adequação de suas funcionalidades e estrutura de trabalho ao processo de SLR adotado neste estudo, conforme proposto por Kitchenham e Charters (207). Essa ferramenta facilita a revisão, abordando todas as fases da revisão sistemática (planejamento, condução e relatório), permitindo uma navegação ágil e eficiente pelos títulos e resumos durante a fase de filtragem, além da identificação automática de artigos duplicados. A Figura 3.1 ilustra os artigos que permaneceram após cada etapa de condução.

O período de coleta dos estudos através da string de busca foi até dezembro de 2023, resultando em um total de 574 artigos (3 da ACM, 150 da IEEEExplore, 35 da Web of Science, 115 do SCOPUS e 271 da Springer Link), conforme visto na Figura 3.1. Foram removidos 17 estudos duplicados, restando 557 artigos (3 da ACM, 143 da IEEEExplore, 33 da Web of Science, 106 do SCOPUS e 269 da Springer Link) para aplicar os critérios de inclusão e exclusão, lendo o título e o resumo. Embora os artigos analisados pelo título e resumo abordassem o tema de requisitos, 517 foram removidos devido à não observância de um ou mais critérios. Por último, foi realizada a leitura do texto completo para uma análise mais aprofundada e aplicação da avaliação de qualidade de 40 artigos (zero da ACM, 19 da IEEEExplore, 8 da Web of Science, 6 do SCOPUS e 7 da Springer Link), o que resultou na remoção de 17 trabalhos e no número final de 22 artigos de revisão selecionados (zero da ACM, 7 da IEEE, 8 da Web of Science, 3 do SCOPUS e 4 da Springer Link).



CI = critério de inclusão, CE = critério de exclusão, Q = avaliação de qualidade.

Figura 3.1: Artigos restantes após cada etapa da SLR.

3.7 RESULTADOS DA SLR

Esta seção apresenta uma análise detalhada dos resultados obtidos a partir da extração de dados efetuada na Revisão Sistemática da Literatura (SLR), visando o esclarecimento das questões de pesquisa propostas. Adicionalmente, esta seção propõe uma discussão sobre as práticas e implicações associadas à implementação de métodos de documentação de requisitos. Esta discussão é fundamentada no somatório dos resultados dos estudos selecionados, especialmente considerando aqueles que exploram aspectos similares ou correlatos à documentação de requisitos.

3.7.1 QP.1 - Quais são os artefatos utilizados na literatura para a documentação de requisitos em linguagem natural?

Nos artigos selecionados, foram identificados 12 tipos de artefatos utilizados na documentação de requisitos e presentes em mais de um artigo, como apresentado na Tabela 3.4 . Dentre estes, os *UML diagrams*, *use cases* (casos de uso), *scenarios* (cenários), *conceptual models* (modelos conceituais), *user stories* (histórias de usuário) e *prototypes* (protótipos) foram, respectivamente, os artefatos mais recorrentes 3.2.

Em seguida, as *ontologies* (ontologias), *mind maps* (mapas mentais), *storyboard* (quadro de histórias), *personas*, *system components* (componentes de sistema), *kanban boards* (quadros kanban), *storycards* (cartões de histórias), *tasks* (tarefas) e *visions* (visões) foram citados em mais de um artigo. Apesar de não serem tão citados quanto os seis primeiros, podem indicar uma documentação diferenciada em determinados contextos.

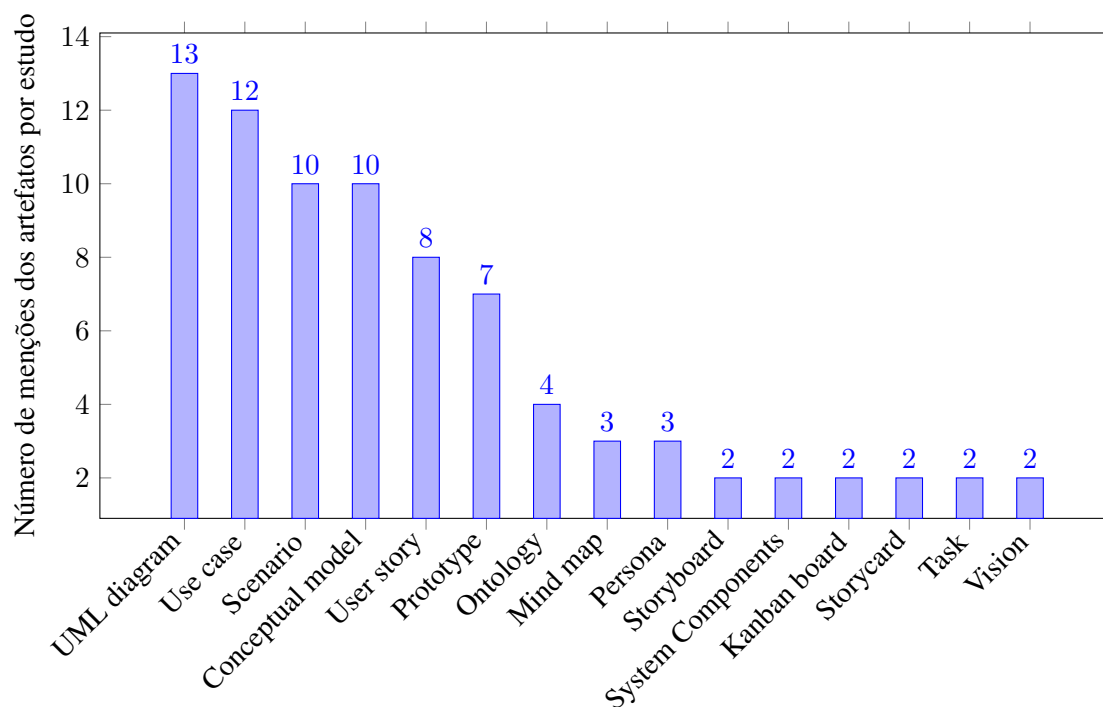


Figura 3.2: Número de menções de cada artefato de documentação nos artigos selecionados. O gráfico só considera artefatos de linguagem natural mencionados mais de uma vez.

Dentre os artefatos citados apenas uma vez, estão as tabelas, especificação formal, relatórios em HTML, figuras, tarefas do quadro de histórias (storyboard tasks), vídeos, experiências de usuário documentadas, parede, pin board, Cadeia de processos orientada a eventos (Event-driven Process Chain - EPC), modelos de domínio, tags, lista de desejos do usuário, jornada do usuário, backlog de produto, *roadmap*, definição de "pronto", gráfico de *burndown* de sprint e termos genéricos como "requisito de produto", "requisito de mercado" e "requisito técnico".

Seguindo a proposta de Schon et. al (2017), serão destacados os artefatos que estão presentes em pelo menos 20% dos artigos (202).

Os diagramas são representações visuais organizadas, utilizadas para representar diversas situações e relacionamentos em nível sequencial, hierárquico ou estrutural (208). A linguagem de modelagem unificada (Unified Modelling Language - UML) permite a criação de diversas instâncias, como diagramas de classe, diagramas casos de uso, diagramas de atividade e diagramas de sequência (208). A notação BPMN, por sua vez, é considerada o padrão na modelagem de processos de negócio (209). Em desenvolvimento ágil, o diagrama de caso de uso foi considerado um complemento importante na documentação de requisitos, uma vez que a documentação em somente um formato (ex: histórias de usuário) mostrou-se insuficiente para o contexto (210). A presença de uma linguagem unificada (UML) e a possibilidade de customização até ao criar extensões novas específicas para determinado contexto (211) - como segurança em IoT (212) - caracterizam-se como os motivos pelos quais este artefato foi o mais citado na literatura encontrada (59,1%).

Logo em seguida, os casos de uso foram quase tão mencionados quanto os diagramas (54,5%). Casos de uso são bastante utilizados na documentação de requisitos de software e são adaptáveis, possibilitando

a inclusão de vários fluxos de comportamento (213). Apesar do risco de apresentarem muita informação e do esforço envolvido ao descrever muitos cenários (214), são considerados artefatos mais técnicos que as histórias de usuário (202), com possibilidade de reaproveitamento do requisito documentado para a geração de manuais e estimativa de esforço (215). Além disso, os casos de uso permitem a identificação precoce de requisitos contraditórios ou incompletos, e podem ser evoluídos, e modificados durante o ciclo de vida do software (213).

Na literatura analisada, os cenários e modelos conceituais foram mencionados em 45,4% dos estudos. Os cenários descrevem interações dinâmicas e contexto-dependentes entre o sistema e atores externos: usuário, rede, outros dispositivos. São bastante significativos para verificar se todos os casos relevantes estão cobertos ou não (216). Além disso, facilitam a automação de testes, impedindo que o desenvolvedor precise interpretar novamente os requisitos e critérios de aceite (216). São frequentemente combinados com outros artefatos, como personas (202), histórias de usuário (216) ou diagramas (217).

Já os modelos conceituais são utilizados como ponte entre stakeholders ou usuários e time técnico, e proporcionam a primeira transição entre requisitos textuais para modelos formais (218). É importante destacar que modelos conceituais são diagramas, porém com nível de abstração maior, e muitas vezes voltado para o negócio, ao invés do foco no funcionamento do software, embora a notação UML também seja a mais utilizada entre profissionais que criam estes artefatos (219).

As histórias de usuário aparecem em 36,3 % dos artigos avaliados. Entretanto, são descritas como o artefato mais utilizado para desenvolvimento ágil de software em um dos artigos (202), mas não em outro (220). As histórias de usuário são artefatos que documentam as necessidades do ponto de vista dos usuários, escritas em linguagem natural e geralmente seguindo um formato semiestruturado (221), como "Eu [tipo de usuário], eu quero [algo] para [objetivo]"(216). Assim como os diagramas, permitem adição e customização de elementos (como critérios de aceite e inclusão de cenários), tornando-a um artefato popular na documentação de requisitos (216).

Em contrapartida aos requisitos descritos em forma textual, os protótipos também são bastante utilizados na documentação de requisitos (31,8%), principalmente relacionados aos aspectos visuais. Entretanto, existem definições diferentes de protótipo na literatura (202), abrangendo tanto desenhos informais em papel (baixa fidelidade) quanto wireframes (média a alta fidelidade) e também mockups interativos em HTML (215). Protótipos de menor fidelidade foram recomendados para documentação quando o propósito é a comunicação entre stakeholders e times ágeis (222), bem como discussões com usuários (223). Já os protótipos interativos foram utilizados tanto para a especificação de requisitos de navegabilidade e comerciais (224) quanto para suporte na elicitação de casos de teste (215).

Os artefatos mais citados encontrados neste trabalho corroboram parcialmente com o trabalho anterior de Schon et. al (202). De fato, histórias de usuário, protótipos, casos de uso e cenários permanecem como artefatos mais abordados na literatura. Entretanto, diagramas e modelos conceituais não apresentaram predominância tão forte quanto a nossa (apenas 11%). Em parte, isso pode ser explicado pelo ano de publicação dos trabalhos, uma vez que a maioria das revisões encontradas em nossa abordagem que mencionavam diagramas são posteriores a 2017 (225, 226, 227, 228, 218, 229, 230, 210, 231, 232), ano de publicação do trabalho anterior (202). Outro possível ponto de divergência é em relação ao escopo, uma vez que o trabalho de Schon et. al (202) realiza um filtro explícito para documentações ágeis, e o nosso

trabalho não possui essa segmentação.

Em relação aos artefatos menos citados, é interessante destacar alguns pontos contraditórios. Primeiramente, as tarefas, quadro de histórias e cartões de história são mais utilizados por times ágeis durante a distribuição de tarefas e execução de cronograma, em detrimento de documentação de requisitos. Entretanto, os artefatos foram incluídos pois nos estudos avaliados, foram utilizados ativamente na documentação. O quadro kanban, normalmente utilizado para divisão e visualização de tarefas, foi apresentado como forma de registro e rastreamento de requisitos (202), bem como as tarefas, quadros de história e cartões de história (202). Alguns artefatos também foram mencionados como complementares para artefatos "principais", como personas (202, 225) e cartões de história (214). Como levantado por Schon et. al (202), é importante destacar que a literatura pode não refletir a totalidade dos artefatos mais utilizados, uma vez que alguns artefatos clássicos (como backlog de produto) são raramente mencionados em estudos (202).

3.7.2 QP.2 - Como diferentes artefatos de documentação são utilizados para abordar requisitos funcionais e não funcionais em projetos de software?

Os artefatos de documentação em desenvolvimento de software possuem características que os tornam mais adequados para registrar e rastrear determinados tipos de requisitos, abordados previamente no capítulo 2. Desta forma, a tabela 3.4 mostra quais tipos de requisitos são ilustrados pelos artefatos encontrados nesta revisão terciária, de acordo com os artigos selecionados.

Artefatos	Descrição	Tipos de requisitos abarcados	Referências
<i>UML Diagram</i>	Representação visual de interações, relacionamentos, arquiteturas e sequências.	-	13: (225, 233, 226, 227, 228, 202, 218, 229, 230, 210, 231, 217, 232).
<i>Use case</i>	Descrição detalhada das funcionalidades do sistema sob a perspectiva do usuário final, incluindo cenários e exceções.	FRs e NFRs.	12: (214, 216, 227, 228, 202, 218, 229, 210, 231, 217, 213, 220).
<i>Scenario</i>	Narrativa que ilustra como um sistema é usado em um contexto específico, destacando funcionalidades.	FRs e NFRs.	11: (225, 214, 216, 234, 210, 231, 217, 213, 232, 202, 230).
<i>Conceptual model</i>	Representação abstrata que define a estrutura e o comportamento de um sistema em alto nível	FRs e NFRs.	10: (226, 225, 218, 235, 202, 229, 230, 210, 220, 228).

Continua na próxima página

Tabela 3.4 – continuação da página anterior

Artefatos	Descrição	Tipos de requisitos abarcados	Referências
<i>User story</i>	Descrição breve e informal de uma funcionalidade do software do ponto de vista do usuário, incluindo critérios de aceite.	Recomendada para FRs, pode ser usada para NFRs.	8: (221, 202, 225, 214, 236, 216, 210, 220).
<i>Prototype</i>	Modelo interativo da interface do usuário, usados para testes e feedbacks sobre o design e a usabilidade.	FR, NFR.	7: (221, 225, 236, 202, 217, 220, 214)
<i>Ontology</i>	Modelo formal que representa o conjunto de conceitos dentro de um domínio, e as relações entre estes	FRs e NFRs.	4: (225, 216, 230, 232).
<i>Mind map</i>	Diagrama usado para organizar visualmente informações ao redor de um conceito central.	FR e NFR.	3: (236, 202, 229).
<i>Storycard</i>	Além da descrição e critérios de aceite, o cartão de história pode conter estimativas de esforço/tempo, prioridade, e anotações sobre discussões ou decisões relevantes.	Recomendada para FRs, pode ser usada para NFRs.	3: (214, 202, 220).
<i>Storyboard</i>	Sequência de desenhos ou imagens que delineiam uma série de ações ou interações do usuário com o sistema	FRs.	2: (202, 221).
Persona	Personagem fictício criado para representar um tipo de usuário	NFR.	3: (202, 225, 220).
<i>System components</i>	Representação de elementos, atores ou módulos individuais que compõem um sistema de software	NFR.	2: (214, 228).
<i>Kanban board</i>	Ferramenta visual para gerenciar e rastrear o progresso de itens de trabalho.	Não relatado.	2: (202, 237).
<i>Tasks</i>	Subdivisão da história de usuário com característica mais técnica.	FR e NFR.	2: (202, 220).

Continua na próxima página

Tabela 3.4 – continuação da página anterior

Artefatos	Descrição	Tipos de requisitos abarcados	Referências
<i>Vision</i>	Descrição abstrata do produto, comum a todos os envolvidos.	FR e NFR.	1: (202, 220).

Tabela 3.4: Artefatos e tipos de documentação encontrados. FR = requisito funcional; NFR: requisito não-funcional.

De forma geral, foi observado que artefatos com características similares tendem a serem utilizados de forma similar na abordagem de requisitos funcionais e não funcionais 3.4. Desta forma, dividimos os artefatos em 3 categorias em relação à predominância de elementos multimídia que apresentam: formato misto, predominantemente textuais e predominantemente visuais. De forma geral, artefatos mistos apresentam pequenos textos, elementos gráficos e/ou imagens e possuem tendência a demonstrar sequências de eventos ou hierarquias com maior nível de abstração. Artefatos predominantemente textuais são compostos de textos estruturados, semiestruturados ou não-estruturados, escritos em linguagem natural. Por último, artefatos predominantemente visuais geralmente estão associados à previsão de interfaces e comportamentos que serão visualizados por um ou mais grupos de usuários.

Artefatos mistos:

Pelo seu maior nível de abstração, os artefatos mistos conseguem abordar tanto requisitos funcionais quanto não funcionais. Entretanto, modelos abstratos podem ter dois papéis conflitantes: devem ser generalistas o suficiente para facilitar a comunicação entre diversos atores com conhecimentos variados, mas também possuir formalidade o suficiente para conter toda a informação necessária (218).

Mesmo assim, os diagramas, modelos conceituais e mapas mentais podem ser utilizados para documentar requisitos funcionais e não funcionais (234, 210). Mornie et. al (210) elenca o uso de diagramas de classe e sequência como facilitadores para a tomada de decisão do time de software em relação à melhor forma de implementar os requisitos funcionais solicitados por stakeholders (210). Nguyen et. al (234) reúne exemplos de diferentes tipos de diagramas para a modelagem de aspectos estáticos e dinâmicos em um sistema, com foco em requisitos não-funcionais de segurança (234). Entretanto, afirma que muitos dos trabalhos que utilizam diagramas para elicitação de requisitos de segurança focam em apenas alguns aspectos da segurança mais próximos do usuário (autorização, autenticação), enquanto o escopo de requisitos de segurança é amplo: confidencialidade, integridade, disponibilidade, dentre outros (234).

Em contraponto, Mustafa et. al (231) explicita que diagramas UML são extensivamente utilizados para abordar requisitos funcionais, mas não são adequados para se utilizar como material de entrada para a geração de casos de testes não-funcionais. Não obstante, a criação de frameworks e notações específicas, como os framework NFR (238) e Tropos (239) permitem a representação de requisitos não-funcionais por modelos conceituais.

Artefatos predominantemente textuais:

Artefatos predominantemente textuais podem ser classificados como aqueles que se utilizam majoritariamente da escrita em linguagem natural para registro e documentação de requisitos de software. São bastante variáveis entre si, principalmente com relação ao nível de detalhes e tipos de informação apresentadas, e por isso a adequação para registro de requisitos funcionais ou não-funcionais depende do artefato em si.

Os cenários, por exemplo, são descrições curtas de interação dependentes de um contexto. Podem ser descritos de forma textual (216), como cenários alternativos ou principais em casos de uso (213) ou representados como diagramas (234). São comumente utilizados para ilustrar e registrar requisitos funcionais levantados durante a etapa de elicitação, porém existem trabalhos na literatura de utilização de cenários para representar requisitos de usabilidade (223), segurança ().

Em relação às histórias de Usuário, há um certo consenso de que servem para registrar requisitos funcionais do ponto de vista do usuário, e são incompletas ou inadequadas para descrever aspectos técnicos ou muito complexos (214, 202, 210, 221). Apesar disso, são artefatos consideravelmente utilizados na documentação de requisitos de software. A literatura selecionada sugere a extensão do formato da história de usuário para a inclusão de mais detalhes, hierarquia e rastreamento, bem como a combinação de HUs com documentações adicionais de requisitos (236, 210, 225). O surgimento de épicos, tarefas, cartões de história e quadros de história derivam da necessidade de segmentação, rastreamento, e/ou inclusão de outras informações - cartões de história, por exemplo, permitem capturar estimativa de tempo, responsável pelo desenvolvimento e metas, priorização, e até mesmo requisitos necessários para os próximos elementos a partir de interações com o cliente (202). Da mesma forma, a divisão de requisitos em unidades menores facilita a leitura e compreensão (214, 202). Alternativamente, Medeiros et. al (214) alertam que a fragmentação de requisitos de software em muitos artefatos e muitos detalhes podem comprometer a qualidade dos requisitos e contribuir para a falta de manutenção na documentação (214).

Por sua vez, casos de uso são considerados histórias de usuários mais completas do ponto de vista técnico (202) e os templates existentes de caso de uso têm foco em pontos de variabilidade entre cenários e em sua rastreabilidade (213). Casos de uso podem ser utilizados para abordar requisitos funcionais (240) e não funcionais, como realizado por Nguyen (241), o qual estendeu um template já existente de caso de uso para a especificação de requisitos não-funcionais - performance, usabilidade e segurança. Entretanto, a complexidade suportada pelos casos de uso também foi apresentada como ponto negativo por Medeiros et. al (214), destacando a quantidade excessiva de informação e o esforço necessário para a descrição de cenários, sugerindo a necessidade de equilíbrio entre a abrangência de requisitos e a extensão do texto.

As ontologias são utilizadas para mitigar problemas de ambiguidade em requisitos (225) e podem abordar requisitos funcionais e não-funcionais. Bellendorf et. al (232) utiliza ontologias para requisitos de interoperabilidade, uma vez que um mesmo termo pode ter usos diferentes dentro do contexto de cada tipo de sistema (232). Em outro estudo, as ontologias são utilizadas como informação adicional para a geração de casos de teste (230). Por fim, o artefato de visão, apesar de mencionado em dois estudos, não foi atribuído à abordagem de requisitos funcionais ou não funcionais.

Artefatos predominantemente visuais:

Artefatos predominantemente visuais estão associados à representação que mimetiza intencionalmente os elementos de interface do software que serão visualizados por um ou mais grupos de usuários finais, de

forma a prever e testar a apresentação de um sistema. Estão bastante relacionados à requisitos funcionais pela proximidade de interação com o usuário final, entretanto, podem ser utilizados para abordar aspectos não funcionais. Os protótipos podem elucidar fluxos funcionais solicitados por stakeholders e usuários, bem como auxiliar na especificação de requisitos de usabilidade (202). Ogata et. al sugere a combinação de protótipos funcionais com diagramas UML para abordagem de requisitos funcionais e não funcionais de forma mais completa (217). As personas, de forma geral, são mencionadas como artefatos que abordam principalmente o requisitos de usabilidade (202, 220).

Em geral, uma tendência notável na análise dos artigos é a atenção dada aos requisitos não funcionais de segurança (235, 213, 234, 214, 202, 230, 231, 232, 225), usabilidade (236, 213, 234, 214, 202, 232) e performance (235, 214, 234, 229, 232) . Tal fator pode ser explicado pela complexidade crescente de sistemas de software e a ameaça de ataques cibernéticos que violem dados pessoais e sensíveis, exigindo cada vez mais requisitos de segurança. Além disso, os usuários modernos têm expectativas mais elevadas em relação à usabilidade e performance. As próprias regulamentações e padrões às quais as empresas estão sujeitas também reforçam a adoção de medidas preventivas e protetivas.

3.7.3 QP.3 - Quais são as diretrizes, estruturas ou formatos utilizados nos artefatos de documentação dos requisitos em linguagem natural?

Além de modelos para geração de artefatos, existem normas e diretrizes gerais estabelecidas para a escrita eficaz de requisitos de software. Nos artigos analisados, também houveram referências a modelos de linguagem, *Frameworks* de elicitação de requisitos e propostas de avaliação da qualidade dos requisitos. Entretanto, nesta revisão, estamos nos atendo principalmente a modelos ou normas que gerem ou sirvam de base para a geração de artefatos documentais referentes à especificação de requisitos. Desta forma, ferramentas utilizadas para gerar artefatos com as notações encontradas não foram incluídos na análise. Em 2009, O template PLUS com extensão de requisitos não funcionais de Nguyen (241) estende o template de Gooma para especificação de requisitos não funcionais.

Tipo	Padrão	Itens mencionados
		<ul style="list-style-type: none"> • <i>Unified Model Language</i> (UML) (242, 233); • <i>Event-Driven Process Chain</i> (EPC) (229); • <i>Business Process Model Notation</i> (BPMN) (229) ; • Archimate (243);
<i>UML Diagram</i>	Notação	<ul style="list-style-type: none"> • <i>Systems Modeling Language</i> (SYSML) (242); • <i>Modeling and Analysis of Real-Time and Embedded systems</i> (MARTE) (244); • <i>Object Constraint Language</i> (OCL) (245) • <i>MARTE Clock Constraint Specification Language</i> (CCSL) (246); • <i>Entity Relationship Models</i> (ERM) (226); • BPMN4TOSCA (211)
	Framework	<i>Physics of Notation</i> (PoN) (247, 218);
	Modelo	<i>UML RA Model</i> (217);

Continua na próxima página

Tabela 3.5 – continuação da página anterior

Tipo	Padrão	Itens mencionados
<i>Use cases</i>	Artefato	<ul style="list-style-type: none"> • Catálogo de padrões de caso de uso (202, 248); • Meta-modelo de caso de uso (202, 248); • PLUC (Product Line Use Cases) by Bertolino and Gnesi (249, 213); • PLUS de Gooma (250); • Template de Bragança and Machado (251); • PLUSS de Eriksson et.al (252); • Use Case Elicitation Template (UCET) (253); • Template de Choi (254); • Use Case Editor (UCed) (255); • PLUS com extensão de requisitos não funcionais (241); • Template de Jirapanthong (256); • Modelling Scenario Variability as Crosscutting Mechanisms (MSVCM) (257); • Template de Oliveira (258);
<i>Conceptual models</i>	Framework	<ul style="list-style-type: none"> • i* (259) ; • KAOS (239) ; • NFR <i>Framework</i> (238) ; • Tropos (239) ;
	Notação	<i>Goal-oriented Requirement Language (GRL)</i> (228);

Continua na próxima página

Tabela 3.5 – continuação da página anterior

Tipo	Padrão	Itens mencionados
Mind map	Framework	SnapMind (202, 260).
<i>User story card</i>	Artefato	<i>W8 User Story Card Model</i> (240, 202).
<i>Scenarios</i>	Artefato	BDD - cenários de teste (261, 216).
<i>User story, Prototype, Scenario, Persona</i>	Framework	HCD para desenvolvimento ágil (ISO 9241-210) (262).
ISO/IEC/IEEE 9241-210	Norma	Ergonomics of human-system interaction — Part 220: Processes for enabling, executing and assessing human-centred design within organizations (175).
ISO/IEC/IEEE 29100	Norma	Requisitos em segurança e privacidade de software (263).
ISO/IEC/IEEE 29148:2011	Norma	Padrões de qualidade em requisitos (264).
ISO/IEC/IEEE 830:1998	Norma	Prática recomendada para Especificações de Requisitos de Software. Substituída pela ISO/IEC/IEEE 29148:2011 (265).

Tabela 3.5: Padrões em documentação de software.

Diagramas UML

A notação UML é extensivamente utilizada em metodologias ágeis (210), e serve de base para as notações e modelos específicos em diagramas de software (208). Assim, cada uma das notações de diagramas apresentadas na tabela 3.5 se propõe a atender um contexto específico. A notação BPMN é uma das vertentes UML mais conhecidas, empregada para modelar aspectos negociais (218). Outra vertente da UML é a ERM (modelo de relacionamento de entidade), comumente utilizado para representar aspectos estruturais como banco de dados (210).

Como a modelagem de negócios ainda pode ser considerada um escopo amplo, existem derivações da BPMN que facilitam sua utilização em contextos específicos. Na literatura analisada, a extensão BPMN4TOSCA é descrita como a BPMN que possui acesso direto a elementos de topologia, facilitando assim a representação da implantação, portabilidade, interoperabilidade e reusabilidade de componentes individuais das aplicações em nuvem (232, 211). Outra derivação é a EPC - Cadeia de Eventos Orientados

a Processo - que possui maior simplicidade gráfica, porém é baseada no desencadeamento de processos a partir de eventos e possibilita a predição de comportamentos ainda não observados (229). Já a Archimate se propõe a representar diferentes camadas da arquitetura empresarial (218).

Assim como estas, as outras notações também são indicadas para contextos específicos. MARTE é utilizado em conjunto com UML para capturar requisitos de comportamento em sistemas embarcados por sua característica temporal (233), enquanto os diagramas de definição de bloco da SYSML são mais indicados para especificar aspectos estruturais. Para especificação abstrata de propriedades/ restrições, padrões como OCL e CCSL são frequentemente utilizados (233). Em contrapartida, uma revisão mais recente (218), relata que um dos artigos explicita a insatisfação com o crescente aumento de notações existentes, alegando uma lacuna na literatura que demonstre a adoção de novas notações por profissionais de requisitos (219).

Além das notações visuais, o framework Physics of Notation (PoN) compreende uma série de recomendações mais populares e abrangentes para garantir notações cognitivamente eficazes (227, 247). Complementarmente, o modelo UML RA foi proposto por Ogata e Matsura (217) para intercambiar diagramas em interfaces. O modelo consiste em diagramas de atividades, um diagrama de classes e diagramas de objetos. Os três tipos de diagramas são transformados em um protótipo de interface de usuário usando as regras de correspondência entre elementos UML e componentes da interface (217).

Casos de uso

Os casos de uso apresentam vários templates mencionados, com foco especial em facilitar a variabilidade (213). O template PLUC (Product Line Use Cases), proposto por Bertolino e Gnesi em 2003, é projetado para permitir a descrição de variações em casos de uso pela inclusão explícita de tags dentro das seções dos casos de uso para indicar as partes variáveis (249). O template PLUS, proposto por Gomaa em 2005, se destaca por permitir a especificação de variações granulares em elementos como nome, tipo, linha do caso de uso (alvo da variação) e descrição (250). Além disso, propõe a descrição das variações ao final dos casos de uso para tornar a identificação mais fácil. Por fim, o template também possui uma estrutura de seção específica, que, assim como a descrição ao final, descreve a variabilidade sem afetar a descrição do cenário principal do caso de uso.

No mesmo ano, Bragança e Machado propuseram um template que apresenta o uso de Tags opcionais e alternativas especificar variantes utilizando qualquer fragmento de texto do caso de uso, pelo uso de pares de tags no estilo XML como <variant> e </variant>. Outra característica interessante é a possibilidade de inserir perguntas relacionadas aos pontos de variação (251). Ainda em 2005, Eriksson et. al (252) propõem um template que suporta os cinco tipos de variabilidade (213) e possui identificador de etapa - que também é usado para especificar o comportamento da variante. Outra característica distintiva deste modelo é o uso de variáveis locais (com \$) e globais (com @) (252).

Em 2007, Gallina e Guelfi (253) propuseram o Use Case Elicitation Template (UCET), que fornece campos especiais de informação, como obrigatoriedade do caso de uso, descrição dos pontos de variação e descrição da variação da falha, utilizada para descrever as falhas fortemente relacionadas aos pontos de variação. No ano seguinte, Choi et al. (254) propuseram uma notação de tags mais simples do que Bertolino e Gnesi (2003), na qual são utilizadas apenas para marcar pontos de variação em cenários. Cada tag é expandida em uma seção chamada “Variações” e mapeada para o Modelo de Variabilidade Ortogonal

(OVM).

Também em 2008, Anthonysamy e Somé (255) foi proposta uma abordagem de modelagem de casos de uso orientada a aspectos que é suportada pela ferramenta Use Case Editor (UCed). Em 2009, Nguyen (241) estende o modelo de Goma (250) para especificar requisitos não funcionais. Neste modelo, seções adicionais são usadas para especificar requisitos de desempenho, usabilidade e segurança. Posteriormente, Jirapanthong propõe um template de variação baseada em tags, mas com um atributo diferencial para especificar o domínio e membro do produto no qual o caso de uso é especificado.

Em 2009 também foi proposta a "Modelling Scenario Variability as Crosscutting Mechanisms"(MSVCM) por Bonifacio e Borba (257), para lidar com a variabilidade de cenários como uma composição de diferentes artefatos: modelo de casos de uso, modelo de recursos, configuração de produto e conhecimento de configuração (257). Já em 2011, Issa e Al-Ali (248) analisaram vários casos de uso para criar o catálogo de padrões de caso de uso e um meta-modelo de especificação, que pode ser utilizado como checklist para funcionalidades e ser um parâmetro para uma versão inicial de modelo de caso de uso. Os autores avaliaram a qualidade do catálogo por entrevistas regulares com usuários, e relataram a diminuição no tempo gasto na fase da engenharia de requisitos (248). Por fim, o modelo apresentado por Oliveira et al. (2013) é uma adaptação do modelo de Eriksson et al. (252), no qual sugere-se que os nomes dos recursos devem ser claramente especificados em um elemento chamado "Recurso associado" (252).

Modelos conceituais

A modelagem de objetivos é aplicada a fim de entender as necessidades e expectativas dos stakeholders, abstraindo informações desnecessárias, como detalhes de implementação. Por este contexto, as linguagens de modelagem de objetivos - incluindo KAOS, i*, NFR Framework, Tropos e GRL - foram criadas (228).

O guia KAOS, proposto em 1993, enfatiza a completude e consistência dos requisitos, porém não oferece suporte para bibliotecas de modelos de objetivos, o que limita a reutilização (259). Já o framework i* se concentra na intencionalidade dos stakeholders, modelando as metas e relações sociais entre eles para elucidar como requisitos são influenciados por diferentes atores (266). É considerado difícil de reutilizar devido à sua reutilização limitada a uma cópia básica e colagem de elementos designados (228).

O NFR Framework foi projetado para especificar requisitos não funcionais (NFRs) e sua influência no processo de desenvolvimento de software. Diferente de outros frameworks, foca exclusivamente em softgoals sem cobrir outros elementos intencionais como metas, tarefas e recursos (238).

Tropos é um framework baseado em agentes e objetivos, similar ao i*. Entretanto, abrange todo o ciclo de vida do desenvolvimento de software, desde a captação inicial de requisitos até a implementação (239). Por fim, o Goal-oriented Requirement Language (GRL), integrante da User Requirements Notation (URN), suporta a representação de estratégias e crenças dos atores envolvidos, porém não define claramente o contexto no qual estão inseridos (228).

Mapa mental

Mapas mentais podem ser úteis principalmente no início do desenvolvimento de software, para comunicação com *stakeholders* e montagem do *backlog* de produto (267). Proposto por Wanderley et. al (260) em 2014, o *SnapMind* é um *framework* de modelagem de requisitos centrado no usuário, pela definição de linguagem visual de requisitos baseada em mapas mentais (260). Foi criado para promover a consistência

entre histórias de usuário e e modelos de domínio por meio de *snapshots* (técnica de animação de modelos).

Cartão de História de usuário

Apesar de histórias de usuário possuírem uma estrutura bastante conhecida e utilizada, os cartões de história atuam como complementos às HUs. Nesse sentido, Farid (240) definiu o *W8 User Story Card Model*, como um artefato que deve conter estas 8 informações: quem, o quê, porquê, não ignorando, enquanto é bom ter, dentro de, qual a prioridade e possível impacto.

Cenários

O Desenvolvimento Guiado por Comportamento (BDD, do inglês Behavior Driven Development) utiliza a linguagem natural para descrever os resultados esperados a partir dos requisitos, usando cenários de testes no padrão "Dado-Quando-Então", úteis tanto para a geração de casos de teste quanto para a execução de relatórios. Além disso, Farooq et. al (216) destaca que uma parte fundamental do BDD é a refatoração, utilizada para melhorar a manutenibilidade dos artefatos (216).

Framework HCD para desenvolvimento ágil

Maguire et. al (262) utilizaram o Framework HCD, definido na ISO 9241-210, como referência para o desenvolvimento ágil. O framework consiste, principalmente, nas atividades de a) entender e especificar o contexto de uso, b) especificar os requisitos do usuário, c) desenhar soluções e d) avaliar se as soluções atendem aos requisitos. Maguire et. Al propõem uma extensão ao framework HCD para o contexto ágil, sugerindo métodos diversos para a execução das diferentes atividades, e também recomenda a elaboração de artefatos bem definidos que reúnam as informações coletadas.

Normas ISO/IEC/IEEE

As normas ISO/IEC/IEEE são referências de padrões de qualidade na área de engenharia de software (265), portanto, as citações de padrões nos artigos da área são recorrentes. De fato, uma norma técnica estabelece definições, classificações, delineamento de procedimentos, medição de qualidade, processos, produtos, sistemas, serviços, métodos de teste e procedimentos de amostragem uniformes de engenharia ou técnicos (268).

A ISO/IEC/IEEE 830-1998 (Prática recomendada para Especificações de Requisitos de Software) descreve o conteúdo e as qualidades de uma boa especificação de requisitos de software (SRS), bem como apresenta esboços de especificações. Define conceitos iniciais de features, objetos, respostas, hierarquia funcional, modos de sistema, estímulos e classes de usuários (265). Foi substituída pela ISO/IEC/IEEE 29148:2011 (Norma Padrões de qualidade em requisitos).

A norma ISO/IEC/IEEE 29148:2011 fornece definições em detalhes, tanto para o manejo de requisitos quanto para a área de negócio. Em relação ao gerenciamento de requisitos, são definidos conceitos como gestão de mudanças e mensuração de requisitos. Para a área de negócio, também estão presentes definições como atores, objetivos, processos, cenários, leis, e modelo de qualidade. A norma também especifica que, durante o projeto, devem ser produzidos pelo menos 3 itens: 1) documento de especificação dos stakeholders (StRS); 2) documento de especificação do sistema (SyRS); e 3) documento de especificação dos requisitos de software (SRS), em concordância com a ISO/IEC 12207 (264). Entretanto, a norma não define um artefato específico, apenas informações gerais que devem constar nos documentos. Por fim, o

documento também define os conceitos de requisitos dos usuários (264).

No mesmo ano, a ISO/IEC/IEEE 29100:2011 foi publicada para tratar especificamente dos requisitos em segurança e privacidade de software (263). Norteia a proteção de dados pessoais em sistemas de TIC ao especificar terminologias comuns de privacidade, definindo atores e seus papéis no processamento de informações pessoais identificáveis (PII). Adicionalmente, descreve considerações para a proteção da privacidade e fornece referências a princípios de privacidade conhecidos para a tecnologia da informação (263).

Por fim, a ISO/IEC/IEEE 9241-210 - Ergonomics of human-system interaction compõe o framework HCD (175), por meio de orientações sobre design centrado no ser humano, enfatizando princípios de adequação para tarefas do usuário, autoexplicabilidade, conformidade de expectativas, aprendizagem, controlabilidade, robustez de erro de uso e engajamento. O documento visa a implementação do design centrado no ser humano como parte de um processo do desenvolvimento e ciclo de vida de um sistema, bem como avaliação da qualidade e melhoria da centralização de processos no usuário por parte de uma empresa ou sistema.

3.8 SUMÁRIO DO CAPÍTULO

O capítulo realiza uma Revisão Sistemática da Literatura (SLR) focada na documentação de requisitos em linguagem natural para software. Identificou-se uma variedade de artefatos utilizados para essa finalidade, nos quais os mais referenciados na literatura foram diagramas, casos de uso, modelos conceituais, histórias de usuário e protótipos. A análise destacou que os artefatos são aplicados de forma distinta para requisitos funcionais e não-funcionais. De forma geral, os diagramas, casos de uso, cenários e protótipos podem ser utilizados para os dois tipos de requisitos, a depender do teor (usabilidade, segurança, etc). Entretanto, as histórias de usuários e artefatos derivados são mais recomendadas para requisitos funcionais e possuem limitações para requisitos não-funcionais. Além disso, o estudo explorou diferentes diretrizes, estruturas e formatos empregados nos artefatos de documentação, refletindo a diversidade nas práticas de documentação de requisitos em projetos de software.

4 REVISÃO DE CARACTERÍSTICAS CONVERSACIONAIS EM CHATBOTS

O presente capítulo apresenta uma Revisão Sistemática da Literatura (SLR) seguindo o protocolo de Kitchenham e Charters (205) para investigar de forma específica os produtos da documentação de requisitos de software no desenvolvimento de chatbots, de forma a responder a QP.3 deste estudo: "Como são documentados os requisitos conversacionais de chatbots, e quais aspectos conversacionais se fazem necessários na documentação de requisitos?".

4.1 QUESTÃO DE PESQUISA

A motivação para conduzir uma Revisão Sistemática da Literatura (SLR) reside na natureza diferencial de chatbots em relação a outros softwares. Foi observado na revisão anterior (Capítulo 3) que muitos artefatos podem ser utilizados para a apresentação e registro de requisitos funcionais e não-funcionais de software, e os principais requisitos não-funcionais registrados foram os de segurança, usabilidade e performance. Entretanto, não há menções diretas aos requisitos conversacionais apresentados na Seção 2.2, que são o foco deste trabalho.

Portanto, neste capítulo é abordada a questão de pesquisa 3 desta dissertação - "*Como são documentados os requisitos conversacionais de chatbots, e quais aspectos conversacionais se fazem presentes na documentação de requisitos?*". A questão de pesquisa é direcionada aos requisitos de software elicitados pelos analistas e engenheiros de requisitos.

4.2 SEQUÊNCIA TEXTUAL DE BUSCA

Seguindo o padrão do capítulo anterior, o Framework escolhido para a criação da sequência textual de busca foi o PICOC (em português: População, Intervenção, Comparação, Resultado, Contexto) (203). Neste framework, a população se refere ao objeto de estudo; a intervenção é o meio utilizado pela população; a comparação é o que está sendo comparado com a intervenção; o resultado é o desfecho da intervenção; e o contexto é o foco do estudo, suas restrições e limitações.

A Tabela 4.1 mostra a definição final dos termos PICOC para construir a cadeia de pesquisa genérica. A comparação não é aplicável porque a intervenção não é comparada. A sequência textual de busca foi traduzida e aplicada em inglês, visto que a maior parte da literatura encontra-se nesta língua. As bases de dados digitais escolhidas para aplicar a sequência de palavras foram as mesmas do capítulo anterior, em virtude de sua relevância na pesquisa em engenharia de software (204), na abrangência na indexação de conferências e jornais e na possibilidade de aplicar a sequência de busca genérica diretamente no campo de pesquisa.

PICOC	Palavras-chave	Palavras relacionadas
População	Chatbot	Interface de conversação, interface conversacional, agente conversacional
Intervenção	Documentação	Especificação, requisito, elicitación
Comparação	Não aplicável	Não aplicável, pois a comparação não é necessária neste contexto.
Resultado	Artefato	Guia, conjunto de ferramentas, recomendação, técnica, prática, framework, documento, formato
Contexto	NOT "chabots para requisito"	NOT "chabot para elicitación"

Tabela 4.1: Termos PICOC - em português - utilizados para construção de sequência de busca.

Sequência de busca:

("chatbot"OR "conversation interface"OR "conversational agent"OR "conversational interface") AND ("documentation"OR "elicitation"OR "requirement"OR "specification") AND ("artifact"OR "document"OR "format"OR "framework"OR "guideline"OR "practice"OR "recommendation"OR "technique"OR "toolkit") AND (NOT "chatbots for requirement"OR NOT "chabot to elicit")

Em sequência ao capítulo anterior, foram utilizadas as mesmas bases de dados para a pesquisa, seguindo a metodologia de Kitchenham (205): ACM, IEEEExplore, SCOPUS, Springer Link e Web of Science. A sequência de busca inicial foi adaptada para cada uma das fontes e a pesquisa foi feita pelo título, palavras-chave e resumo (abstract) dos artigos.

**Base de in- Sequência Textual
dexação**

ACM Digital Library	<pre>[[[Title: "chatbot"] OR [Title: "conversation interface"] OR [Title: "conversational agent"] OR [Title: "conversational interface"]] AND [[Title: "documentation"] OR [Title: "elicitation"] OR [Title: "requirement"] OR [Title: "specification"]] AND [[Title: "artifact"] OR [Title: "document"] OR [Title: "format"] OR [Title: "framework"] OR [Title: "guideline"] OR [Title: "practice"] OR [Title: "recommendation"] OR [Title: "technique"] OR [Title: "toolkit"] OR [Title: "design"]] AND NOT [[Title: "chatbots for requirement"] AND [Title: "chatbot to elicit"]]] OR [[Abstract: "chatbot"] OR [Abstract: "conversation interface"] OR [Abstract: "conversational agent"] OR [Abstract: "conversational interface"]] AND [[Abstract: "documentation"] OR [Abstract: "elicitation"] OR [Abstract: "requirement"] OR [Abstract: "specification"]] AND [[Abstract: "artifact"] OR [Abstract: "document"] OR [Abstract: "format"] OR [Abstract: "framework"] OR [Abstract: "guideline"] OR [Abstract: "practice"] OR [Abstract: "recommendation"] OR [Abstract: "technique"] OR [Abstract: "toolkit"] OR [Abstract: "design"]] AND NOT [[Abstract: "chatbots for requirement"] AND [Abstract: "chatbot to elicit"]]] OR [[[Keywords: "chatbot"] OR [Keywords: "conversation interface"] OR [Keywords: "conversational agent"] OR [Keywords: "conversational interface"]] AND [[Keywords: "documentation"] OR [Keywords: "elicitation"] OR [Keywords: "requirement"] OR [Keywords: "specification"]] AND [[Keywords: "artifact"] OR [Keywords: "document"] OR [Keywords: "format"] OR [Keywords: "framework"] OR [Keywords: "guideline"] OR [Keywords: "practice"] OR [Keywords: "recommendation"] OR [Keywords: "technique"] OR [Keywords: "toolkit"] OR [Keywords: "design"]] AND NOT [[Keywords: "chatbots for requirement"] AND [Keywords: "chatbot to elicit"]]]]</pre>
IEEE Xplore	<pre>("chatbot"OR "conversation interface"OR "conversational agent"OR "conversational interface") AND ("documentation"OR "elicitation"OR "requirement"OR "specification") AND ("artifact"OR "document"OR "format"OR "framework"OR "guideline"OR "practice"OR "recommendation"OR "technique"OR "toolkit")</pre>
Scopus	<pre>(TITLE-ABS-KEY ("chatbot"OR "conversation interface"OR "conversational agent"OR "conversational interface")) AND (TITLE-ABS-KEY ("documentation"OR "elicitation"OR "requirement"OR "specification")) AND (TITLE-ABS-KEY ("artifact"OR "document"OR "format"OR "framework"OR "guideline"OR "practice"OR "recommendation"OR "technique"OR "toolkit")) AND DOCTYPE(re) AND PUBYEAR > 2013 AND PUBYEAR < 2024</pre>

Continua na próxima página

Tabela 4.2 – continuação da página anterior

Base de indexação	Sequência Textual
Springer Link*	("chatbot"OR "conversation interface"OR "conversational agent"OR "conversational interface") AND ("documentation"OR "elicitation"OR "requirement"OR "specification") AND ("artifact"OR "document"OR "format"OR "framework"OR "guideline"OR "practice"OR "recommendation"OR "technique"OR "toolkit")
Web of Science	(TS=("chatbot"OR "conversation interface"OR "conversational agent"OR "conversational interface") AND ("documentation"OR "elicitation"OR "requirement"OR "specification") AND ("artifact"OR "document"OR "format"OR "framework"OR "guideline"OR "practice"OR "recommendation"OR "technique"OR "toolkit")) NOT TS=("chatbots for requirement"OR "chabot to elicit")

Tabela 4.2: Adaptação de sequência textual às diferentes bases bibliográficas.

4.3 CRITÉRIOS DE SELEÇÃO

4.3.1 Critérios de inclusão

Em consonância à proposta de Kitchenham (201), foram definidos critérios de inclusão e exclusão. Os critérios de inclusão foram baseados no tema principal dos artigos, alinhados ao tema do trabalho e cujo texto aborde artefatos que documentam especificamente softwares conversacionais, a saber:

- (IC 1) Os artigos devem apresentar informações relacionadas aos requisitos conversacionais em chatbots;
- (IC 2) O artigo apresenta artefatos contendo especificação de requisitos em chatbots;
- (IC 3) O artigo apresenta análises empíricas (estudos de caso, experimentos, pesquisas) ou teóricas (revisões sistemáticas, análises conceituais) sobre documentação de requisitos de chatbots.

4.3.2 Critérios de exclusão

Os critérios de exclusão têm como objetivo excluir trabalhos que atendam aos critérios de inclusão, mas não apresentem alguma metodologia, foco, aspecto ou abordagem. Os critérios de exclusão (EC) são mostrados abaixo:

- (EC 1) Artigos que não estejam relacionados à documentação dos requisitos do chatbot ou que não abordem diretamente as questões de pesquisa mencionadas acima;

- (EC 2) Artigos que sejam escritos em um idioma diferente do compreendido pelos autores (português, espanhol e inglês);
- (EC 3) O artigo não está disponível ou foi retirado pelo editor;
- (EC 4) O foco do artigo não é a análise dos artefatos de documentação para chatbots, mas outros procedimentos (i.e: automação, classificação com NLP, aplicação de chatbots para elicitación ou documentação de requisitos gerais, técnicas de elicitación, etc);
- (EC 5) O texto não segue o formato de artigo científico (artigos de opinião, livros, editoriais, relatórios técnicos, teses, dissertações, relatórios de conferências, resumos de apresentações ou pôsteres).
- (EC 6) O assunto principal do artigo não está relacionado à engenharia de software ou áreas afins, mas sim a outras áreas de pesquisa.

É importante mencionar que não foi incluído um critério de data, visto que procuramos ao máximo não restringir os artigos que mencionassem os temas de interesse.

4.4 AVALIAÇÃO DE QUALIDADE

Os critérios de inclusão e exclusão são filtros importantes para trazer apenas os artigos relevantes para as questões de pesquisa. Entretanto, é necessário verificar a qualidade dos artigos selecionados, para garantir que as informações inferidas sejam representativas para a literatura de software. Portanto, os critérios utilizados para avaliação da qualidade foram:

- (Q1) Os artefatos de especificação que abordam os requisitos conversacionais são pragmáticos e replicáveis?
- (Q2) As limitações e ameaças à validade do estudo foram evidenciadas e explicadas?
- (Q3) Pelo menos um dos artefatos de documentação de chatbots presentes no artigo aborda requisitos de conversação?
- (Q4) O artigo mostra o artefato completo ou um exemplo?

Cada artigo pré-selecionado foi avaliado em relação às 4 questões, e somente os artigos que atendiam a todos os critérios propostos foram incluídos na análise. As pontuações possíveis para cada critério foram:

- Não: 0;
- Parcialmente: 0,5;
- Sim: 1,0.

Somente os trabalhos cuja resposta foi "Sim" para cada um dos critérios foram incluídos, ou seja, somente trabalhos que fizeram 4 pontos.



CI = critério de inclusão, CE = critério de exclusão, AQ = avaliação de qualidade.

Figura 4.1: Artigos restantes após cada etapa de condução da SLR.

4.5 EXTRAÇÃO DE DADOS

Para responder às questões propostas, foram extraídos dados referentes à publicação (título, autores, ano de publicação, resumo, DOI, palavras-chave e tipo de artigo (artigo de pesquisa, revisão de literatura, revisão sistemática, meta-revisão ou revisão terciária). Além disso, foram extraídos dados referentes às questões de pesquisa em si, como tipo de chatbot abordado (baseado em regras, generativo, input de texto ou voz, etc), artefatos utilizados, requisitos conversacionais especificados, contexto de uso do artefato, e recomendações ou observações do artigo, quando houvesse.

4.6 CONDUÇÃO DA PESQUISA

Para a realização desta revisão, foi utilizada a ferramenta Parsifal, uma plataforma web gratuita e de código aberto destinada ao apoio à Revisão Sistemática da Literatura (SLR), assim como no capítulo anterior 3. A Figura 4.1 ilustra os artigos que permaneceram após cada etapa de condução.

O período de coleta dos estudos através da string de busca ocorreu durante o mês de janeiro de 2024, resultando em um total de 933 artigos (18 da ACM, 42 da IEEE Xplore, 409 da SCOPUS, 416 da Springer Link e 48 da Web of Science), conforme visto na Figura 4.1. Foram removidos 137 estudos duplicados, restando 796 artigos (9 da ACM, 28 da IEEE Xplore, 348 da SCOPUS, 378 da Springer Link e 33 da Web of Science) para aplicar os critérios de inclusão e exclusão, lendo o título e o resumo. Embora os artigos analisados pelo título e resumo abordassem o tema de requisitos, 649 foram removidos devido à não observância de um ou mais critérios. Por último, foi realizada a leitura do texto completo para uma análise mais aprofundada e aplicação da avaliação de qualidade de 147 artigos (5 da ACM, 7 da IEEE Xplore, 58 da SCOPUS, 61 da Springer Link e 16 da Web of Science), o que resultou na remoção de 105 trabalhos e no número final de 42 artigos de revisão selecionados (1 da ACM, 2 da IEEE, 22 da SCOPUS, 10 da Springer Link e 7 da Web of Science), como apresentado na tabela 4.3.

Base de indexação	Estudos
ACM	(269).
IEEE Xplore	(270), (271).
SCOPUS	(272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 43, 287, 288, 289, 290, 291, 292).
Springer Link	(293, 294, 295, 296, 297, 298, 299, 300, 301, 302).
Web of Science	(303, 304, 305, 306, 307, 308, 309).

Tabela 4.3: Estudos restantes após aplicação de todas as etapas da SLR.

4.7 RESULTADOS DA SLR

Esta seção apresenta uma análise detalhada dos resultados obtidos pela extração de dados efetuada na Revisão Sistemática da Literatura (SLR). Até o momento (janeiro de 2024), não foi encontrado um artigo de revisão que apresente o estado da arte sobre a documentação de requisitos conversacionais, e portanto o presente trabalho visa preencher esta lacuna por meio da investigação e discussão sobre as práticas de registro dos requisitos conversacionais realizados por times de software. Esta discussão é fundamentada no somatório dos resultados dos estudos selecionados, especialmente considerando aqueles que exploram aspectos similares ou correlatos à documentação de requisitos conversacionais.

4.7.1 QP 1. Como são documentados os requisitos conversacionais de chatbots, e quais aspectos conversacionais se fazem necessários na documentação de requisitos?

4.7.1.1 Como são documentados os requisitos conversacionais de chatbots?

Nos artigos selecionados, foram identificados 26 tipos de artefatos utilizados na documentação de requisitos conversacionais, dos quais 15 estão presentes em pelo menos dois artigos, como apresentado na figura 4.3 .

Todos os artigos selecionados foram escritos na língua inglesa e, dentre estes, a maior parte foi publicada entre 2019 e 2024 (4.2), apesar de não ter sido estabelecido um parâmetro para a data de publicação. os *UML diagrams*, *prototypes*, *tables of requirements*, *conversational flows* e *scenarios* foram, respectivamente, os artefatos mais recorrentes dentre os artigos selecionados.

De forma geral, os nomes foram mantidos da mesma forma que apareciam nos trabalhos selecionados. Entretanto, termos similares ou artefatos que se encaixavam na mesma categoria foram agrupados e apresentados em um nome único. Como exemplo, citamos os *conversational flows* (fluxos conversacionais), que apareciam nos estudos como *“conversational streams”*, *“dialogue flows”* ou *“interaction flows”*, e foram agrupados em *“conversational flows”*.

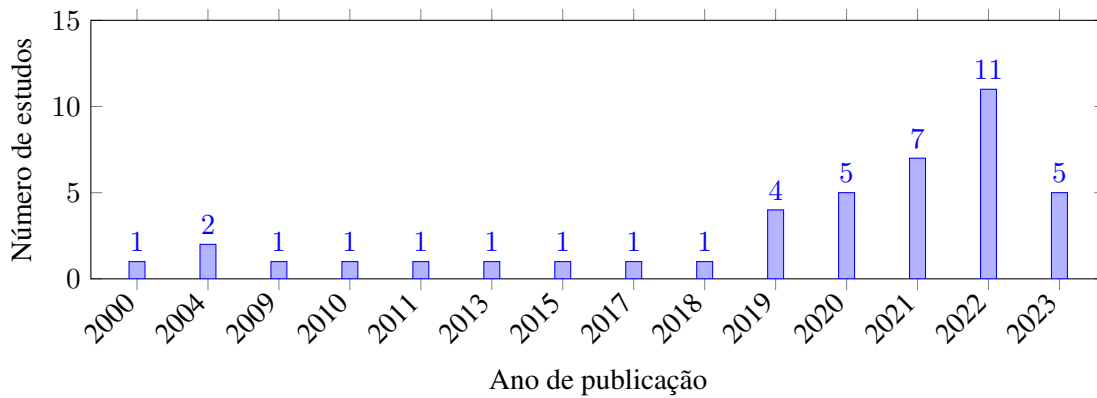


Figura 4.2: Distribuição dos anos de publicação dos estudos da SLR.

Foi observado que os *UML diagrams*, *prototypes*, *tables of requirements*, *conversational flows*, e *scenarios* foram os mais utilizados nos estudos abordados (figura 4.3).

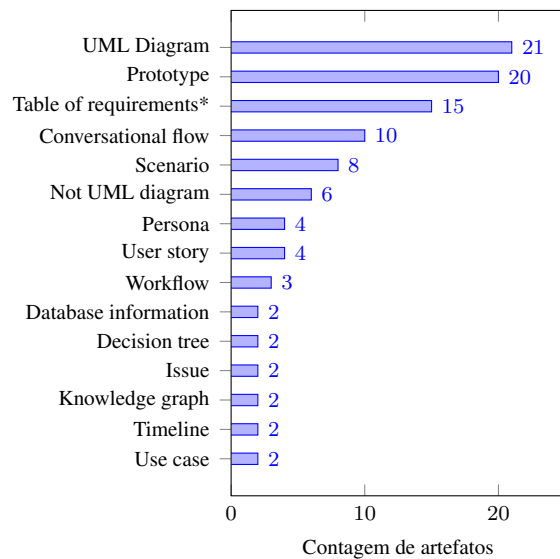


Figura 4.3: Contagem de artefatos encontrados nos estudos.

Além dos artefatos supracitados, na sequência artefatos como outros diagramas (fora dos padrões UML), personas, histórias de usuários, *workflows*, informações de banco de dados, *decision trees*, *issues*, gráficos de conhecimento, *timelines* e casos de uso também foram mencionados em mais de um estudo. Embora menos citados, sua presença pode indicar necessidades específicas de documentação em diferentes contextos.

Entre os artefatos mencionados apenas uma vez (ou seja, não aparecem no gráfico), estão as ontologias, questionário de usabilidade de chatbot, checklist de design e implementação, vídeos, mapas mentais, mapas de empatia, análise de documentos, modelo e metamodelo de decisões de design, árvore de resultados, mapas de processos de negócios, e arquivo de história.

Além da quantidade de menções, os chatbots apresentados nos artigos também foram agrupados pela combinação dos atributos mais frequentes encontrados (tabela 4.4), a saber: entrada/saída (texto, fala, híbrido), domínio de conhecimento (específico ou amplo) e geração de respostas (baseada em regras, ge-

nerativa, híbrida ou baseada em tarefas). Alguns estudos também abordaram a plataforma de operação do chatbot (web, mobile ou dentro de um aplicativo existente).

Domínio de conhecimento	Geração de resposta	Entrada e saída	Citações
Específico	Baseada em regras	Texto	22
Específico	Generativa (IA)	Texto	19
Específico	Baseada em regras	Híbrido	14
Específico	Baseada em regras	Fala	13
Ampla	Baseada em regras	Híbrido	4
Específico	Baseada em tarefas	Fala	4
Específico	Generativa (IA)	Híbrido	2
Específico	Híbrido	Texto	2

Tabela 4.4: Número de chatbots por classificação agrupada.

Dentre as categorias agrupadas acima, a ocorrência mais comum encontrada foi de chatbots de conhecimento específico e com geração de respostas baseada em regras. A entrada e a saída variaram de texto para fala ou híbrida (tabela 4.4). A maioria dos estudos inclusos são trabalhos primários, motivo pelo qual a presença de chatbots com domínio restrito e geração de respostas menos robusta nas fases iniciais da investigação não é surpreendente.

A distribuição dos artefatos pelos grupos de categorias mais frequentes apresentou padrão semelhante aos resultados encontrados, com exceção da predominância de diagramas UML e protótipos, que variaram entre eles. As demais combinações não seguiram o padrão, mas apresentaram pelo menos um dos seis primeiros artefatos mais frequentes, conforme mostrado na tabela 4.4.

No geral, os resultados da SLR mostram que a documentação de requisitos para chatbots tem sido abordada na literatura, apesar de não ser o foco da maioria dos artigos. Podemos concluir que este é um tema de interesse de pesquisa recente, uma vez que nossa análise mostrou que a maioria dos estudos foi publicada entre 2019 e 2023 4.2.

Também é possível notar que os artefatos mais utilizados para chatbots são comuns no desenvolvimento de software em geral, como diagramas, protótipos e cenários. Isto se assemelha às descobertas de (43), que conduziu uma pesquisa com profissionais de desenvolvimento de chatbots. No capítulo anterior (3), os diagramas, casos de uso, histórias de usuário e protótipos também foram os artefatos mais encontrados. Curiosamente, os casos de uso e histórias de usuário, bastante utilizados pelos times de software, não são comuns ao registro de requisitos para chatbots, pelos nossos resultados.

Em relação a artefatos diferenciais, os fluxos conversacionais, estruturas em árvore e tabelas de requisitos, tão presentes em nosso estudo 4.3, não são comuns em outras vertentes do desenvolvimento de

softwares, pela revisão realizada anteriormente (capítulo 3).

Esta pode ser uma diferença entre a literatura e a prática em si, pois este padrão não foi encontrado em trabalhos relacionados. Por serem de certa forma abstratos, esta constatação pode estar relacionada às reclamações mencionadas por (43) em seu trabalho: os profissionais entrevistados relataram dificuldades encontradas ao desenvolver um chatbot, uma vez que os requisitos dos usuários e das partes interessadas são difíceis de desenvolver, testar, e estão sujeitos à subjetividade.

A subjetividade também poderia explicar a presença de um protótipo como um dos artefatos de documentação mais utilizados. Protótipos são artefatos flexíveis que podem abordar a aparência visual 4.4 e também comportamentos esperados verbais ou não-verbais, como dicas sociais, empatia, comportamento dependente do contexto e tratamento de erros, possuindo a capacidade de tornar a ideação menos abstrata. Os protótipos apresentam enorme flexibilidade, variando na fidelidade (baixa, média e alta fidelidade), e também na nomenclatura (sendo também chamados de mock-ups ou wireframes (202)).

Outro tópico de interesse relevante em nossa revisão é a presença encontrada de chatbots baseados em regras, específicos de domínio e de entrada de texto na literatura (tabela 4.4). Isto pode ser contraditório com o interesse atual da comunidade de software em chatbots de inteligência artificial (IA). No entanto, a implementação de chatbots de IA pode ser de alto custo, especialmente para ideias em estágio inicial. Os chatbots baseados em regras também podem ser mais fáceis de testar, pois são mais restritivos. Outra explicação possível é que o foco atual nos chatbots de IA não esteja fortemente relacionado à documentação, mas sim a questões relacionadas a dados ou privacidade, ou à análise sobre chatbots comerciais existentes.

4.7.2 Quais aspectos conversacionais estão presentes na documentação de requisitos?

Por "aspectos conversacionais", consideramos aqueles que aparecem diretamente para o usuário durante a conversa, como tom de voz, respostas condicionais, presença de elementos textuais e não textuais (emojis, indicadores de pausa, etc).

Alguns tipos de artefatos que são frequentemente utilizados na documentação de chatbots não abordam requisitos conversacionais com a mesma frequência, a exemplo dos diagramas UML 4.4. Apesar de estarem muito presentes em documentações de chatbot 3.2, seu conteúdo é versátil, variando entre arquitetura do software, relacionamentos entre componentes internos de reconhecimento e geração de resposta, dentre outros assuntos distintos na construção de um software. Com exceção deles, de forma geral os artefatos utilizados para documentação em chatbots abordam pelo menos algum requisito conversacional (figura 4.3).

A análise do conteúdo nos artefatos apresentou grande variabilidade de requisitos conversacionais (figure 4.5). A maior parte dos artefatos abordou a resposta diferencial adequada em situações específicas. Pelo fato de a maioria dos chatbots encontrados serem baseados em regras, este requisito efetivamente estabelece a base para a implementação de condutas que o chatbot deve seguir, similar às regras de etiqueta.

Assertividade, atitude humanizada e tratamento de erros aparecem em seguida na ordem de frequência. Os três requisitos demonstram que o objetivo é a facilidade que o usuário precisa ter em obter uma resposta

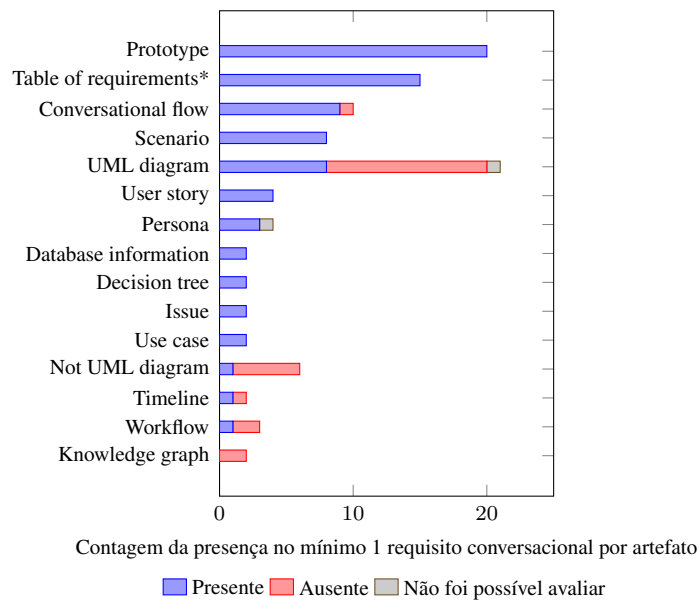


Figura 4.4: Presença de requisitos conversacionais em cada tipo de artefato.

correta e inteligível, independente da qualidade da entrada que recebeu. Logo após, o requisito de *feedback* está parcialmente relacionado, uma vez que os usuários finais. Aspectos subjetivos como proatividade, flexibilidade, dicas sociais, adaptabilidade, consciência emocional e empatia também foram abordados, mas em menor número (Figura 4.5). A aparência visual também foi uma preocupação relevante, principalmente no que diz respeito à percepção de entrada ou a primeira conversa com um chatbot. As questões relacionadas à segurança também foram abordadas de forma dispersa, pois os artigos abordaram diversas facetas da segurança, relacionada aos dados do usuário, à confiança, à privacidade bem como à transparência do agente conversacional.

A predominância de requisitos como "Comportamento dependente do contexto" e "Assertividade" destaca a necessidade de chatbots que não apenas respondam de forma relevante às entradas do usuário, mas também sejam capazes de antecipar necessidades e adaptar suas respostas com base no contexto da conversação. A capacidade de reconhecer uma entrada de texto ou fala não é mais suficiente: as expectativas dos usuários aumentaram para dialogar com um aplicativo semelhante ao humano.

Além disso, a ênfase no "Tratamento de erros" e na "Atitude humana" sublinha o desafio de desenvolver chatbots que possam lidar de forma agradável com ambiguidades, erros de digitação e erros inevitáveis em conversas naturais. A capacidade de um chatbot de responder a erros de maneira amigável pode melhorar significativamente a aceitação do usuário. Isto sugere que além da robustez técnica, a personalidade do chatbot e a modelagem de suas interações são aspectos críticos e devem estar sob atenção no desenvolvimento de um novo chatbot ou na melhoria de um já existente.

Por último, a diversidade de requisitos identificados, incluindo "Acessibilidade", "Histórico de Chat" e "Sugestões Sociais", indica a complexidade de projetar chatbots que sejam ao mesmo tempo funcionais, acessíveis e capazes de fornecer um ambiente rico e experiência responsiva. Essa variedade abordada em nossa pesquisa mostra que o desenvolvimento bem-sucedido de chatbots exige abordagens mais holísticas e interdisciplinares, envolvendo ciência da computação, psicologia, linguística e outros domínios para

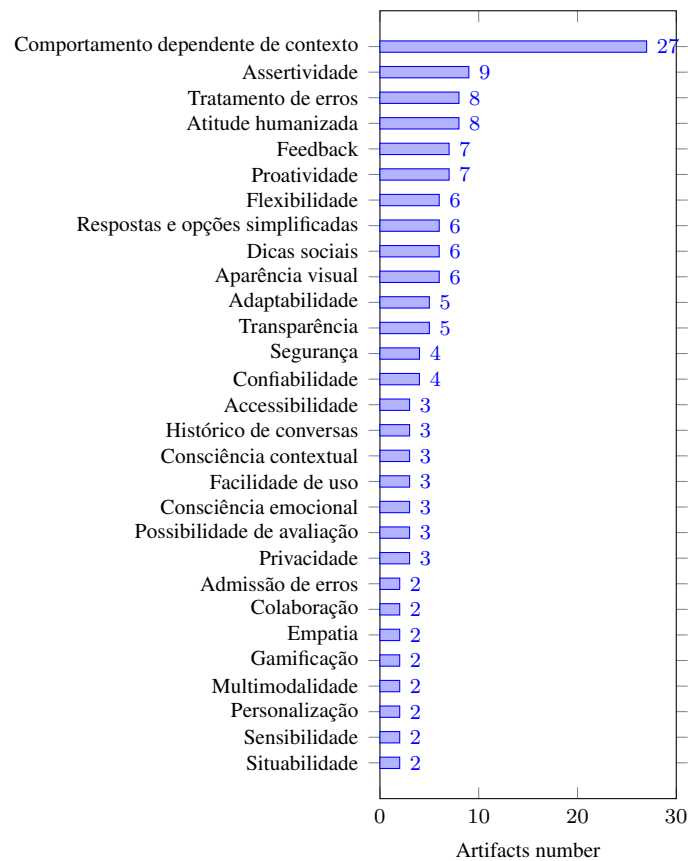


Figura 4.5: Requisitos conversacionais encontrados na literatura.

melhorar a boa experiência do usuário.

4.8 SUMÁRIO DO CAPÍTULO

O capítulo realiza uma Revisão Sistemática da Literatura (SLR) em artigos primários com foco em catalogar os artefatos existentes na literatura para documentação de requisitos em chatbots, e especificamente quais destes abordam requisitos conversacionais, e qual o conteúdo destes requisitos. Identificou-se que os artefatos mais recorrentes - diagramas UML, protótipos, fluxos conversacionais, tabelas de requisitos e cenários - são também utilizados para abordar requisitos tradicionais e conversacionais, com exceção dos diagramas UML. Dentre os artefatos mais recorrentes, os protótipos, tabelas de requisito e os fluxos conversacionais são os que proporcionalmente mais abordaram requisitos conversacionais. A partir do conteúdo, foi observado que os requisitos conversacionais mais abordados na literatura e registrados nestes artefatos foram, respectivamente, o comportamento dependente de contexto, a assertividade, o tratamento de erros e a atitude humanizada. O estudo demonstra que a presença de artigos com documentação dos requisitos de chatbots é recente na literatura, e que dentre os artefatos mais utilizados para a documentação de chatbots, alguns são similares a outros tipos de software (diagramas UML, protótipos e cenários), enquanto alguns são bem específicos do contexto conversacional (fluxos de conversação, árvores de decisão e tabelas de requisito, por exemplo).

5 PROPOSTA E VALIDAÇÃO DE DOCUMENTAÇÃO

5.1 PROPOSTA DE META-MODELO DE ARTEFATO CONVERSACIONAL

Com base nas características encontradas nas SLRs dos capítulos 3 e 4, foi possível identificar padrões que nos permitissem propor um meta-modelo para documentação de requisitos conversacionais de chatbots. O metamodelo é orientado pelas necessidades observadas anteriormente e, principalmente, pelas lacunas existentes, a exemplo da representação de requisitos, definição de métricas para teste, e exemplo de conversação em um cenário definido. Guiados pelos artefatos mais populares para o registro de requisitos conversacionais em chatbots (figura 4.4) - protótipos, tabela de requisitos, fluxos conversacionais, cenários e diagramas - desenvolvemos um *template* que pode ser adaptado à realidade dos profissionais que aplicam a engenharia de requisitos em softwares de chatbot. Nossas premissas para construir o metamodelo foram as seguintes:

1. Não pretende substituir todos os artefatos existentes; O intuito é ser um artefato robusto que reduza a necessidade de vários documentos.
2. Deve ser adaptável a diversos formatos, apresentando as mesmas informações importantes independente da visualização (tabela, protótipo, fluxo conversacional, diagrama UML, mapa mental);
3. Deve ser flexível, de acordo com as necessidades dos profissionais;
4. Deve promover pontos de conexão entre requisitos e elementos conversacionais;
5. Deve ser capaz de representar a aparência visual;
6. Deve fornecer um local para descrever os requisitos;
7. Deve mostrar uma sequência de interações entre o chatbot e o usuário;

A fim de cumprir as premissas descritas acima, foram definidas as partes que o metamodelo deveria apresentar, e elementos opcionais que auxiliam na compreensão dos requisitos expostos. O metamodelo proposto é composto por 6 partes obrigatórias e 3 partes opcionais.

O título, descrição do cenário, mensagem do chatbot, mensagem do usuário, requisito e anotação do usuário são a mínima composição do metamodelo. A partir destas, é possível apresentar e exemplificar um cenário, bem como evidenciar a ligação entre pontos das mensagens e requisitos conversacionais (tabela 5.1).

As partes opcionais incluem um épico, um número sequencial de turnos (explicita a ordem das interações e facilita a citação de mensagens anteriores). Adicionalmente, decidimos incluir um campo opcional para KPIs (310) (ou critérios de aceitação), vinculado a cada requisito, devido à subjetividade comum dos requisitos conversacionais (43). O formato do modelo pode ser adaptado para diferentes visualizações, como tabela (Tabela 5.2), protótipo (Figura 5.2) e diagrama UML (Figura 5.1).

Componente	Obrigatório ou opcional	Descrição
Título	Obrigatório	Apresenta a situação específica que será documentada.
Épico	Opcional	Caso o cenário específico seja parte de uma situação maior, o épico representa o cenário-pai.
Descrição do cenário	Obrigatório	Descreve um resumo da situação a ser representada. Pode conter informações que os times de software considerem importantes, como pré-condições.
Número sequencial	Opcional	Representa a posição de uma fala específica na sequência de interações entre o chatbot e o usuário. Apesar de ser opcional, é recomendado para facilitar a visualização e citação de mensagem prévia.
Mensagem do chatbot	Obrigatório	Mostra a resposta ou interação do chatbot.
Mensagem do usuário	Obrigatório	Mostra a resposta ou interação do usuário.
Requisito	Obrigatório	Explicação do requisito.
KPI ou critério de aceite	Opcional	Descrição de qual forma o requisito será considerado aceite. Importante principalmente para os requisitos subjetivos durante uma interação conversacional.
Anotação do usuário	Obrigatório	Destaque de elementos relevantes na resposta do usuário.

Tabela 5.1: Composição do metamodelo de artefato para documentação de requisitos conversacionais.

5.1.1 Representações do Mapa conversacional integrado

A fim de testar e validar se o metamodelo proposto é flexível para encaixar em diversas representações visuais, utilizamos as partes estruturais do metamodelo em três tipos de representação diferente: tabelas, diagramas UML e protótipos. As representações foram escolhidas pelas diferenças intrínsecas, e pela expressividade na documentação de requisitos, conforme apresentado nos Capítulos 3 e 4.

5.1.1.1 Tabela

A representação em tabela é a mais acessível das visualizações, pois pode ser feita em qualquer software que manipule planilhas ou texto. Nesta representação, porém, a aparência visual é menos enfatizada, portanto a fidelidade visual é baixa. Pode-se acrescentar imagens anexas na planilha, entretanto, optamos por não apresentar desta forma em nosso exemplo (Tabela 5.2).

Título*: Oferecendo auxílio help desk específico para um usuário já logado		Épico: Help desk		
Descrição*: O usuário já está logado na aplicação. O usuário inicia uma conversa ao fazer uma pergunta ao chatbot sobre o problema que está tendo. O chatbot responde tanto sobre a dúvida relacionada ao problema quanto a solução para cada possível problema. O usuário se sente satisfeito com a resposta e agradece ao chatbot. O chatbot responde educadamente e deixa o usuário confortável para futuras interações.				
Sequência	Turno*	Conteúdo da mensagem*	Requisito*	KPI
1	Usuário	Bom dia, a tecla do ponto de interrogação do meu computador não está funcionando, o que pode ser?	cumprimenta; Faz uma pergunta sem "?". Pede ajuda.	
2	Chatbot	<p>Bom dia Joana, sinto muito pelo seu problema. Se uma chave de pergunta não estiver funcionando, pode ser por alguns motivos:</p> <ul style="list-style-type: none"> • Configuração: Verifique se a configuração do teclado no sistema operacional está correta; • Sujeira ou obstrução: Tente limpar o teclado com ar comprimido. <p>Se nenhuma das opções acima funcionar, considere ligar para uma assistência técnica.</p>	<p>1.Tratamento de erros: Entenda e responda à intenção do usuário, mesmo que a entrada contenha um ou mais erros de digitação.</p> <p>2.Personalização: O chatbot responde usando o nome do usuário.</p> <p>3.Empatia: O Chatbot reconhece que o usuário está passando por um momento ruim e expressa sua empatia.</p> <p>4.Proatividade: O Chatbot responde à pergunta e também fornece a solução.</p> <p>5.Direcionar a um especialista: Quando um usuário solicita uma resposta técnica, jurídica ou médica, no final da resposta deverá constar a orientação para entrar em contato com um profissional humano.</p>	<p>Apresentar um problema (técnico ou não) e receber um reconhecimento educado da situação.</p> <p>Faça uma pergunta técnica e receba uma resposta e uma solução.</p>
3	Usuário	Muito obrigado, funcionou!	Obrigado ao chatbot pela ajuda em uma tarefa bem sucedida.	
4	Chatbot	Fico feliz em ouvir isso! Se você tiver mais dúvidas ou precisar de mais ajuda, fique à vontade para perguntar!	6.Mostrar disponibilidade ao usuário: Encerrar conversa mostrando ao usuário que o chatbot ainda está disponível caso ele tenha mais dúvidas.	

Tabela 5.2: Mapa conversacional integrado na visualização de tabela.

5.1.1.2 Diagrama UML

Para a construção da representação em diagrama UML, utilizamos o Draw.io, um software online e gratuito com suporte para a notação UML. Porém isso pode ser feito em qualquer software com notações UML e é quase tão versátil quanto a representação de tabelas. A tabela de requisitos e KPIs no lado direito foi adaptada dos diagramas de classe UML. As mensagens foram adaptadas de objetos UML. Porém, formas geométricas simples podem ser utilizadas para compor o diagrama, independente da notação.

Além disso, a representação em diagrama (Figura 5.1) é mais versátil que a tabela e pode ser adaptada e incrementada com outros recursos, como fotos ou *emojis*. Também dá uma maior sensação de aparência visual do que as tabelas porque no lado esquerdo há a simulação de mensagens em balão é fácil de usar. O endereçamento dos requisitos pode ser feito destacando as frases com a cor atribuída ao requisito 5.1. Essa representação pode ser classificada como de média fidelidade, e assim como as tabelas, pode ser escalada para interações maiores.

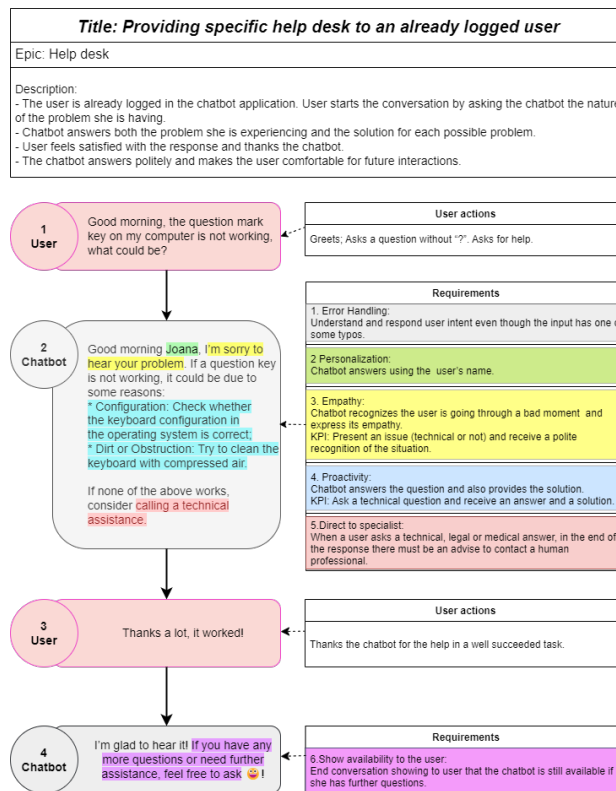


Figura 5.1: Representação do mapa conversacional integrado em diagrama UML.

5.1.1.3 Protótipo

Para a representação do protótipo (Figura 5.2), foi utilizado Figma. Figma também é uma ferramenta online gratuita mundial, usada para projetar e simular protótipos de softwares interativos com alta fidelidade. O processo de criação consistiu em escolher um dispositivo e representar uma conversa como um usuário a veria. Cada requisito recebe uma cor e é destacado conforme aparece na interação.

A representação do protótipo mostra comportamentos semelhantes ao diagrama, mas aborda diretamente a aparência visual e os requisitos de usabilidade, e também pode mostrar interações dinâmicas, pois podem ser configuradas no arquivo Figma, ou outros programas de prototipagem. A versão paga também permite a inclusão de recursos audiovisuais como vídeos ou áudio, o que pode ser uma opção mais completa caso o chatbot seja baseado em fala ou exija interação por vídeo. Em geral, os recursos audiovisuais raramente são representados na documentação de software (Figura 4.3).

Comparados aos artefatos mais utilizados para especificação de requisitos conversacionais encontrados em nossos estudos, os mapas conversacionais integrados aparecem como uma abordagem versátil e robusta, que comporta diferentes visualizações (Tabela 5.1). Cada uma das representações possui diferentes destaques, como a aparência visual oferecida pelos protótipos, por exemplo (Figura 5.2), a descrição detalhada dos requisitos encontrada nas tabelas de requisitos (Tabela 5.2), e a visualização sequencial global oferecida por diagramas, cenários ou mapas mentais UML (Figura 5.1).

O modelo foi proposto a partir das características de destaque de artefatos comuns aos profissionais da engenharia de software. O título e o épico, a saber, vêm de histórias de usuários, que são artefatos de



Figura 5.2: Representação do mapa conversacional integrado em formato de protótipo.

documentação populares para outros tipos de software. Eles ajudam a restringir solicitações grandes e inespecíficas, especialmente requisitos subjetivos não funcionais. A descrição abaixo é derivada de cenários e ajuda a compreender o quadro geral da situação. Pode ser utilizado como referência e primeiro ponto de comunicação entre os stakeholders e a equipe de desenvolvimento.

Os turnos conversacionais, seguidos ou não de número, são necessários para ilustrar diretamente como deve ser a interação na prática e podem servir de referência na fase de teste. Os turnos conversacionais são associados diretamente a requisitos conversacionais pelo mapa conversacional integrado pela evidência espacial e colorimétrica, e esta conexão é a principal contribuição deste modelo (tabela 5.3), uma vez que nenhum dos artefatos atuais aborda diretamente esta ligação.

Os requisitos podem apresentar ou não um KPI, pois é um campo opcional (tabela 5.1). No entanto, recomendamos que cada requisito tenha uma ou mais KPIs. Baseando-se nas *User Stories*, elas se comportariam como critérios de aceitação. Nossa proposta de KPI é semelhante ao trabalho de Nguyen et al (241), que adaptou casos de uso para registrar requisitos não-funcionais. A principal razão para esta proposta, brevemente explicada acima, é aumentar a importância das medidas de qualidade estabelecidas para requisitos subjetivos, uma vez que representam um desafio nos testes de software por não possuírem resultado bem definido (43).

Nosso metamodelo atualmente não inclui um campo específico para mostrar componentes ou entidades arquiteturais, como intenções e entidades. Eles não foram incluídos intencionalmente porque o metamodelo foi projetado para ser um documento de fácil utilização e melhorar a colaboração entre as partes interessadas e as equipes de desenvolvimento e teste. No entanto, este metamodelo deve ser avaliado pelos profissionais para determinar se está aderente ao seu propósito ou se as características devem ser alteradas, sendo passível de adaptação para cada time ou projeto.

Artefato	Aparência Visual	Descrição de requisitos	Sequência de interações	Conexão entre o requisito e elementos da fala
Mapa conversacional integrado	Sim	Sim	Sim	Sim
Fluxo Conversacional	Não	Não	Sim	Não
Árvore de decisão	Não	Não	Sim	Não
Cenário	Não	Não	Sim	Não
Protótipo	Sim	Não	Sim	Não
Diagrama UML	Não	Não	Sim	Não
Tabela de requisitos	Não	Sim	Não	Não

Tabela 5.3: Comparação entre mapa conversacional integrado e artefatos para especificação de chatbots.

Outra melhoria que pode ser implementada em trabalhos futuros é a divisão hierárquica de cenários dentro de épicos e contextos maiores, semelhante à visualização de ontologias, mapas mentais ou visualizações de árvores de decisão, melhorando a organização dinâmica. O metamodelo pode ser incluído em bibliotecas automatizadas para ser acessível em todo o mundo. Até mesmo a geração de uma visualização pode gerar automaticamente as outras. Existe também a possibilidade de utilizar IA e *machine learning* para prever os requisitos mais comuns do projeto e fazer uma sugestão prévia com base na documentação anterior.

É importante destacar que o metamodelo pode ter a ênfase aumentada em elementos diferentes a depender da visualização. Por exemplo, as tabelas tendem a enfatizar a descrição de requisitos; já os protótipos enfatizam o aspecto visual de apresentação da aplicação. Outro ponto importante é que a carga cognitiva (esforço necessário para leitura e compreensão) pode aumentar conforme a extensão do diálogo, e, portanto, este é recomendado para interações mais curtas ou fragmentos de interação.

5.2 VALIDAÇÃO DO MAPA CONVERSACIONAL INTEGRADO

Neste tópico será apresentada a fase de avaliação e validação do mapa conversacional integrado. Após a criação do artefato apresentado na seção (5.1), bem como guia de utilização, é necessário observar o comportamento de diferentes tipos de usuário, que representam atores distintos, com o artefato, e observar os pontos de evolução, correção ou melhoria.

Para tal, foi realizada a validação com profissionais experientes em análise de requisitos de chatbot, durante duas fases de simulação. A primeira fase consistiu na apresentação de um exemplo de fluxo conversacional, no qual os profissionais tinham que especificar requisitos sem um formato de documentação definido. Na segunda fase, o Mapa Conversacional integrado é apresentado e os profissionais são convidados a utilizá-lo para documentar requisitos.

Para o planejamento da pesquisa, sua execução e análise das respostas, utilizou-se a diretriz proposta por Pfleeger e Kitchenham (311). O questionário online foi desenvolvido no software gratuito Google Forms e divulgado organicamente nas redes sociais e entre profissionais da área.

Para a primeira fase, os participantes foram convidados a fornecer seus endereços de e-mail, apenas para que em outro momento pudéssemos contatá-los novamente para solicitar participação na segunda fase da pesquisa. Somente os participantes que concordaram em participar voluntariamente e concordaram em usar seus dados anonimamente para publicações foram levados em consideração para geração e análise dos resultados. Além disso, a pesquisa online permitiu ao participante desistir a qualquer momento, sem qualquer desconforto, pois o processo não requer supervisão. Ambos os formulários estiveram disponíveis de 18 de março a 15 de junho de 2024 (Tabela 5.4).

Os participantes receberam um exemplo de fluxo conversacional e foram convidados a criar documentação de requisitos conversacionais com base no fluxo recebido, sem modelo definido. Após a elaboração de uma proposta de documentação de requisitos, foi enviado outro formulário com acesso às instruções de utilização do Mapa Conversacional Integrado e três *templates* (tabela, diagrama e protótipo), podendo escolher um de sua preferência para continuar.

Ao receberem acesso às instruções e modelos, os profissionais foram novamente convidados a realizar a documentação dos requisitos, utilizando o modelo proposto. Por fim, na entrega da documentação de requisitos preenchida, foram feitas perguntas sobre a percepção de uso do artefato. Esperava-se que as duas fases do questionário apresentadas a seguir durassem entre 15 e 20 minutos para serem respondidas.

5.3 RESULTADOS DA VALIDAÇÃO DO MODELO DE ARTEFATO CONVERSACIONAL

5.3.1 Survey

A fase 1 da pesquisa teve 9 respostas até o dia 17/08/2024. 2 dos 9 profissionais da fase 1 declararam não possuir experiência com análise de requisitos para chatbots, conforme mostrado na Tabela 5.5. Ao final da pesquisa 1, todos os participantes concordaram em participar da fase 2. Como foram realizadas em momentos separados para evitar viés de memória dos praticantes, nem todos os participantes que responderam à fase 1 responderam à fase 2, resultando em 5 respostas.

Quanto à escolha do formato pelos usuários, na fase 1, 4 profissionais escreveram a especificação dos requisitos em texto simples, utilizando tópicos para organizar o texto. Um dos participantes escreveu histórias de usuários para especificação, outro documentou os requisitos em formato de caso de uso e um usou tópicos e diagramas para documentação. Na fase 2, 3 profissionais usaram o modelo de tabela e 2 profissionais utilizaram o modelo de protótipo para documentar os requisitos.

A figura 5.3 mostra as respostas dos participantes para cada questão da segunda fase da pesquisa. Uma maior concordância nas questões 1,3,5,6 e 7 demonstra satisfação com o metamodelo, uma vez que as afirmações são positivas. As sentenças 2 e 4 são negativas e, portanto, quando um usuário concorda, significa que este é um ponto para investigação e melhoria.

A maior satisfação é observada nos itens 1 e 6, seguidos pelos itens 5 e 7. Todas as respostas foram positivas para os itens 1,5,6 e 7. O item 1 aborda se o modelo é percebido como fácil de usar. O item 2

Fase	ID	Questão
Questionário 1	Q1	Declaro que fui informado e esclarecido verbalmente sobre este documento, compreendendo todos os termos acima expostos, e que aceito voluntariamente participar deste estudo.
	Q2	Realizo ou realizei análise de requisitos para chatbots
	Q3	Nome completo
	Q4	E-mail (necessário para entrar em contato na fase 2)
	Q5	Há quanto tempo você atua ou atuou na área de análise de requisitos?
	Q6	Envie sua especificação de requisitos aqui
	Q7	Ao submeter este formulário, concordo em ser contatado para a segunda fase do questionário.
Questionário 2	Q1	Declaro que fui informado e esclarecido verbalmente sobre este documento, compreendendo todos os termos acima expostos, e que aceito voluntariamente participar deste estudo.
	Q2	Nome completo
	Q3	Envie aqui a especificação de seus requisitos usando o Mapa Conversacional Integrado.
	Q4	Sobre a documentação de requisitos: 1. O modelo proposto na fase 2 foi fácil de usar 2. O modelo que propus era mais fácil de usar 3. O modelo proposto na fase 2 é escalável 4. O modelo proposto é demorado para documentar 5. Eu usaria o modelo novamente 6. O modelo proposto é fácil de aprender 7. Estou satisfeito com o modelo proposto na fase 2
	Q5	Obrigado por preencher nosso formulário. Este é um espaço para observações adicionais, sugestões e/ou críticas, se houver.

Tabela 5.4: Descrição dos questionários

compara o modelo proposto com o formato que o participante escolheu anteriormente para a documentação de requisitos do chatbot, onde 3 participantes consideram o modelo mais fácil de usar (Figura 5.3).

O item 3 aborda a escalabilidade do modelo e recebeu respostas mistas. O item 4 é um ponto para investigação, pois todas as respostas foram neutras. Este item específico compara o tempo necessário para

ID	Experiência com requisitos de chatbot?	Tempo de experiência
P1	Não	-
P2	Sim	Mais de 5 anos
P3	Sim	Mais de 5 anos
P4	Sim	Até 1 ano
P5	Sim	1 a 3 anos
P6	Sim	Até 1 ano
P7	Sim	Até 1 ano
P8	Não	-
P9	Sim	1 a 3 anos

Tabela 5.5: Descrição anonimizada dos participantes.

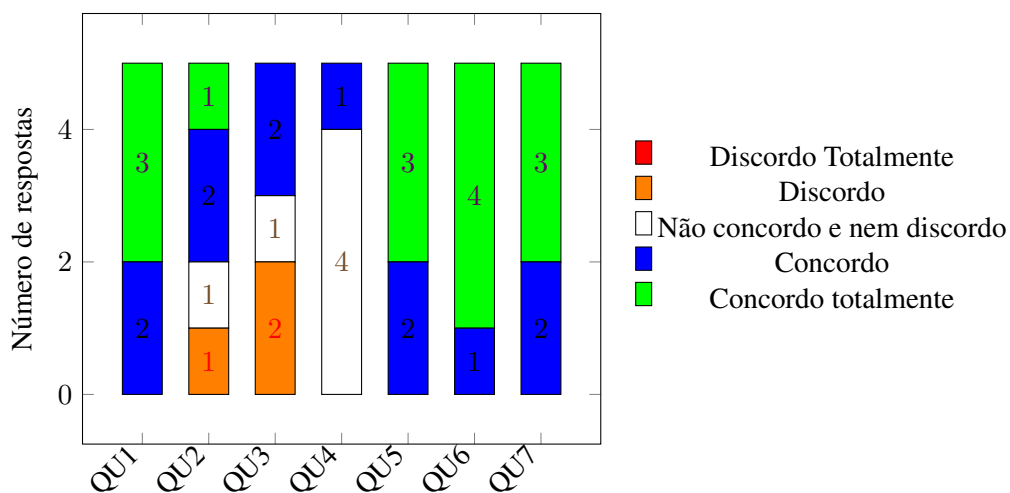


Figura 5.3: Respostas por questão na escala *likert*.

documentar requisitos com e sem modelo. A partir dos resultados, pode-se inferir que os praticantes não percebem muita diferença. O item 5 pergunta ao participante se ele usaria o modelo novamente. Todas as respostas do item 5 foram positivas. Os itens 6 e 7 também receberam apenas respostas positivas, e estão relacionados à facilidade de aprendizagem e à satisfação com o artefato (Figura 5.3). Dois participantes forneceram sugestões adicionais (tabela 5.6), em relação à completude do modelo e sua adequação em cenários maiores de conversação.

5.3.2 Utilização orgânica

Como mencionado anteriormente, os *templates* foram disponibilizados de forma editável aos participantes em 3 formatos: Tabela, diagrama e protótipo. O protótipo foi disponibilizado em Figma Community, o repositório *online* no qual designers e desenvolvedores publicam gratuitamente seus projetos, e a

ID	Sugestão adicional do participante
P7	"Acho que o modelo disponibilizado poderia vir com a conversa de exemplo preenchida."
P9	"O modelo é um pouco demorado para documentar, mas parece bastante fácil de explicar para outras pessoas. Uma sugestão é tornar o modelo mais flexível para diálogos longos, e permitir pontos de decisão nas conversas para saber qual caminho tomar."

Tabela 5.6: Sugestões extras dos participantes para o modelo.

ferramenta permite que sejam utilizados como *template* por outros usuários. Em acesso realizado no dia 22 de agosto de 2024, após 3 meses de disponibilização, o *template* conta com mais de 100 acessos orgânicos, visto que não foi realizada divulgação da publicação.

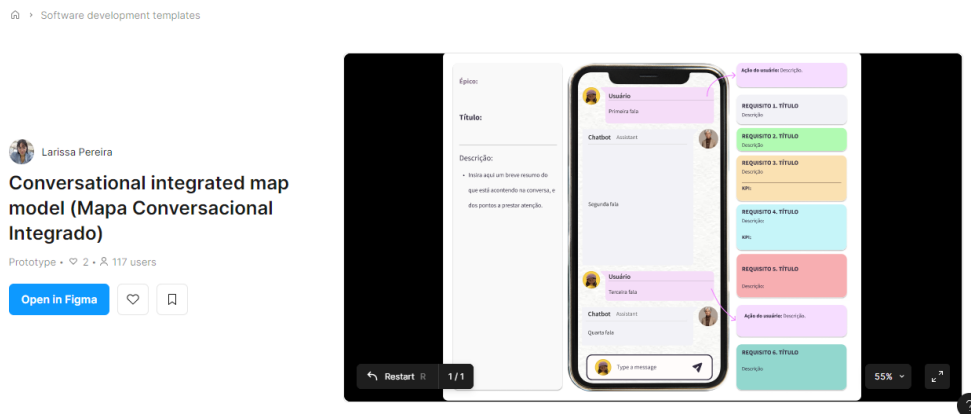


Figura 5.4: Desempenho do mapa conversacional integrado na comunidade do Figma.

5.4 DISCUSSÃO

O modelo proposto compreende as principais características dos artefatos mais utilizados, vistos no capítulo 3 e 4. O título e o épico vêm de histórias de usuários, que são artefatos de documentação populares para outros tipos de software. Eles ajudam a restringir solicitações grandes e inespecíficas, especialmente requisitos subjetivos não funcionais. Aliados aos turnos conversacionais, os requisitos associados são a principal contribuição deste modelo, uma vez que os artefatos atuais não abordam diretamente esta ligação.

Os requisitos podem apresentar ou não um KPI, pois é um campo opcional 5.1. No entanto, recomendamos que cada requisito tenha um ou mais KPIs. Nas histórias de usuários, elas se comportariam como critérios de aceitação. Nossa proposta de KPI é semelhante a Nguyen et al. (241) trabalho, que adaptou casos de uso para requisitos não funcionais (241). A principal razão para esta proposta explicada brevemente acima é aumentar a importância das medidas de qualidade estabelecidas para requisitos subjetivos, uma vez que representam um desafio nos testes e o sucesso ou o fracasso são difíceis de abordar (43, 291).

Em trabalhos futuros, uma melhoria a se implementar é a divisão hierárquica de cenários dentro de épicas e contextos maiores, semelhante à visualização de ontologias, mapas mentais ou visualizações de árvores de decisão, melhorando a organização dinâmica. O metamodelo pode ser incluído em bibliotecas automatizadas para ser acessível mundialmente. Até mesmo a geração de uma visualização pode gerar automaticamente as outras. Existe também a possibilidade de utilizar IA e *machine learning* para prever os requisitos mais comuns do projeto e fazer uma sugestão prévia com base na documentação anterior. As sugestões feitas pelos participantes, a saber, disponibilização dos *templates* com modelos pré-prontos e adequação para conversas maiores também serão incorporados na criação da segunda versão do modelo.

5.5 MELHORIAS A SEREM IMPLEMENTADAS NA VERSÃO 2.0

Na versão 2.0, objetiva-se a implementação de melhorias sugeridas após a observação do uso e validação dos usuários, segundo nossa primeira premissa, de criar um artefato robusto e que minimize a necessidade de vários documentos. As melhorias a serem implementadas são:

- Primeiramente, o campo de métricas, que abrange KPIs e critérios de aceite, será tornado obrigatório, garantindo que todas as entregas sejam acompanhadas por indicadores claros.
- Além disso, será disponibilizado um template com um exemplo já preenchido, facilitando a compreensão e o uso do modelo, como sugerido pelos participantes 5.6.
- Também será promovida uma maior flexibilidade para diálogos longos, permitindo interações mais dinâmicas e naturais, possivelmente pela criação de um template que seja mais similar ao Fluxo Conversacional, artefato citado previamente.
- Por fim, a inclusão de pontos de tomada de decisão no modelo auxiliará os usuários a visualizar um contexto mais abrangente de possibilidades.

5.6 LIMITAÇÕES E AMEAÇAS À VALIDADE

Uma limitação importante a destacar é a potencial discrepância entre as descrições e discussões encontradas na literatura acadêmica e as práticas reais no desenvolvimento e implementação de chatbots. A literatura pode não captar totalmente as complexidades e desafios enfrentados pelos profissionais da área, o que sugere a necessidade de complementar os estudos bibliográficos com investigações empíricas.

Também realizamos a validação do modelo com 5 profissionais que tiveram contato com a necessidade de chatbots durante o desenvolvimento do software. A divisão dos questionários em 2 fases e a dupla restrição proposta durante a seleção impediram um maior número de participantes.

5.6.1 Metamodelo proposto

O metamodelo proposto neste trabalho foi baseado em pesquisas anteriores, mas ainda apresenta alguns desafios. Primeiro, algumas variações de representação tendem a enfatizar ou não alguns campos específicos do modelo. Por exemplo, na Tabela, o remetente da mensagem (usuário ou chatbot) não é tão visível como no diagrama ou na visualização do protótipo. O segundo ponto é que este modelo é recomendado para cenários e conversas curtas. Interações mais longas com tamanha quantidade de informações podem trazer uma alta carga cognitiva na visualização. Algumas outras ações podem ser feitas para superar esse problema, como incluir uma opção para ocultar o excesso de informações quando desejado.

Uma observação importante é que o metamodelo deve ser avaliado entre os profissionais envolvidos na criação e desenvolvimento de projetos de chatbot, a fim de avaliar se a solução proposta atende às necessidades dos profissionais de software. Também deve ser avaliado se a documentação é compreensível pelos *stakeholders* (202) e se pode ser utilizada como documentação oficial para desenvolvimento de chatbot. Outra preocupação é que o processo de criação pode ser muito manual, pois há a necessidade de destacar pontos específicos dentro da conversa e apresentá-los externamente.

5.6.2 Pesquisa

A validade interna deste trabalho é ameaçada principalmente pela natureza on-line e não supervisionada da nossa pesquisa. Não tivemos controle sobre os participantes enquanto eles respondiam, mas disponibilizamos nossos dados de contato caso tivessem alguma dúvida. Do lado positivo, o ambiente não supervisionado significava que os participantes não eram pressionados a responder e podiam desistir a qualquer momento se não estivessem comprometidos, o que nos leva a acreditar que aqueles que responderam ao inquérito forneceram respostas significativas. A validade externa é ameaçada pelo fato de que nossa pesquisa incluiu apenas profissionais brasileiros, limitando a generalização de nossos resultados. Embora o tamanho da nossa amostra seja pequeno, vale ressaltar que existem vários outros estudos na literatura que também conduziram pesquisas com um pequeno número de participantes.

A avaliação geral do mapa conversacional integrado pelos participantes foi positiva em relação ao modelo. Foi considerado fácil de usar, fácil de aprender e todos os participantes questionados que utilizariam o modelo novamente (Figura 5.3). No entanto, estudos futuros devem investigar se este modelo e os artefatos existentes são suficientes para abordar todos os aspectos complexos dos requisitos conversacionais específicos dos chatbots ou requerem maior adaptação. Outra questão remanescente seria a priorização e teste de requisitos subjetivos, um desafio adicional colocado pelos chatbots.

Os próximos passos deste trabalho deverão ser promover um guia para utilização de artefatos, publicar os demais *templates* na internet e melhorar sua facilidade de uso, automação e escalabilidade. Além disso, estudos futuros devem investigar como projetar testes e métricas para requisitos subjetivos.

5.7 SUMÁRIO DO CAPÍTULO

A validação do modelo para documentação de requisitos conversacionais de chatbot foi realizada com participantes que atuam ou atuaram na área de requisitos de software, especificamente em projetos com chatbots. Para a validação do modelo, os participantes foram convidados a especificar os requisitos sem modelo definido na fase 1, e utilizando a nossa proposta de modelo na fase 2. Em seguida, foram questionados em relação à diferença de usabilidade e percepções entre um e outro. De forma geral, os participantes avaliaram positivamente o mapa conversacional integrado, e a utilização espontânea por mais de 100 usuários na comunidade online do Figma evidenciou que o modelo possui potencial prático de aplicação.

6 DISCUSSÃO E CONCLUSÃO

Neste trabalho, investigamos as diferenças de chatbots em relação a outros softwares, com foco na diferença de requisitos de software presentes, especificação e documentação de requisitos, e em quais artefatos esta documentação é realizada. Com base na investigação, foi elaborado um novo artefato, cuja ênfase reside na documentação de requisitos conversacionais.

Em nossa primeira pergunta de pesquisa, observamos que a maior diferença dos chatbots em relação aos outros software é a sua capacidade de interagir com os usuários através de linguagem natural e um fluxo conversacional. A interação entre o software e os usuários é antropomórfica, em sua maioria. Em relação aos requisitos de software, chatbots costumam apresentar requisitos conversacionais e emocionais, além dos requisitos funcionais e não- funcionais já conhecidos em outros tipos de softwares.

Na segunda fase deste trabalho, identificamos que a documentação de requisitos em linguagem natural é relatada na literatura em uma variedade de artefatos utilizados para essa finalidade, com ênfase em diagramas, casos de uso, modelos conceituais, histórias de usuário e protótipos, aplicados de forma distinta para requisitos funcionais e não-funcionais. Pontua-se que os diagramas e casos de uso apresentaram expressiva quantidade de modelos disponíveis para utilização em relação aos outros tipos de artefato 4. Diagramas, casos de uso, cenários e protótipos de forma geral são utilizados nos dois tipos de requisitos, a depender do teor. Entretanto, histórias de usuários e artefatos derivados são mais recomendadas para requisitos funcionais e possuem limitações para requisitos não-funcionais.

Em seguida, ao comparar a tendência de documentação de requisitos acima com a documentação de requisitos conversacionais de chatbots, identificou-se que artefatos como protótipos, cenários e diagramas UML também estão entre os mais utilizados para chatbots. Entretanto, os diagramas UML não são utilizados com frequência para requisitos conversacionais. Dentre os artefatos mais recorrentes, os protótipos, tabelas de requisito e os fluxos conversacionais são os que proporcionalmente mais abordaram requisitos conversacionais.

Com relação aos requisitos conversacionais, o comportamento dependente de contexto, a assertividade, o tratamento de erros e a atitude humanizada são os mais presentes nos estudos analisados. Também foi visto que a literatura contém artefatos praticamente especializados para a representação de requisitos de chatbots, visto que não aparecem comumente como artefatos de documentação em outros softwares, como fluxos de conversação, árvores de decisão e tabelas de requisitos.

As características encontradas nos capítulos 3 e 4, culminaram na proposta de um modelo de documentação adaptável com enfoque em situações conversacionais, visando preencher lacunas existentes nos artefatos, a exemplo da representação de requisitos, definição de métricas para teste, e exemplo de conversação em um cenário definido. Com base nas características de destaque dos artefatos citados acima, o modelo visa manter a especificação de requisitos conversacionais específica, intuitiva, criando uma relação direta entre o requisito e a fala do chatbot. Também visa a utilização durante várias etapas do ciclo de software, ou seja, durante a elicitação e especificação de requisitos, desenvolvimento e teste de software, visando reduzir e/ou amenizar a subjetividade durante o processo.

O modelo foi exemplificado nos formatos de tabela, diagrama e protótipo, utilizando linguagem natural. Em seguida, ele foi compartilhado com profissionais de tecnologia que possuíam, duplamente, experiência com requisitos de software e com o processo de desenvolvimento de chatbots, para a coleta de opiniões e investigação da percepção destes profissionais acerca do modelo. De forma geral, os participantes avaliaram positivamente a usabilidade, curva de aprendizagem e aplicação do mapa conversacional integrado.

Contudo, é importante pontuar que o trabalho analisou somente a literatura para a proposição do metamodelo, e esta pode ser discrepante de práticas reais de profissionais. Além disso, o campo de chatbots está em rápida expansão, o que sugere a necessidade de complementar os estudos bibliográficos com investigações empíricas frequentes. Embora a validação tenha sido realizada com profissionais experientes na área, a especificidade do recrutamento de participantes restringiu o número de respostas recebidas, de forma que mais testes são necessários para atestar a contribuição do modelo para as práticas da engenharia de requisitos.

A proposta do modelo inicial foi submetida à validação e ainda possui espaço para evoluções. Nossos próximos passos incluem a aplicação de pesquisas adicionais com grupos mais abrangentes de participantes, a saber, um survey de artefatos utilizados para documentar especificamente requisitos conversacionais, e a simulação de utilização de um guia online de utilização do template para analistas de requisitos e interessados em requisitos conversacionais.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 KHAN, R.; DAS, A. *Build Better Chatbots*. Apress, 2018. ISBN 9781484231111. Disponível em: <<http://dx.doi.org/10.1007/978-1-4842-3111-1>>.
- 2 ADAMOPOULOU, E.; MOUSSIADES, L. Chatbots: History, technology, and applications. *Machine Learning with Applications*, v. 2, p. 100006, 2020. ISSN 2666-8270. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2666827020300062>>.
- 3 BRAUN, D.; MATTHES, F. Towards a framework for classifying chatbots. In: FILIPE, J.; SMIALEK, M.; BRODSKY, A.; HAMMOUDI, S. (Ed.). *Proceedings of the 21st International Conference on Enterprise Information Systems, ICEIS 2019, Heraklion, Crete, Greece, May 3-5, 2019, Volume 1*. SciTePress, 2019. p. 496–501. Disponível em: <<https://doi.org/10.5220/0007772704960501>>.
- 4 KURTANOVIĆ, Z.; MAALEJ, W. Automatically classifying functional and non-functional requirements using supervised machine learning. In: IEEE. *2017 IEEE 25th International Requirements Engineering Conference (RE)*. [S.l.], 2017. p. 490–495.
- 5 CANEDO, E. D.; CALAZANS, A. T. S.; SILVA, G. R. S.; MASSON, E. T. S. ICT practitioners' perception of working from home during the covid-19 pandemic: Exploring gender differences. In: MAIA, M. de A.; DORÇA, F. A.; ARAÚJO, R. D.; FLACH, C. von; NAKAGAWA, E. Y.; CANEDO, E. D. (Ed.). *SBES 2022: XXXVI Brazilian Symposium on Software Engineering, Virtual Event Brazil, October 5 - 7, 2022*. ACM, 2022. p. 47–57. Disponível em: <<https://doi.org/10.1145/3555228.3555248>>.
- 6 WEIZENBAUM, J. ELIZA - A computer program for the study of natural language communication between man and machine (reprint). *Commun. ACM*, v. 26, n. 1, p. 23–28, 1983. Disponível em: <<https://doi.org/10.1145/357980.357991>>.
- 7 MANIOU, T.; VEGLIS, A. 'selfie journalism': Current practices in digital media. *Studies in Media and Communication*, v. 4, n. 1, 2016. Disponível em: <<https://redfame.com/journal/index.php/smc/article/view/1637>>.
- 8 SEMIGRAN, H. L.; LINDER, J. A.; GIDENGIL, C.; MEHROTRA, A. Evaluation of symptom checkers for self diagnosis and triage: audit study. *BMJ*, BMJ Publishing Group Ltd, v. 351, 2015. Disponível em: <<https://www.bmj.com/content/351/bmj.h3480>>.
- 9 SIDNER, C. L.; KIDD, C. D.; LEE, C.; LESH, N. Where to look: a study of human-robot engagement. In: VANDERDONCKT, J.; NUNES, N. J.; RICH, C. (Ed.). *Proceedings of the 9th International Conference on Intelligent User Interfaces, IUI 2004, Funchal, Madeira, Portugal, January 13-16, 2004*. ACM, 2004. p. 78–84. Disponível em: <<https://doi.org/10.1145/964442.964458>>.
- 10 HILDEBRAND, C.; BERGNER, A. Ai-driven sales automation: Using chatbots to boost sales. *NIM Marketing Intelligence Review*, v. 11, n. 2, p. 36–41, 2019. Disponível em: <<https://doi.org/10.2478/nimmir-2019-0014>>.
- 11 HU, T.; XU, A.; LIU, Z.; YOU, Q.; GUO, Y.; SINHA, V.; LUO, J.; AKKIRAJU, R. Touch your heart: A tone-aware chatbot for customer care on social media. *CoRR*, abs/1803.02952, 2018. Disponível em: <<http://arxiv.org/abs/1803.02952>>.
- 12 DANIEL, F.; MATERA, M.; ZACCARIA, V.; DELL'ORTO, A. Toward truly personal chatbots: on the development of custom conversational assistants. In: NEZHAD, H. R. M.; MIKKILINENI, R.; BENATALLAH, B.; CASATI, F.; DUSTDAR, S.; DODIG-CRNKOVIĆ, G.; MOS, A. (Ed.).

- Proceedings of the 1st International Workshop on Software Engineering for Cognitive Services, SE4COG@ICSE 2018, Gothenburg, Sweden, May 28-2, 2018.* ACM, 2018. p. 31–36. Disponível em: <<https://doi.org/10.1145/3195555.3195563>>.
- 13 BUKHARI, S. S. A.; HUMAYUN, M.; SHAH, S. A. A.; JHANJHI, N. Improving requirement engineering process for web application development. In: *2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*. [S.l.: s.n.], 2018. p. 1–5.
- 14 BROOKS, F.; KUGLER, H. *No silver bullet*. [S.l.]: April, 1987.
- 15 CHRISTEL, M. G.; KANG, K. C. *Issues in requirements elicitation*. [S.l.]: Carnegie Mellon University, Software Engineering Institute, 1992.
- 16 KOTONYA, G.; SOMMERVILLE, I. *Requirements engineering: processes and techniques*. [S.l.]: Wiley Publishing, 1998.
- 17 BERENBACH, B.; PAULISH, D. J.; KAZMEIER, J.; RUDORFER, A. *Software & systems requirements engineering: in practice*. [S.l.]: McGraw-Hill Education, 2009.
- 18 CHRISTEL, M.; KANG, K. *Issues in Requirements Elicitation*. Pittsburgh, PA, 1992. Disponível em: <<http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=12553>>.
- 19 PACHECO, C. L.; GARCÍA, I. A.; REYES, M. Requirements elicitation techniques: a systematic literature review based on the maturity of the techniques. *IET Softw.*, v. 12, n. 4, p. 365–378, 2018. Disponível em: <<https://doi.org/10.1049/iet-sen.2017.0144>>.
- 20 BARROS, R. C. A importância da gestão de requisitos para projetos de desenvolvimento de software. 2018.
- 21 MESQUITA, R.; SILVA, G. R. S.; CANEDO, E. D. On the experiences of practitioners with requirements elicitation techniques. In: *Proceedings of the XXXVII Brazilian Symposium on Software Engineering, SBES 2023, Campo Grande, Brazil, September 25-29, 2023*. ACM, 2023. p. 442–451. Disponível em: <<https://doi.org/10.1145/3613372.3613410>>.
- 22 HAINDL, P.; PLÖSCH, R. Tailoring stakeholder interests to task-oriented functional requirements. *CoRR*, abs/2201.06567, 2022. Disponível em: <<https://arxiv.org/abs/2201.06567>>.
- 23 THEUNISSEN, T.; HEESCH, U. van; AVGERIOU, P. A mapping study on documentation in continuous software development. *Inf. Softw. Technol.*, v. 142, p. 106733, 2022. Disponível em: <<https://doi.org/10.1016/j.infsof.2021.106733>>.
- 24 AGHAJANI, E.; NAGY, C.; VEGA-MÁRQUEZ, O. L.; LINARES-VÁSQUEZ, M.; MORENO, L.; BAVOTA, G.; LANZA, M. Software documentation issues unveiled. In: ATLEE, J. M.; BULTAN, T.; WHITTLE, J. (Ed.). *Proceedings of the 41st International Conference on Software Engineering, ICSE 2019, Montreal, QC, Canada, May 25-31, 2019*. IEEE / ACM, 2019. p. 1199–1210. Disponível em: <<https://doi.org/10.1109/ICSE.2019.00122>>.
- 25 PRESSMAN, R. S. *Engenharia de Software. Quinta edição*. [S.l.]: McGraw-Hill, 2001.
- 26 ANANJEVA, A.; PERSSON, J. S.; BRUUN, A. Integrating UX work with agile development through user stories: An action research study in a small software company. *J. Syst. Softw.*, v. 170, p. 110785, 2020. Disponível em: <<https://doi.org/10.1016/j.jss.2020.110785>>.
- 27 MANIFESTO, A. Disponível em: <<http://agilemanifesto.org/>>. 2001. *Acesso em*, v. 19, 2023.

- 28 ROBIOLO, G.; SCOTT, E.; MATALONGA, S.; FELDERER, M. Technical debt and waste in non-functional requirements documentation: An exploratory study. In: FRANCH, X.; MÄNNISTÖ, T.; MARTÍNEZ-FERNÁNDEZ, S. (Ed.). *Product-Focused Software Process Improvement*. Cham: Springer International Publishing, 2019. p. 220–235.
- 29 SILVA, G. R. S.; CANEDO, E. D. Towards user-centric guidelines for chatbot conversational design. *CoRR*, abs/2301.06474, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2301.06474>>.
- 30 QASIM, M. M.; AHMAD, M.; OMAR, M. Persuasive strategies in mobile healthcare: A systematic literature review. *Journal of Engineering and Applied Sciences*, v. 12, p. 8706–8713, 2017. Disponível em: <<https://medwelljournals.com/abstract/?doi=jeasci.2017.8706.8713>>.
- 31 ALZAYED, A.; AL-HUNAIYYAN, A. A bird's eye view of natural language processing and requirements engineering. *International Journal of Advanced Computer Science and Applications*, Science and Information (SAI) Organization Limited, v. 12, n. 5, 2021.
- 32 HE, Z. From ELIZA to chatgpt: The evolution of chatbots in public health. *XRDS*, v. 29, n. 3, p. 59, 2023. Disponível em: <<https://doi.org/10.1145/3589658>>.
- 33 GKINKO, L.; ELBANNA, A. R. AI in the workplace: Exploring chatbot use and users' emotions. In: DENNEHY, D.; GRIVA, A.; POULOU DI, N.; DWIVEDI, Y. K.; PAPPAS, I. O.; MÄNTYMÄKI, M. (Ed.). *Responsible AI and Analytics for an Ethical and Inclusive Digitized Society - 20th IFIP WG 6.11 Conference on e-Business, e-Services and e-Society, I3E 2021, Galway, Ireland, September 1-3, 2021, Proceedings*. Springer, 2021. (Lecture Notes in Computer Science, v. 12896), p. 18–28. Disponível em: <https://doi.org/10.1007/978-3-030-85447-8_2>.
- 34 HAUGELAND, I. K. F.; FØLSTAD, A.; TAYLOR, C.; BJØRKL I, C. A. Understanding the user experience of customer service chatbots: An experimental study of chatbot interaction design. *Int. J. Hum. Comput. Stud.*, v. 161, p. 102788, 2022. Disponível em: <<https://doi.org/10.1016/j.ijhcs.2022.102788>>.
- 35 SUCAMELI, I. Improving the level of trust in human-machine conversation. *Adv. Robotics*, v. 35, n. 9, p. 553–560, 2021. Disponível em: <<https://doi.org/10.1080/01691864.2021.1884132>>.
- 36 SHEVAT, A. *Designing bots: Creating conversational experiences*. [S.l.]: "O'Reilly Media, Inc.", 2017.
- 37 NG, M.; COOPAMOOTOO, K. P. L.; TOREINI, E.; AITKEN, M.; ELLIOTT, K.; MOORSEL, A. van. Simulating the effects of social presence on trust, privacy concerns & usage intentions in automated bots for finance. In: *IEEE European Symposium on Security and Privacy Workshops, EuroS&P Workshops 2020, Genoa, Italy, September 7-11, 2020*. IEEE, 2020. p. 190–199. Disponível em: <<https://doi.org/10.1109/EuroSPW51379.2020.00034>>.
- 38 SHARMA, S.; PANDEY, S. K. Revisiting requirements documentation techniques and challenges. In: PANT, M.; SHARMA, T. K.; VERMA, O. P.; SINGLA, R.; SIKANDER, A. (Ed.). *Soft Computing: Theories and Applications*. Singapore: Springer Singapore, 2020. p. 793–802. ISBN 978-981-15-0751-9.
- 39 SOHAN, S. M.; MAURER, F.; ANSLOW, C.; ROBILLARD, M. P. A study of the effectiveness of usage examples in rest api documentation. In: *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. [S.l.: s.n.], 2017. p. 53–61.
- 40 JENNEBOER, L.; HERRANDO, C.; CONSTANTINIDES, E. The impact of chatbots on customer loyalty: A systematic literature review. *Journal of theoretical and applied electronic commerce research*, MDPI, v. 17, n. 1, p. 212–229, 2022.

- 41 TUFAIL, H.; MASOOD, M. F.; ZEB, B.; AZAM, F.; ANWAR, M. W. A systematic review of requirement traceability techniques and tools. In: *2nd International Conference on System Reliability and Safety, ICSRS 2017, Milan, Italy, December 20-22, 2017*. IEEE, 2017. p. 450–454. Disponível em: <<https://doi.org/10.1109/ICSRS.2017.8272863>>.
- 42 KITCHENHAM, B. A. Systematic review in software engineering: where we are and where we should be going. In: *Proceedings of the 2nd international workshop on Evidential assessment of software technologies*. [S.l.: s.n.], 2012. p. 1–2.
- 43 SILVA, G. R. S.; CANEDO, E. D. Requirements engineering challenges and techniques in building chatbots. In: ROCHA, A. P.; STEELS, L.; HERIK, H. J. van den (Ed.). *Proceedings of the 14th International Conference on Agents and Artificial Intelligence, ICAART 2022, Volume 1, Online Streaming, February 3-5, 2022*. SCITEPRESS, 2022. p. 180–187. Disponível em: <<https://doi.org/10.5220/0010801800003116>>.
- 44 RADZIWILL, N. M.; BENTON, M. C. Evaluating quality of chatbots and intelligent conversational agents. *CoRR*, abs/1704.04579, 2017. Disponível em: <<http://arxiv.org/abs/1704.04579>>.
- 45 TURING, A. Computing machinery and intelligence. *Mind*, v. 59, n. 236, p. 433–460, 1950.
- 46 ROGERS, C. R. *Client-Centered Therapy: Its Current Practice, Implications and Theory*. [S.l.]: Houghton Mifflin, Boston, 1951.
- 47 COLBY, K. M. *Artificial Paranoia: A Computer Simulation of Paranoid Processes*. [S.l.]: Pergamon Press, 1975.
- 48 FRYER, L.; CARPENTER, R. Bots as language learning tools. University of Hawaii National Foreign Language Resource Center, 2006.
- 49 MAULDIN, M. L. Chatterbots, tinymuds, and the turing test: Entering the loebner prize competition. In: *Proceedings of the Eleventh National Conference on Artificial Intelligence*. [S.l.: s.n.], 1994. p. 16–21.
- 50 ZDRAVKOVA, K. Conceptual framework for an intelligent chatterbot. In: *Proceedings of the 22nd International Conference on Information Technology Interfaces*. [S.l.: s.n.], 2000. p. 189–194.
- 51 WALLACE, R. S. The anatomy of a.l.i.c.e. In: _____. *Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer*. Dordrecht: Springer Netherlands, 2009. p. 181–210. ISBN 978-1-4020-6710-5. Disponível em: <https://doi.org/10.1007/978-1-4020-6710-5_13>.
- 52 TURING, A. Computing machinery and intelligence. In: _____. *Parsing the Turing Test*. [S.l.]: Springer, Dordrecht, 2003. p. 23–65.
- 53 BRANDTZÆG, P. B.; FØLSTAD, A. Why people use chatbots. In: KOMPATSIARIS, I.; CAVE, J.; SATSIU, A.; CARLE, G.; PASSANI, A.; KONTOPOULOS, E.; DIPLARIS, S.; MCMILLAN, D. (Ed.). *Internet Science - 4th International Conference, INSCI 2017, Thessaloniki, Greece, November 22-24, 2017, Proceedings*. Springer, 2017. (Lecture Notes in Computer Science, v. 10673), p. 377–392. Disponível em: <https://doi.org/10.1007/978-3-319-70284-1_30>.
- 54 HEATON, J. Ian goodfellow, yoshua bengio, and aaron courville: Deep learning - the MIT press, 2016, 800 pp, ISBN: 0262035618. *Genet. Program. Evolvable Mach.*, v. 19, n. 1-2, p. 305–307, 2018. Disponível em: <<https://doi.org/10.1007/s10710-017-9314-z>>.
- 55 CHURCH, K.; OLIVEIRA, R. de. What's up with whatsapp?: comparing mobile instant messaging behaviors with traditional SMS. In: ROHS, M.; SCHMIDT, A.; ASHBROOK, D.; RUKZIO, E. (Ed.). *15th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '13, Munich, Germany, August 27 - 30, 2013*. ACM, 2013. p. 352–361. Disponível em: <<https://doi.org/10.1145/2493190.2493225>>.

- 56 KUMAR, B. *How Conversational Commerce Is Forever Changing the Way We Shop*. Shopify, 2016. Disponível em: <<https://www.shopify.com/blog/113660229-how-conversational-commerce-is-forever-changing-the-way-we-shop>>.
- 57 RADFORD, A.; WU, J.; CHILD, R.; LUAN, D.; AMODEI, D.; SUTSKEVER, I. Language models are unsupervised multitask learners. OpenAI LP, 2019.
- 58 VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2017. p. 5998–6008.
- 59 WU, T.; HE, S.; LIU, J.; SUN, S.; LIU, K.; HAN, Q.; TANG, Y. A brief overview of chatgpt: The history, status quo and potential future development. *IEEE CAA J. Autom. Sinica*, v. 10, n. 5, p. 1122–1136, 2023. Disponível em: <<https://doi.org/10.1109/JAS.2023.123618>>.
- 60 NURUZZAMAN, M.; HUSSAIN, O. K. A survey on chatbot implementation in customer service industry through deep neural networks. In: *15th IEEE International Conference on e-Business Engineering, ICEBE 2018, Xi'an, China, October 12-14, 2018*. IEEE Computer Society, 2018. p. 54–61. Disponível em: <<https://doi.org/10.1109/ICEBE.2018.00019>>.
- 61 SHEEHAN, B.; JIN, H. S.; GOTTLIEB, U. Customer service chatbots: Anthropomorphism and adoption. *Journal of Business Research*, v. 115, p. 14–24, 2020. ISSN 0148-2963. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0148296320302484>>.
- 62 GNEWUCH, U.; MORANA, S.; ADAM, M. T. P.; MAEDCHE, A. Opposing effects of response time in human-chatbot interaction. *Bus. Inf. Syst. Eng.*, v. 64, n. 6, p. 773–791, 2022. Disponível em: <<https://doi.org/10.1007/s12599-022-00755-x>>.
- 63 XIE, Y.; QU, J.; ZHANG, Y.; ZHOU, R.; CHAN, A. H. S. Speaking, fast or slow: how conversational agents' rate of speech influences user experience. *Universal Access in the Information Society*, May 2023. ISSN 1615-5297. Disponível em: <<https://doi.org/10.1007/s10209-023-01000-2>>.
- 64 BELDA-MEDINA, J.; CALVO-FERRER, J. R. Using chatbots as ai conversational partners in language learning. *Applied Sciences*, MDPI, v. 12, n. 17, p. 8427, 2022.
- 65 GO, E.; SUNDAR, S. S. Humanizing chatbots: The effects of visual, identity and conversational cues on humanness perceptions. *Comput. Hum. Behav.*, v. 97, p. 304–316, 2019. Disponível em: <<https://doi.org/10.1016/j.chb.2019.01.020>>.
- 66 SILVA, A. de B.; GOMES, M. M.; COSTA, C. A. da; RIGHI, R. da R.; BARBOSA, J. L. V.; PESSIN, G.; DONCKER, G. D.; FEDERIZZI, G. Intelligent personal assistants: A systematic literature review. *Expert Systems with Applications*, Elsevier, v. 147, p. 113193, 2020.
- 67 FITZPATRICK, K. K.; DARCY, A.; VIERHILE, M. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): a randomized controlled trial. *JMIR mental health*, JMIR Publications Inc., Toronto, Canada, v. 4, n. 2, p. e7785, 2017.
- 68 MESKO, D. B. *The Medical Futurist*. 2020. <<https://medicalfuturist.com/magazine/>>. [Online; accessed on (07 de julho de 2023)].
- 69 GUPTA, J.; SINGH, V.; KUMAR, I. Florence- a health care chatbot. In: *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*. [S.l.: s.n.], 2021. v. 1, p. 504–508.

- 70 SURANI, A.; DAS, S. Understanding privacy and security postures of healthcare chatbots. In: *Chi*. [S.l.: s.n.], 2022. v. 22, p. 1–7.
- 71 WHITE, G. *Child advice chatbots fail to spot sexual abuse*. 2018. <<https://www.bbc.com/news/technology-46507900>>. Accessed: 07-07-2023.
- 72 GUPTA, S.; JAGANNATH, K.; AGGARWAL, N.; SRIDAR, R.; WILDE, S.; CHEN, Y. Artificially intelligent (AI) tutors in the classroom: A need assessment study of designing chatbots to support student learning. In: WEI, K. K.; HUANG, W. W.; LEE, J. K.; XU, D.; JIANG, J. J.; KIM, H. (Ed.). *23rd Pacific Asia Conference on Information Systems, PACIS 2019, X'ian, China, July 8-12, 2019*. [s.n.], 2019. p. 213. Disponível em: <<https://aisel.aisnet.org/pacis2019/213>>.
- 73 GOMES, R. S.; BARBOSA, D. N. F.; GEYER, C. F. R. Lassalinho: um agente pedagógico animado em um ambiente multiagente para educação a distância. *RENOTE: revista novas tecnologias na educação [recurso eletrônico]*. Porto Alegre, RS, 2005.
- 74 SANDU, N.; GIDE, E. Adoption of ai-chatbots to enhance student learning experience in higher education in india. In: *18th International Conference on Information Technology Based Higher Education and Training, ITHET 2019, Magdeburg, Germany, September 26-27, 2019*. IEEE, 2019. p. 1–5. Disponível em: <<https://doi.org/10.1109/ITHET46829.2019.8937382>>.
- 75 CHEN, Y.; JENSEN, S.; ALBERT, L. J.; GUPTA, S.; LEE, T. Artificial intelligence (ai) student assistants in the classroom: Designing chatbots to support student success. *Information Systems Frontiers*, Springer, v. 25, n. 1, p. 161–182, 2023.
- 76 CHEN, Y.; JENSEN, S.; ALBERT, L. J.; GUPTA, S.; LEE, T. Artificial intelligence (AI) student assistants in the classroom: Designing chatbots to support student success. *Inf. Syst. Frontiers*, v. 25, n. 1, p. 161–182, 2023. Disponível em: <<https://doi.org/10.1007/s10796-022-10291-4>>.
- 77 XU, R.; FENG, Y.; CHEN, H. Chatgpt vs. google: A comparative study of search performance and user experience. *CoRR*, abs/2307.01135, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2307.01135>>.
- 78 BORJI, A. A categorical archive of chatgpt failures. *CoRR*, abs/2302.03494, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2302.03494>>.
- 79 ZHANG, C.; ZHANG, C.; LI, C.; QIAO, Y.; ZHENG, S.; DAM, S. K.; ZHANG, M.; KIM, J. U.; KIM, S. T.; CHOI, J.; PARK, G.; BAE, S.; LEE, L.; HUI, P.; KWEON, I. S.; HONG, C. S. One small step for generative ai, one giant leap for AGI: A complete survey on chatgpt in AIGC era. *CoRR*, abs/2304.06488, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2304.06488>>.
- 80 NIMAVAT, K.; CHAMPANERIA, T. Chatbots: An overview types, architecture, tools and future possibilities. *Int. J. Sci. Res. Dev*, v. 5, n. 7, p. 1019–1024, 2017.
- 81 KUCHERBAEV, P.; BOZZON, A.; HOUBEN, G. Human-aided bots. *IEEE Internet Comput.*, v. 22, n. 6, p. 36–43, 2018. Disponível em: <<https://doi.org/10.1109/MIC.2018.252095348>>.
- 82 THORAT, S. A.; JADHAV, V. A review on implementation issues of rule-based chatbot systems. In: *Proceedings of the International Conference on Innovative Computing & Communications (ICICC) 2020*. [s.n.], 2020. Disponível em: <<https://ssrn.com/abstract=3567047>>.
- 83 HIEN, H. T.; CUONG, P.; NAM, L. N. H.; NHUNG, H. L. T. K.; DINH, T. L. Intelligent assistants in higher-education environments: The fit-ebot, a chatbot for administrative and learning support. In: *Proceedings of the Ninth International Symposium on Information and Communication Technology, SoICT 2018, Danang City, Vietnam, December 06-07, 2018*. ACM, 2018. p. 69–76. Disponível em: <<https://doi.org/10.1145/3287921.3287937>>.

- 84 ALBAYRAK, N.; OZDEMIR, A.; ZEYDAN, E. An overview of artificial intelligence based chatbots and an example chatbot application. In: *26th Signal Processing and Communications Applications Conference, SIU 2018, Izmir, Turkey, May 2-5, 2018*. IEEE, 2018. p. 1–4. Disponível em: <<https://doi.org/10.1109/SIU.2018.8404430>>.
- 85 CHENG, X.; ZHANG, X.; COHEN, J.; MOU, J. Human vs. ai: Understanding the impact of anthropomorphism on consumer response to chatbots from the perspective of trust and relationship norms. *Information Processing & Management*, v. 59, n. 3, p. 102940, 2022. ISSN 0306-4573. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0306457322000620>>.
- 86 JIN, S. V.; YOUN, S. Why do consumers with social phobia prefer anthropomorphic customer service chatbots? evolutionary explanations of the moderating roles of social phobia. *Telematics and Informatics*, v. 62, p. 101644, 2021. ISSN 0736-5853. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0736585321000836>>.
- 87 Google Cloud. *Google Cloud Dialogflow Documentation*. 2023. Website. Disponível em: <<https://cloud.google.com/dialogflow/docs>>.
- 88 RAPP, A.; CURTI, L.; BOLDI, A. The human side of human-chatbot interaction: A systematic literature review of ten years of research on text-based chatbots. *Int. J. Hum. Comput. Stud.*, v. 151, p. 102630, 2021. Disponível em: <<https://doi.org/10.1016/j.ijhcs.2021.102630>>.
- 89 SHI, W.; WANG, X.; OH, Y.; ZHANG, J.; SAHAY, S.; YU, Z. Effects of persuasive dialogues: Testing bot identities and inquiry strategies. In: BERNHAUPT, R.; MUELLER, F. F.; VERWEIJ, D.; ANDRES, J.; MCGRENERE, J.; COCKBURN, A.; AVELLINO, I.; GOGUEY, A.; BJØN, P.; ZHAO, S.; SAMSON, B. P.; KOCIELNIK, R. (Ed.). *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*. ACM, 2020. p. 1–13. Disponível em: <<https://doi.org/10.1145/3313831.3376843>>.
- 90 LUNSFORD, R.; OVIATT, S. L. Human perception of intended addressee during computer-assisted meetings. In: QUEK, F. K. H.; YANG, J.; MASSARO, D. W.; ALWAN, A. A.; HAZEN, T. J. (Ed.). *Proceedings of the 8th International Conference on Multimodal Interfaces, ICMI 2006, Banff, Alberta, Canada, November 2-4, 2006*. ACM, 2006. p. 20–27. Disponível em: <<https://doi.org/10.1145/1180995.1181002>>.
- 91 HOY, M. B. Alexa, siri, cortana, and more: An introduction to voice assistants. *Medical Reference Services Quarterly*, Routledge, v. 37, n. 1, p. 81–88, 2018. PMID: 29327988. Disponível em: <<https://doi.org/10.1080/02763869.2018.1404391>>.
- 92 BABY, C. J.; KHAN, F. A.; SWATHI, J. N. Home automation using iot and a chatbot using natural language processing. In: *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*. [S.l.: s.n.], 2017. p. 1–6.
- 93 WASIELEWSKI, A. “midjourney can’t count”: Questions of representation and meaning for text-to-image generators. *The Interdisciplinary Journal of Image Sciences*, v. 37, n. 1, p. 71–82, 2023.
- 94 BIRD, C.; UNGLESS, E. L.; KASIRZADEH, A. Typology of risks of generative text-to-image models. *CoRR*, abs/2307.05543, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2307.05543>>.
- 95 EHRENPREIS, M.; DELOOPER, J. Implementing a chatbot on a library website. *Journal of Web Librarianship*, Routledge, v. 16, n. 2, p. 120–142, 2022. Disponível em: <<https://doi.org/10.1080/19322909.2022.2060893>>.
- 96 CUI, L.; HUANG, S.; WEI, F.; TAN, C.; DUAN, C.; ZHOU, M. Superagent: A customer service chatbot for e-commerce websites. In: *Proceedings of ACL 2017, system demonstrations*. [S.l.: s.n.], 2017. p. 97–102.

- 97 STIEGLITZ, S.; BRACHTEN, F.; ROSS, B.; JUNG, A. Do social bots dream of electric sheep? A categorisation of social media bot accounts. In: *Australasian Conference on Information Systems, ACIS 2017, Hobart, Tasmania, Australia, 4-6 December 2017*. [s.n.], 2017. p. 89. Disponível em: <<https://aisel.aisnet.org/acis2017/89>>.
- 98 Tribunal Superior Eleitoral (TSE). *TSE e Telegram formalizam parceria contra desinformação*. 2022. Website. Disponível em: <<https://www.tse.jus.br/comunicacao/noticias/2022/Maio/tse-e-telegram-formalizam-parceria-contradesinformacao>>.
- 99 World Health Organization (WHO). *WHO launches a chatbot on Facebook Messenger to combat COVID-19 misinformation*. 2023. Website. Disponível em: <<https://www.who.int/news-room/feature-stories/detail/who-launches-a-chatbot-powered-facebook-messenger-to-combat-covid-19-misinformation>>.
- 100 LIN, J. E.; WU, L. Examining the psychological process of developing consumer-brand relationships through strategic use of social media brand chatbots. *Comput. Hum. Behav.*, v. 140, p. 107488, 2023. Disponível em: <<https://doi.org/10.1016/j.chb.2022.107488>>.
- 101 ALEPIS, E.; PATSAKIS, C. Monkey says, monkey does: Security and privacy on voice assistants. *IEEE Access*, v. 5, p. 17841–17851, 2017.
- 102 ALOUFI, R.; HADDADI, H.; BOYLE, D. *Emotionless: Privacy-Preserving Speech Analysis for Voice Assistants*. 2019.
- 103 TANG, C.; GUERIN, F.; LIN, C. *Recent Advances in Neural Text Generation: A Task-Agnostic Survey*. 2023.
- 104 KOUDENBURG, N.; POSTMES, T.; GORDIJN, E. H. Conversational flow and entitativity: The role of status. *British Journal of Social Psychology*, Wiley Online Library, v. 53, n. 2, p. 350–366, 2014.
- 105 SILVA, G. R. S.; RODRIGUES, G. N.; CANEDO, E. D. A modeling strategy for the verification of context-oriented chatbot conversational flows via model checking. *Journal of Universal Computer Science*, Pensoft Publishers, v. 29, n. 7, p. 805, 2023.
- 106 KOUDENBURG, N.; POSTMES, T.; GORDIJN, E. H. Beyond content of conversation: The role of conversational form in the emergence and regulation of social structure. *Personality and Social Psychology Review*, Sage Publications Sage CA: Los Angeles, CA, v. 21, n. 1, p. 50–71, 2017.
- 107 CLARK, A. E.; KASHIMA, Y. Stereotypes help people connect with others in the community: A situated functional analysis of the stereotype consistency bias in communication. *Journal of personality and social psychology*, American Psychological Association, v. 93, n. 6, p. 1028, 2007.
- 108 ANDICS, A.; GÁBOR, A.; GÁCSI, M.; FARAGÓ, T.; SZABÓ, D.; MIKLÓSI, A. Neural mechanisms for lexical processing in dogs. *Science*, v. 353, n. 6303, p. 1030–1032, 2016. Disponível em: <<https://www.science.org/doi/abs/10.1126/science.aaf3777>>.
- 109 PAIKARI, E.; HOEK, A. van der. A framework for understanding chatbots and their future. In: SHARP, H.; SOUZA, C. R. B. de; GRAZIOTIN, D.; LEVY, M.; SOCHA, D. (Ed.). *Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering, ICSE 2018, Gothenburg, Sweden, May 27 - June 03, 2018*. ACM, 2018. p. 13–16. Disponível em: <<https://doi.org/10.1145/3195836.3195859>>.
- 110 MOORE, R. J.; ARAR, R. *Conversational UX Design: A Practitioner's Guide to the Natural Conversation Framework*. ACM, 2019. v. 27. (ACM Books, v. 27). ISBN 978-1-4503-6301-3. Disponível em: <<https://doi.org/10.1145/3304087>>.

- 111 CAMERON, G.; CAMERON, D.; MEGAW, G.; BOND, R.; MULVENNA, M.; O'NEILL, S.; ARMOUR, C.; MCTEAR, M. Back to the future: lessons from knowledge engineering methodologies for chatbot design and development. In: BCS LEARNING & DEVELOPMENT LTD. *British HCI Conference 2018*. [S.l.], 2018.
- 112 RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach (4th Edition)*. Pearson, 2020. ISBN 9780134610993. Disponível em: <<http://aima.cs.berkeley.edu/>>.
- 113 MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient estimation of word representations in vector space. In: BENGIO, Y.; LECUN, Y. (Ed.). *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*. [s.n.], 2013. Disponível em: <<http://arxiv.org/abs/1301.3781>>.
- 114 MOHAMMED, M.; AREF, M. M. Chatbot system architecture. *CoRR*, abs/2201.06348, 2022. Disponível em: <<https://arxiv.org/abs/2201.06348>>.
- 115 JANARTHANAM, S. *Hands-On Chatbots and Conversational UI Development: Build Chatbots and Voice User Interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills*. [S.l.]: Packt Publishing, 2017. ISBN 1788294661.
- 116 LIN, W. Y. Prototyping a chatbot for site managers using building information modeling (BIM) and natural language understanding (NLU) techniques. *Sensors*, v. 23, n. 6, p. 2942, 2023. Disponível em: <<https://doi.org/10.3390/s23062942>>.
- 117 MIELKE, S. J.; ALYAFEAI, Z.; SALESKY, E.; RAFFEL, C.; DEY, M.; GALLÉ, M.; RAJA, A.; SI, C.; LEE, W. Y.; SAGOT, B.; TAN, S. Between words and characters: A brief history of open-vocabulary modeling and tokenization in NLP. *CoRR*, abs/2112.10508, 2021. Disponível em: <<https://arxiv.org/abs/2112.10508>>.
- 118 SHARMA, A.; AMRITA; CHAKRABORTY, S.; KUMAR, S. Named entity recognition in natural language processing: A systematic review. In: GUPTA, D.; KHANNA, A.; KANSAL, V.; FORTINO, G.; HASSANIEN, A. E. (Ed.). *Proceedings of Second Doctoral Symposium on Computational Intelligence*. Singapore: Springer Singapore, 2022. p. 817–828. ISBN 978-981-16-3346-1.
- 119 VAN-HORENBEKE, F. A.; PEER, A. Activity, plan, and goal recognition: A review. *Frontiers in Robotics and AI*, v. 8, 2021. ISSN 2296-9144. Disponível em: <<https://www.frontiersin.org/articles/10.3389/frobt.2021.643010>>.
- 120 HIGH, R. *The Era of Cognitive Systems: An Inside Look at IBM Watson and How It Works*. [S.l.], 2012.
- 121 PORTMAN, H. *Project Success QRC Standish Group Chaos Report 2020*. 2021. <<https://hennyportman.files.wordpress.com/2021/01/project-success-qrc-standish-group-chaos-report-2020.pdf>>. Accessed: 2023-07-09.
- 122 THORNE, C. Chatbots for troubleshooting: A survey. *Language and Linguistics Compass*, v. 11, n. 10, p. e12253, 2017.
- 123 DAI, Y.; YU, H.; JIANG, Y.; TANG, C.; LI, Y.; SUN, J. A survey on dialog management: Recent advances and challenges. *CoRR*, abs/2005.02233, 2020. Disponível em: <<https://arxiv.org/abs/2005.02233>>.
- 124 GATT, A.; KRAHMER, E. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *J. Artif. Intell. Res.*, v. 61, p. 65–170, 2018. Disponível em: <<https://doi.org/10.1613/jair.5477>>.

- 125 MA, Z.; DOU, Z.; ZHU, Y.; ZHONG, H.; WEN, J.-R. One chatbot per person: Creating personalized chatbots based on implicit user profiles. In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: Association for Computing Machinery, 2021. (SIGIR '21), p. 555–564. ISBN 9781450380379. Disponível em: <<https://doi.org/10.1145/3404835.3462828>>.
- 126 MAZUMDER, S.; MA, N.; LIU, B. *Towards a Continuous Knowledge Learning Engine for Chatbots*. 2018.
- 127 LOUCOPOULOS, P.; KARAKOSTAS, V. *System requirements engineering*. [S.l.]: McGraw-Hill, Inc., 1995.
- 128 VIERLBOECK, M.; NILCHIANI, R. R. Requirement engineering in the age of system and product complexity – a literature review. In: *2021 IEEE International Symposium on Systems Engineering (ISSE)*. [S.l.: s.n.], 2021. p. 1–8.
- 129 MATHARU, G. S.; MISHRA, A.; SINGH, H.; UPADHYAY, P. Empirical study of agile software development methodologies: A comparative analysis. Association for Computing Machinery, New York, NY, USA, v. 40, n. 1, p. 1–6, feb 2015. ISSN 0163-5948. Disponível em: <<https://doi.org/10.1145/2693208.2693233>>.
- 130 ANWER, S.; WEN, L.; WANG, Z. A systematic approach for identifying requirement change management challenges: Preliminary results. In: *Proceedings of the 23rd International Conference on Evaluation and Assessment in Software Engineering*. New York, NY, USA: Association for Computing Machinery, 2019. (EASE '19), p. 230–235. ISBN 9781450371452. Disponível em: <<https://doi.org/10.1145/3319008.3319031>>.
- 131 GROUP, S. et al. Chaos report 2020. *The Standish Group International, Inc*, p. 1–9, 2020.
- 132 MAXIM, B. R.; PRESSMAN, R. S. *Software engineering: A practitioner's approach*. *Britania Raya: McGraw-Hill Education*, 2014.
- 133 ECKHARDT, J.; VOGELSANG, A.; FERNÁNDEZ, D. M. On the distinction of functional and quality requirements in practice. In: *Lecture Notes in Computer Science*. [S.l.: s.n.], 2016. p. 31–47.
- 134 NURBOJATMIKO; BUDIARDJO, E. K.; WIBOWO, W. C. Slr on identification & classification of non-functional requirements attributes, and its representation in functional requirements. In: *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence, CSAI 2018, the 10th International Conference on Information and Multimedia Technology, ICIMT 2018, Shenzhen, China, December 08-10, 2018*. ACM, 2018. p. 151–157. Disponível em: <<https://doi.org/10.1145/3297156.3297200>>.
- 135 OLSINA, L.; BECKER, P. Linking business and information need goals with functional and non-functional requirements. In: GENERO, M.; KALINOWSKI, M.; MOLINA, J. G.; PINO, F.; CONTE, T.; MARÍN, B.; BRITO, I.; GIACHETTI, G. (Ed.). *Proceedings of the XXI Iberoamerican Conference on Software Engineering, Bogota, Colombia, April 23-27, 2018*. [S.l.]: Curran Associates, 2018. p. 381–394.
- 136 SHEHADEH, K.; ARMAN, N.; KHAMAYSEH, F. Semi-automated classification of arabic user requirements into functional and non-functional requirements using NLP tools. In: *International Conference on Information Technology, ICIT 2021, Amman, Jordan, July 14-15, 2021*. IEEE, 2021. p. 527–532. Disponível em: <<https://doi.org/10.1109/ICIT52682.2021.9491698>>.
- 137 SADIQ, M.; DEVI, V. S. Fuzzy-soft set approach for ranking the functional requirements of software. *Expert Syst. Appl.*, v. 193, p. 116452, 2022. Disponível em: <<https://doi.org/10.1016/j.eswa.2021.116452>>.

- 138 SCHIMANSKI, C. P.; PRADHAN, N. L.; CHALTSEV, D.; Pasetti Monizza, G.; MATT, D. T. Integrating bim with lean construction approach: Functional requirements and production management software. *Automation in Construction*, v. 132, p. 103969, 2021. ISSN 0926-5805. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0926580521004209>>.
- 139 FERNANDES, J. M.; MACHADO, R. J.; FERNANDES, J. M.; MACHADO, R. J. Requirements elicitation. *Requirements in Engineering Projects*, Springer, p. 85–117, 2016.
- 140 JARZEBOWICZ, A.; WEICHBROTH, P. A qualitative study on non-functional requirements in agile software development. *IEEE Access*, v. 9, p. 40458–40475, 2021. Disponível em: <<https://doi.org/10.1109/ACCESS.2021.3064424>>.
- 141 KELLNER, D. W. Management of availability requirement for complex systems. In: *2017 Annual Reliability and Maintainability Symposium (RAMS)*. [S.l.: s.n.], 2017. p. 1–6.
- 142 IGNATIUS, G.; MOORE, B.; MALKAWI, M.; VOTTA, L. Availability work products—a strategic approach. In: *IEEE Signal Processing Society 5th WSES International Conference, Crete*. [S.l.: s.n.], 2001.
- 143 MALKAWI, M.; AL-ZOUBI, K.; SHATNAWI, A. Quasi real-time intermodulation interference method: Analysis and performance. *Int. J. Commun. Networks Inf. Secur.*, v. 13, n. 1, 2021. Disponível em: <<https://www.ijcnis.org/index.php/ijcnis/article/view/4924>>.
- 144 FLORIO, V. D. Software assumptions failure tolerance: Role, strategies, and visions. *CoRR*, abs/1605.02040, 2016. Disponível em: <<http://arxiv.org/abs/1605.02040>>.
- 145 WANG, C.; WANG, Q.; REN, K.; CAO, N.; LOU, W. Toward secure and dependable storage services in cloud computing. *IEEE Trans. Serv. Comput.*, v. 5, n. 2, p. 220–232, 2012. Disponível em: <<https://doi.org/10.1109/TSC.2011.24>>.
- 146 ZISSIS, D.; LEKKAS, D. Addressing cloud computing security issues. *Future Gener. Comput. Syst.*, v. 28, n. 3, p. 583–592, 2012. Disponível em: <<https://doi.org/10.1016/j.future.2010.12.006>>.
- 147 LUO, L.; MENG, S.; QIU, X.; DAI, Y. Improving failure tolerance in large-scale cloud computing systems. *IEEE Trans. Reliab.*, v. 68, n. 2, p. 620–632, 2019. Disponível em: <<https://doi.org/10.1109/TR.2019.2901194>>.
- 148 ABERKANE, A. Automated gdpr-compliance in requirements engineering. In: KROGSTIE, J.; OUYANG, C.; RALYTÉ, J. (Ed.). *Proceedings of the Doctoral Consortium Papers Presented at the 33rd International Conference on Advanced Information Systems Engineering (CAiSE 2021), Melbourne, Australia, June 28 - July 2, 2021*. CEUR-WS.org, 2021. (CEUR Workshop Proceedings, v. 2906), p. 21–29. Disponível em: <<https://ceur-ws.org/Vol-2906/paper3.pdf>>.
- 149 ROCHA, L. D.; SILVA, G. R. S.; CANEDO, E. D. Privacy compliance in software development: A guide to implementing the LGPD principles. In: HONG, J.; LANPERNE, M.; PARK, J. W.; CERNÝ, T.; SHAHRIAR, H. (Ed.). *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing, SAC 2023, Tallinn, Estonia, March 27-31, 2023*. ACM, 2023. p. 1352–1361. Disponível em: <<https://doi.org/10.1145/3555776.3577615>>.
- 150 BREAUX, T. D.; ANTÓN, A. I.; SPAFFORD, E. H. A distributed requirements management framework for legal compliance and accountability. *computers & security*, Elsevier, v. 28, n. 1-2, p. 8–17, 2009.
- 151 SYREYSHCHIKOVA, N. V.; PIMENOV, D. Y.; MIKOLAJCZYK, T.; MOLDOVAN, L. Information safety process development according to is 27001 for an industrial enterprise. *Procedia Manufacturing*, v. 32, p. 278–285, 2019. ISSN 2351-9789. 12th International Conference Interdisciplinarity in

- Engineering, INTER-ENG 2018, 4–5 October 2018, Tirgu Mures, Romania. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2351978919302501>>.
- 152 NIELSEN, J. Ten usability heuristics. <http://www.nngroup.com/articles/ten-usability-heuristics/>, 2005.
- 153 FORNALCZYK, K.; BORTKO, K.; JANKOWSKI, J. Improving user attention to chatbots through a controlled intensity of changes within the interface. In: WATRÓBSKI, J.; SALABUN, W.; TORO, C.; ZANNI-MERK, C.; HOWLETT, R. J.; JAIN, L. C. (Ed.). *Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES-2021, Virtual Event / Szczecin, Poland, 8-10 September 2021*. Elsevier, 2021. (Procedia Computer Science, v. 192), p. 5112–5121. Disponível em: <<https://doi.org/10.1016/j.procs.2021.09.289>>.
- 154 INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models*. 2011. Disponível em: <<https://www.iso.org/standard/35733.html>>.
- 155 ALSOLAI, H.; ROPER, M. A systematic literature review of machine learning techniques for software maintainability prediction. *Inf. Softw. Technol.*, v. 119, 2020. Disponível em: <<https://doi.org/10.1016/j.infsof.2019.106214>>.
- 156 BEVAN, N. Usability. *Encyclopedia of database systems*, Springer, p. 3247–3251, 2009.
- 157 SCHNEIDER, M.; MEITINGER, S.; STOCK, D.; BAUERNHANSL, T. Software deployment in manufacturing environments: A requirements analysis. In: *27th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2022, Stuttgart, Germany, September 6-9, 2022*. IEEE, 2022. p. 1–8. Disponível em: <<https://doi.org/10.1109/ETFA52439.2022.9921690>>.
- 158 ABDEEN, W.; CHEN, X.; UNTERKALMSTEINER, M. An approach for performance requirements verification and test environments generation. *Requir. Eng.*, v. 28, n. 1, p. 117–144, 2023. Disponível em: <<https://doi.org/10.1007/s00766-022-00379-3>>.
- 159 AL-SARAYREH, K. T.; MERIDJI, K.; ABRAN, A.; TRUDEL, S. System performance requirements: A standards-based model for early identification, allocation to software functions and size measurement. *e Informatica Softw. Eng. J.*, v. 14, n. 1, 2020. Disponível em: <<https://doi.org/10.37190/e-inf200105>>.
- 160 MERTZ, J.; NUNES, I.; TOFFOLA, L. D.; SELAKOVIC, M.; PRADEL, M. Satisfying increasing performance requirements with caching at the application level. *IEEE Softw.*, v. 38, n. 3, p. 87–95, 2021. Disponível em: <<https://doi.org/10.1109/MS.2020.3033508>>.
- 161 ROZANSKI, N.; WOODS, E. *Software systems architecture: working with stakeholders using viewpoints and perspectives*. [S.l.]: Addison-Wesley, 2012.
- 162 GHAFFARI, F.; BERTIN, E.; CRESPI, N. User profile and mobile number portability for beyond 5g: Blockchain-based solution. In: LOPEZ, D.; MONTPETIT, M.; CERRONI, W.; MAURO, M. D.; BORYLO, P. (Ed.). *26th Conference on Innovation in Clouds, Internet and Networks, ICIN 2023, Paris, France, March 6-9, 2023*. IEEE, 2023. p. 187–194. Disponível em: <<https://doi.org/10.1109/ICIN56760.2023.10073486>>.
- 163 HERBST, N. R.; KOUNEV, S.; REUSSNER, R. Elasticity in cloud computing: What it is, and what it is not. In: *10th international conference on autonomic computing (ICAC 13)*. [S.l.: s.n.], 2013. p. 23–27.

- 164 BRATAAS, G.; FÆGRI, T. E. Agile scalability requirements. In: BINDER, W.; CORTELLESA, V.; KOZIOLEK, A.; SMIRNI, E.; POESS, M. (Ed.). *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering, ICPE 2017, L'Aquila, Italy, April 22-26, 2017*. ACM, 2017. p. 413–416. Disponível em: <<https://doi.org/10.1145/3030207.3030240>>.
- 165 ALSAQAF, W.; DANEVA, M.; WIERINGA, R. J. Quality requirements challenges in the context of large-scale distributed agile: An empirical study. *Inf. Softw. Technol.*, v. 110, p. 39–55, 2019. Disponível em: <<https://doi.org/10.1016/j.infsof.2019.01.009>>.
- 166 BRATAAS, G.; MARTINI, A.; HANSSSEN, G. K.; RÆDER, G. Agile elicitation of scalability requirements for open systems: A case study. *J. Syst. Softw.*, v. 182, p. 111064, 2021. Disponível em: <<https://doi.org/10.1016/j.jss.2021.111064>>.
- 167 BRATAAS, G.; HANSSSEN, G. K.; HERBST, N.; HOORN, A. van. Agile scalability engineering: The scrumscale method. *IEEE Softw.*, v. 37, n. 4, p. 77–84, 2020. Disponível em: <<https://doi.org/10.1109/MS.2019.2923184>>.
- 168 PATHIRAGE, D.; SHIN, M. E.; JANG, D. Modeling of security fault-tolerant requirements for secure systems. *Int. J. Softw. Eng. Knowl. Eng.*, v. 33, n. 1, p. 23–53, 2023. Disponível em: <<https://doi.org/10.1142/S0218194022500644>>.
- 169 HAKEEM, S. A. A.; HUSSEIN, H. H.; KIM, H. Security requirements and challenges of 6g technologies and applications. *Sensors*, v. 22, n. 5, p. 1969, 2022. Disponível em: <<https://doi.org/10.3390/s22051969>>.
- 170 KUMAR, R.; BHATIA, M. P. S. A systematic review of the security in cloud computing: Data integrity, confidentiality and availability. In: *2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON)*. [S.l.: s.n.], 2020. p. 334–337.
- 171 WANG, D.; BAI, B.; LEI, K.; ZHAO, W.; YANG, Y.; HAN, Z. Enhancing information security via physical layer approaches in heterogeneous iot with multiple access mobile edge computing in smart city. *IEEE Access*, v. 7, p. 54508–54521, 2019. Disponível em: <<https://doi.org/10.1109/ACCESS.2019.2913438>>.
- 172 SINGH, J.; BELLO, Y.; REFAEY, A.; MOHAMED, A. Five-layers sdp-based hierarchical security paradigm for multi-access edge computing. *CoRR*, abs/2007.01246, 2020. Disponível em: <<https://arxiv.org/abs/2007.01246>>.
- 173 SHAFAI, W. E.; HEMDAN, E. E. Robust and efficient multi-level security framework for color medical images in telehealthcare services. *J. Ambient Intell. Humaniz. Comput.*, v. 14, n. 4, p. 3675–3690, 2023. Disponível em: <<https://doi.org/10.1007/s12652-021-03494-1>>.
- 174 YANG, B.; HU, H. Dynamic implementation of security requirements in business processes. *IEEE Trans. Dependable Secur. Comput.*, v. 19, n. 2, p. 1352–1363, 2022. Disponível em: <<https://doi.org/10.1109/TDSC.2020.3012729>>.
- 175 INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO 9241-210:2010 - Ergonomics of human-system interaction*. 2023. Available: <<https://www.iso.org/standard/63500.html>>. Accessed: 2023-07-09.
- 176 PERGENTINO, A. C. D. S.; CANEDO, E. D.; LIMA, F.; MENDONÇA, F. L. L. de. Usability heuristics evaluation in search engine. In: MARCUS, A.; ROSENZWEIG, E. (Ed.). *Design, User Experience, and Usability. Interaction Design - 9th International Conference, DUXU 2020, Held as Part of the 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19-24, 2020, Proceedings, Part I*. Springer, 2020. (Lecture Notes in Computer Science, v. 12200), p. 351–369. Disponível em: <https://doi.org/10.1007/978-3-030-49713-2_25>.

- 177 COSTA, R. P. da; CANEDO, E. D. A set of usability heuristics for mobile applications. In: KUROSU, M. (Ed.). *Human-Computer Interaction. Perspectives on Design - Thematic Area, HCI 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26-31, 2019, Proceedings, Part I*. Springer, 2019. (Lecture Notes in Computer Science, v. 11566), p. 180–193. Disponível em: <https://doi.org/10.1007/978-3-030-22646-6_13>.
- 178 COSTA, R. P. da; CANEDO, E. D. Evaluation of a proposed set of usability heuristics. In: DAVIS, G.; BROWN, S.; SUBRAMANI, M. R.; ALLEN, G. N.; JOSHI, K. D.; SCHEIBE, K. P. (Ed.). *28th Americas Conference on Information Systems, AMCIS 2022, Minneapolis, MN, USA, August 10-14, 2022*. Association for Information Systems, 2022. Disponível em: <https://aisel.aisnet.org/amcis2022/sig_hci/sig_hci/12>.
- 179 SAUER, J.; SONDEREGGER, A.; SCHMUTZ, S. Usability, user experience and accessibility: towards an integrative model. *Ergonomics*, Taylor e Francis, v. 63, n. 10, p. 1207–1220, 2020. PMID: 32450782. Disponível em: <<https://doi.org/10.1080/00140139.2020.1774080>>.
- 180 CARMO, G. M. do; PAIVA, D. M. B.; CAGNIN, M. I. Incorporating accessibility in web interface development for deaf users. In: SANTOS, R. P. dos; OLIVEIRA, S. R. B.; SANTOS, I. de S.; MARCZAK, S.; VIANA, D.; CANEDO, E. D.; ALBUQUERQUE, A. B.; PALUDO, M.; ROCHA, A. R.; MACHADO, I.; KULESZA, U.; GARCIA, V.; SANTOS, G.; VALENTIM, N. M. C.; BARCELLOS, M.; ROCHA, A. R. C. da; BARCELLOS, M. P.; REINEHR, S. S.; CONTE, T.; FONTÃO, A. L.; LIMA, C. (Ed.). *SBQS '21: XX Brazilian Symposium on Software Quality, Virtual Event, Brazil, November 8 - 11, 2021*. ACM, 2021. p. 7. Disponível em: <<https://doi.org/10.1145/3493244.3493246>>.
- 181 ISHAQ, M. Improved web accessibility evaluation of open learning contents for individuals with learning disabilities. *CoRR*, abs/2306.10039, 2023. Disponível em: <<https://doi.org/10.48550/arXiv.2306.10039>>.
- 182 MORRIS, M. R. Accessibility as an opportunity and challenge for intelligent user interfaces. In: HAMMOND, T.; VERBERT, K.; PARRA, D.; KNIJNENBURG, B. P.; O'DONOVAN, J.; TEALE, P. (Ed.). *IUI '21: 26th International Conference on Intelligent User Interfaces, College Station, TX, USA, April 13-17, 2021*. ACM, 2021. p. 4. Disponível em: <<https://doi.org/10.1145/3397481.3457419>>.
- 183 SPIJKMAN, T. Conversational requirements engineering: Pinpointing requirements-relevant information in conversations. In: FERRARI, A.; PENZENSTADLER, B.; HADAR, I.; OYEDEJI, S.; ABUALHAIJA, S.; VOGELSANG, A.; DESHPANDE, G.; RACHMANN, A.; GULDEN, J.; WOHLGEMUTH, A.; HESS, A.; FRICKER, S.; GUIZZARDI, R. S. S.; HORKOFF, J.; PERINI, A.; SUSI, A.; KARRAS, O.; DALPIAZ, F.; MOREIRA, A.; AMYOT, D.; SPOLETINI, P. (Ed.). *Joint Proceedings of REFSQ-2023 Workshops, Doctoral Symposium, Posters & Tools Track and Journal Early Feedback co-located with the 28th International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2023), Barcelona, Catalunya, Spain, April 17-20, 2023*. CEUR-WS.org, 2023. (CEUR Workshop Proceedings, v. 3378). Disponível em: <<https://ceur-ws.org/Vol-3378/DS-paper5.pdf>>.
- 184 LIU, L.; DUFFY, V. G. Exploring the future development of artificial intelligence (AI) applications in chatbots: A bibliometric analysis. *Int. J. Soc. Robotics*, v. 15, n. 5, p. 703–716, 2023. Disponível em: <<https://doi.org/10.1007/s12369-022-00956-0>>.
- 185 SERBAN, I.; SORDONI, A.; LOWE, R.; CHARLIN, L.; PINEAU, J.; COURVILLE, A.; BENGIO, Y. A hierarchical latent variable encoder-decoder model for generating dialogues. In: *Proceedings of the AAAI conference on artificial intelligence*. [S.l.: s.n.], 2017. v. 31, n. 1.
- 186 CUNNINGHAM-NELSON, S.; BOLES, W.; TROUTON, L.; MARGERISON, E. A review of chatbots in education: Practical steps forward. In: *30th Annual Conference for the Australasian Association for Engineering Education (AAEE 2019): Educators Becoming Agents of Change*:

Innovate, Integrate, Motivate. Australia: Engineers Australia, 2019. p. 299–306. Disponível em: <<https://eprints.qut.edu.au/134323/>>.

187 GHANDEHARIOUN, A.; MCDUFF, D.; CZERWINSKI, M.; ROWAN, K. Towards understanding emotional intelligence for behavior change chatbots. In: *8th International Conference on Affective Computing and Intelligent Interaction, ACII 2019, Cambridge, United Kingdom, September 3-6, 2019*. IEEE, 2019. p. 8–14. Disponível em: <<https://doi.org/10.1109/ACII.2019.8925433>>.

188 DING, Q.; CARMONA, V. I. S.; LIU, M.; PENG, F.; ZHANG, Y.; HU, C. Towards A six-level framework of emotional intelligence for customer service chatbots. In: CHHAYA, N.; JAIDKA, K.; HEALEY, J.; UNGAR, L. H.; SINHA, A. (Ed.). *Proceedings of the 4th Workshop on Affective Content Analysis (AffCon 2021) co-located with Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021), February 9, 2021*. CEUR-WS.org, 2021. (CEUR Workshop Proceedings, v. 2897), p. 35–42. Disponível em: <https://ceur-ws.org/Vol-2897/AffconAAAI-21_paper5.pdf>.

189 SPRING, T.; CASAS, J.; DAHER, K.; MUGELLINI, E.; KHALED, O. A. Empathic response generation in chatbots. In: CIELIEBAK, M.; TUGGENER, D.; BENITES, F. (Ed.). *Proceedings of the 4th Swiss Text Analytics Conference, SwissText 2019, Winterthur, Switzerland, June 18-19, 2019*. CEUR-WS.org, 2019. (CEUR Workshop Proceedings, v. 2458). Disponível em: <<https://ceur-ws.org/Vol-2458/paper1.pdf>>.

190 CHAVES, A. P.; GEROSA, M. A. How should my chatbot interact? A survey on social characteristics in human-chatbot interaction design. *Int. J. Hum. Comput. Interact.*, v. 37, n. 8, p. 729–758, 2021. Disponível em: <<https://doi.org/10.1080/10447318.2020.1841438>>.

191 BENKE, I.; GNEWUCH, U.; MAEDCHE, A. Understanding the impact of control levels over emotion-aware chatbots. *Comput. Hum. Behav.*, v. 129, p. 107122, 2022. Disponível em: <<https://doi.org/10.1016/j.chb.2021.107122>>.

192 MEDEIROS, J.; VASCONCELOS, A. M. L. de; SILVA, C.; GOULÃO, M. Requirements specification for developers in agile projects: Evaluation by two industrial case studies. *Inf. Softw. Technol.*, v. 117, 2020. Disponível em: <<https://doi.org/10.1016/j.infsof.2019.106194>>.

193 BEHUTIYE, W.; SEPPÄNEN, P.; RODRÍGUEZ, P.; OIVO, M. Documentation of quality requirements in agile software development. In: LI, J.; JACCHERI, L.; DINGSØYR, T.; CHITCHYAN, R. (Ed.). *EASE '20: Evaluation and Assessment in Software Engineering, Trondheim, Norway, April 15-17, 2020*. ACM, 2020. p. 250–259. Disponível em: <<https://doi.org/10.1145/3383219.3383245>>.

194 HABIB, B.; ROMLI, R.; ZULKIFLI, M. Identifying components existing in agile software development for achieving “light but sufficient” documentation. *Journal of Engineering and Applied Science*, v. 70, n. 1, p. 75, Jul 2023. ISSN 2536-9512. Disponível em: <<https://doi.org/10.1186/s44147-023-00245-1>>.

195 WAZLAWICK, R. *Engenharia de software: conceitos e práticas*. Elsevier, 2019. 20–32 p. Disponível em: <<https://books.google.com.br/booksid=d1qnDwAAQBAJ&printsec=frontcover>>.

196 LIU, S.; GUO, Z.; LI, Y.; LU, H.; CHEN, L.; XU, L.; ZHOU, Y.; XU, B. Prioritizing code documentation effort: Can we do it simpler but better? *Inf. Softw. Technol.*, v. 140, p. 106686, 2021. Disponível em: <<https://doi.org/10.1016/j.infsof.2021.106686>>.

197 KÖNIGSTORFER, F.; THALMANN, S. Software documentation is not enough! requirements for the documentation of ai. Emerald Publishing Limited, v. 23, n. 5, p. 475–488, 2021. ISSN 2398-5038. Disponível em: <<https://doi.org/10.1108/DPRG-03-2021-0047>>.

- 198 TIZARD, J.; WANG, H.; YOHANNES, L.; BLINCOE, K. Can a conversation paint a picture? mining requirements in software forums. In: DAMIAN, D. E.; PERINI, A.; LEE, S. (Ed.). *27th IEEE International Requirements Engineering Conference, RE 2019, Jeju Island, Korea (South), September 23-27, 2019*. IEEE, 2019. p. 17–27. Disponível em: <<https://doi.org/10.1109/RE.2019.00014>>.
- 199 CASTLE-GREEN, T.; REEVES, S.; FISCHER, J. E.; KOLEVA, B. Decision trees as sociotechnical objects in chatbot design. In: TORRES, M. I.; SCHLÖGL, S.; CLARK, L.; PORCHERON, M. (Ed.). *Proceedings of the 2nd Conference on Conversational User Interfaces, CUI 2020, Bilbao, Spain, July 22-24, 2020*. ACM, 2020. p. 27:1–27:3. Disponível em: <<https://doi.org/10.1145/3405755.3406133>>.
- 200 BÖRSTING, I.; HESENIUS, M. Towards a systematic approach for chatbot development in digital work environments. *Digital Supply Chains and the Human Factor*, Springer, p. 79–94, 2021.
- 201 KITCHENHAM, B. Procedures for performing systematic reviews. *Keele, UK, Keele University, Citeseer*, v. 33, n. 2004, p. 1–26, 2004.
- 202 SCHÖN, E.; THOMASCHEWSKI, J.; ESCALONA, M. J. Agile requirements engineering: A systematic literature review. *Comput. Stand. Interfaces*, v. 49, p. 79–91, 2017. Disponível em: <<https://doi.org/10.1016/j.csi.2016.08.011>>.
- 203 PETTICREW, M.; ROBERTS, H. *Systematic Reviews in the Social Sciences*. Oxford, UK: Blackwell Publishing Ltd, 2006.
- 204 BRERETON, P.; KITCHENHAM, B. A.; BUDGEN, D.; TURNER, M.; KHALIL, M. Lessons from applying the systematic literature review process within the software engineering domain. *J. Syst. Softw.*, v. 80, n. 4, p. 571–583, 2007. Disponível em: <<https://doi.org/10.1016/j.jss.2006.07.009>>.
- 205 KITCHENHAM, B. A.; PRETORIUS, R.; BUDGEN, D.; BRERETON, P.; TURNER, M.; NIAZI, M.; LINKMAN, S. G. Systematic literature reviews in software engineering - A tertiary study. *Inf. Softw. Technol.*, v. 52, n. 8, p. 792–805, 2010. Disponível em: <<https://doi.org/10.1016/j.infsof.2010.03.006>>.
- 206 COSTAL, D.; FARRÉ, C.; FRANCH, X.; QUER, C. Inclusion and exclusion criteria in software engineering tertiary studies: A systematic mapping and emerging framework. In: LANUBILE, F.; KALINOWSKI, M.; BALDASSARRE, M. T. (Ed.). *ESEM '21: ACM / IEEE International Symposium on Empirical Software Engineering and Measurement, Bari, Italy, October 11-15, 2021*. ACM, 2021. p. 30:1–30:6. Disponível em: <<https://doi.org/10.1145/3475716.3484190>>.
- 207 KITCHENHAM, B. A.; PFLEEGER, S. L. Personal opinion surveys. In: _____. *Guide to Advanced Empirical Software Engineering*. London: Springer London, 2008. p. 63–92. ISBN 978-1-84800-044-5. Disponível em: <https://doi.org/10.1007/978-1-84800-044-5_3>.
- 208 KOÇ, H.; ERDOĞAN, A. M.; BARJAKLY, Y.; PEKER, S. Uml diagrams in software engineering research: A systematic literature review. *Proceedings*, v. 74, n. 1, 2021. ISSN 2504-3900. Disponível em: <<https://www.mdpi.com/2504-3900/74/1/13>>.
- 209 CHERGUI, M. E.; BENSLIMANE, S. M. A valid BPMN extension for supporting security requirements based on cyber security ontology. In: ABDELWAHED, E. H.; BELLATRECHE, L.; GOLFARELLI, M.; MÉRY, D.; ORDONEZ, C. (Ed.). *Model and Data Engineering - 8th International Conference, MEDI 2018, Marrakesh, Morocco, October 24-26, 2018, Proceedings*. Springer, 2018. (Lecture Notes in Computer Science, v. 11163), p. 219–232. Disponível em: <https://doi.org/10.1007/978-3-030-00856-7_14>.
- 210 MORNIE, M. N.; JALI, N.; JUNAINI, S. N.; MIT, E.; SHIANG, C. W.; SAEED, S. Visualisation of user stories in UML models: A systematic literature review. *Acta Inform. Pragensia*, Prague University of Economics and Business, v. 12, n. 2, p. 419–438, out. 2023.

- 211 KOPP, O.; BINZ, T.; BREITENBÜCHER, U.; LEYMANN, F. BPMN4TOSCA: A domain-specific language to model management plans for composite applications. In: MENDLING, J.; WEIDLICH, M. (Ed.). *Business Process Model and Notation - 4th International Workshop, BPMN 2012, Vienna, Austria, September 12-13, 2012. Proceedings*. Springer, 2012. (Lecture Notes in Business Information Processing, v. 125), p. 38–52. Disponível em: <https://doi.org/10.1007/978-3-642-33155-8_4>.
- 212 COMPAGNUCCI, I.; CORRADINI, F.; FORNARI, F.; POLINI, A.; RE, B.; TIEZZI, F. A systematic literature review on iot-aware business process modeling views, requirements and notations. *Softw. Syst. Model.*, v. 22, n. 3, p. 969–1004, 2023. Disponível em: <<https://doi.org/10.1007/s10270-022-01049-2>>.
- 213 SANTOS, I. de S.; ANDRADE, R. M. C.; NETO, P. de Alcântara dos S. Templates for textual use cases of software product lines: results from a systematic mapping study and a controlled experiment. *J. Softw. Eng. Res. Dev.*, v. 3, p. 5, 2015. Disponível em: <<https://doi.org/10.1186/s40411-015-0020-3>>.
- 214 MEDEIROS, J.; GOULÃO, M.; VASCONCELOS, A. M. L. de; SILVA, C. T. L. L. Towards a model about quality of software requirements specification in agile projects. In: PAULK, M. C.; MACHADO, R. J.; BRITO, M. A.; GOULÃO, M.; AMARAL, V. (Ed.). *10th International Conference on the Quality of Information and Communications Technology, QUATIC 2016, Lisbon, Portugal, September 6-9, 2016*. IEEE Computer Society, 2016. p. 236–241. Disponível em: <<https://doi.ieeecomputersociety.org/10.1109/QUATIC.2016.058>>.
- 215 NAWROCKI, J. R.; OCHODEK, M.; JURKIEWICZ, J.; KOPCZYNSKA, S.; ALCHIMOWICZ, B. Agile requirements engineering: A research perspective. In: GEFFERT, V.; PRENEEL, B.; ROVAN, B.; STULLER, J.; TJOA, A. M. (Ed.). *SOFSEM 2014: Theory and Practice of Computer Science - 40th International Conference on Current Trends in Theory and Practice of Computer Science, Nový Smokovec, Slovakia, January 26-29, 2014, Proceedings*. Springer, 2014. (Lecture Notes in Computer Science, v. 8327), p. 40–51. Disponível em: <https://doi.org/10.1007/978-3-319-04298-5_5>.
- 216 FAROOQ, M. S.; OMER, U.; RAMZAN, A.; RASHEED, M. A.; ATAL, Z. Behavior driven development: A systematic literature review. *IEEE Access*, v. 11, p. 88008–88024, 2023. Disponível em: <<https://doi.org/10.1109/ACCESS.2023.3302356>>.
- 217 OGATA, S.; MATSUURA, S. A review method for UML requirements analysis model employing system-side prototyping. *Springerplus*, Springer Nature, v. 2, n. 1, p. 134, dez. 2013.
- 218 VERBRUGGEN, C.; SNOECK, M. Practitioners’ experiences with model-driven engineering: a meta-review. *Softw. Syst. Model.*, v. 22, n. 1, p. 111–129, 2023. Disponível em: <<https://doi.org/10.1007/s10270-022-01020-1>>.
- 219 LINDEN, D. van der; HADAR, I.; ZAMANSKY, A. What practitioners really want: requirements for visual notations in conceptual modeling. *Softw. Syst. Model.*, v. 18, n. 3, p. 1813–1831, 2019. Disponível em: <<https://doi.org/10.1007/s10270-018-0667-4>>.
- 220 WAGENAAR, G.; OVERBEEK, S.; LUCASSEN, G.; BRINKKEMPER, S.; SCHNEIDER, K. Working software over comprehensive documentation - rationales of agile teams for artefacts usage. *J. Softw. Eng. Res. Dev.*, v. 6, p. 7, 2018. Disponível em: <<https://doi.org/10.1186/s40411-018-0051-7>>.
- 221 KUSTIAWAN, Y. A.; LIM, T. Y. User stories in requirements elicitation: A systematic literature review. In: *8th IEEE International Conference On Software Engineering and Computer Systems, ICSECS 2023, Penang, Malaysia, August 25-27, 2023*. IEEE, 2023. p. 211–216. Disponível em: <<https://doi.org/10.1109/ICSECS58457.2023.10256364>>.
- 222 LUCIA, A. D.; QUSEF, A. Requirements engineering in agile software development. *J. Emerg. Technol. Web Intell.*, Engineering and Technology Publishing, v. 2, n. 3, ago. 2010.

- 223 OBENDORF, H.; FINCK, M. Scenario-based usability engineering techniques in agile development processes. In: CZERWINSKI, M.; LUND, A. M.; TAN, D. S. (Ed.). *Extended Abstracts Proceedings of the 2008 Conference on Human Factors in Computing Systems, CHI 2008, Florence, Italy, April 5-10, 2008*. ACM, 2008. p. 2159–2166. Disponível em: <<https://doi.org/10.1145/1358628.1358649>>.
- 224 RIVERO, J. M.; GRIGERA, J.; ROSSI, G.; Robles Luna, E.; MONTERO, F.; GAEDKE, M. Mockup-driven development: Providing agile support for model-driven web engineering. *Information and Software Technology*, v. 56, n. 6, p. 670–687, 2014. ISSN 0950-5849. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0950584914000226>>.
- 225 AMNA, A. R.; POELS, G. Systematic literature mapping of user story research. *IEEE Access*, v. 10, p. 51723–51746, 2022. Disponível em: <<https://doi.org/10.1109/ACCESS.2022.3173745>>.
- 226 BOZYIĞIT, F.; AKTAŞ Özlem; KİLİNC, D. Linking software requirements and conceptual models: A systematic literature review. *Engineering Science and Technology, an International Journal*, v. 24, n. 1, p. 71–82, 2021. ISSN 2215-0986. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2215098620342580>>.
- 227 KUCHAR, S.; POLANCIC, G. Conceptualization, measurement, and application of semantic transparency in visual notations. *Softw. Syst. Model.*, v. 20, n. 6, p. 2155–2197, 2021. Disponível em: <<https://doi.org/10.1007/s10270-021-00888-9>>.
- 228 DURAN, M. B.; MUSSBACHER, G. Reusability in goal modeling: A systematic literature review. *Inf. Softw. Technol.*, v. 110, p. 156–173, 2019. Disponível em: <<https://doi.org/10.1016/j.infsof.2019.03.004>>.
- 229 AMJAD, A.; AZAM, F.; ANWAR, M. W.; BUTT, W. H.; RASHID, M. Event-driven process chain for modeling and verification of business requirements-a systematic literature review. *IEEE Access*, v. 6, p. 9027–9048, 2018. Disponível em: <<https://doi.org/10.1109/ACCESS.2018.2791666>>.
- 230 CLARK, A. G.; WALKINSHAW, N.; HIERONS, R. M. Test case generation for agent-based models: A systematic literature review. *Inf. Softw. Technol.*, v. 135, p. 106567, 2021. Disponível em: <<https://doi.org/10.1016/j.infsof.2021.106567>>.
- 231 MUSTAFA, A.; WAN-KADIR, W. M. N.; IBRAHIM, N.; SHAH, M. A.; YUNAS, M.; KHAN, A.; ZAREEI, M.; ALANAZI, F. Automated test case generation from requirements: A systematic literature review. *Comput. Mater. Contin.*, Computers, Materials and Continua (Tech Science Press), v. 67, n. 2, p. 1819–1833, 2021.
- 232 BELLENDORF, J.; MANN, Z. Á. Specification of cloud topologies and orchestration using TOSCA: a survey. *Computing*, v. 102, n. 8, p. 1793–1815, 2020. Disponível em: <<https://doi.org/10.1007/s00607-019-00750-3>>.
- 233 RASHID, M.; ANWAR, M. W.; AZAM, F.; KASHIF, M. Model-based requirements and properties specifications trends for early design verification of embedded systems. In: *11th System of Systems Engineering Conference, SoSE 2016, Kongsberg, Norway, June 12-16, 2016*. IEEE, 2016. p. 1–7. Disponível em: <<https://doi.org/10.1109/SYSESE.2016.7542917>>.
- 234 NGUYEN, P. H.; KRAMER, M. E.; KLEIN, J.; TRAON, Y. L. An extensive systematic review on the model-driven development of secure systems. *Inf. Softw. Technol.*, v. 68, p. 62–81, 2015. Disponível em: <<https://doi.org/10.1016/j.infsof.2015.08.006>>.
- 235 OUHBI, S.; IDRI, A.; ALEMÁN, J. L. F.; TOVAL, A. Software quality requirements: A systematic mapping study. In: MUENCHASRI, P.; ROTHERMEL, G. (Ed.). *20th Asia-Pacific Software Engineering Conference, APSEC 2013, Ratchathewi, Bangkok, Thailand, December 2-5, 2013 - Volume 1*. IEEE Computer Society, 2013. p. 231–238. Disponível em: <<https://doi.org/10.1109/APSEC.2013.40>>.

- 236 HEIKKILÄ, V. T.; DAMIAN, D. E.; LASSENIUS, C.; PAASIVAARA, M. A mapping study on requirements engineering in agile software development. In: *41st Euromicro Conference on Software Engineering and Advanced Applications, EUROMICRO-SEAA 2015, Madeira, Portugal, August 26-28, 2015*. IEEE Computer Society, 2015. p. 199–207. Disponível em: <<https://doi.org/10.1109/SEAA.2015.70>>.
- 237 SANTOS, P. S. M. dos; CAETANO, A.; SOUZA, B. P. de; TRAVASSOS, G. H. On the benefits and challenges of using kanban in software engineering: a structured synthesis study. *J. Softw. Eng. Res. Dev.*, v. 6, p. 13, 2018. Disponível em: <<https://doi.org/10.1186/s40411-018-0057-1>>.
- 238 CHUNG, L.; NIXON, B. A.; YU, E.; MYLOPOULOS, J. *Non-Functional Requirements in Software Engineering*. Springer, 2000. v. 5. (International Series in Software Engineering, v. 5). ISBN 978-1-4613-7403-9. Disponível em: <<https://doi.org/10.1007/978-1-4615-5269-7>>.
- 239 BRESCIANI, P.; PERINI, A.; GIORGINI, P.; GIUNCHIGLIA, F.; MYLOPOULOS, J. Tropos: An agent-oriented software development methodology. *Auton. Agent. Multi. Agent. Syst.*, Springer Nature, v. 8, n. 3, p. 203–236, maio 2004.
- 240 FARID, W. M. The NORMAP methodology: Lightweight engineering of non-functional requirements for agile processes. In: LEUNG, K. R. P. H.; MUENCHAISRI, P. (Ed.). *19th Asia-Pacific Software Engineering Conference, APSEC 2012, Hong Kong, China, December 4-7, 2012*. IEEE, 2012. p. 322–325. Disponível em: <<https://doi.org/10.1109/APSEC.2012.23>>.
- 241 NGUYEN, Q. L. Non-functional requirements analysis modeling for software product lines. In: *ICSE Workshop on Modeling in Software Engineering, MiSE 2009, Vancouver, BC, Canada, May 17-18, 2009*. IEEE Computer Society, 2009. p. 56–61. Disponível em: <<https://doi.org/10.1109/MISE.2009.5069898>>.
- 242 RUMBAUGH, J. E.; JACOBSON, I.; BOOCH, G. *The unified modeling language reference manual - covers UML 2.0, Second Edition*. [S.l.]: Addison-Wesley, 2005. (Addison Wesley object technology series). ISBN 978-0-321-24562-5.
- 243 JONKERS, H.; BURREN, R. van; ARBAB, F.; BOER, F. de; BONSAUGUE, M.; BOSMA, H.; DOEST, H. ter; GROENEWEGEN, L.; SCHOLTEN, J.; HOPPENBROUWERS, S.; IACOB, M.-E.; JANSSEN, W.; LANKHORST, M.; LEEUWEN, D. van; PROPER, E.; STAM, A.; TORRE, L. van der; ZANTEN, G. van. Towards a language for coherent enterprise architecture descriptions. In: *Seventh IEEE International Enterprise Distributed Object Computing Conference, 2003. Proceedings*. [S.l.: s.n.], 2003. p. 28–37.
- 244 FAUGÈRE, M.; BOURBEAU, T.; SIMONE, R. de; GÉRARD, S. MARTE: also an UML profile for modeling AADL applications. In: *12th International Conference on Engineering of Complex Computer Systems (ICECCS 2007), 10-14 July 2007, Auckland, New Zealand*. IEEE Computer Society, 2007. p. 359–364. Disponível em: <<https://doi.org/10.1109/ICECCS.2007.29>>.
- 245 CABOT, J.; GOGOLLA, M. Object constraint language (OCL): A definitive guide. In: BERNARDO, M.; CORTELLESA, V.; PIERANTONIO, A. (Ed.). *Formal Methods for Model-Driven Engineering - 12th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2012, Bertinoro, Italy, June 18-23, 2012. Advanced Lectures*. Springer, 2012. (Lecture Notes in Computer Science, v. 7320), p. 58–90. Disponível em: <https://doi.org/10.1007/978-3-642-30982-3_3>.
- 246 MALLET, F. UML profile for MARTE: time model and CDSL. In: ERMOLAYEV, V.; MAYR, H. C.; NIKITCHENKO, M. S.; SPIVAKOVSKY, A.; ZHOLTKEYVYCH, G.; ZAVILEYSKY, M.; KRAVTSOV, H.; KOBETS, V.; PESCHANENKO, V. S. (Ed.). *Proceedings of the 9th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer*,

- Kherson, Ukraine, June 19-22, 2013. CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 1000), p. 289–294. Disponível em: <<https://ceur-ws.org/Vol-1000/ICTERI-2013-p-289-294-T1.pdf>>.
- 247 MOODY, D. L. The physics of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Trans. Software Eng.*, v. 35, n. 6, p. 756–779, 2009. Disponível em: <<https://doi.org/10.1109/TSE.2009.67>>.
- 248 ISSA, A. A.; ALALI, A. I. Automated requirements engineering: Use case patterns-driven approach. *IET Softw.*, v. 5, n. 3, p. 287–303, 2011. Disponível em: <<https://doi.org/10.1049/iet-sen.2010.0014>>.
- 249 BERTOLINO, A.; GNESI, S. Use case-based testing of product lines. In: PAAKKI, J.; INVERARDI, P. (Ed.). *Proceedings of the 11th ACM SIGSOFT Symposium on Foundations of Software Engineering 2003 held jointly with 9th European Software Engineering Conference, ESEC/FSE 2003, Helsinki, Finland, September 1-5, 2003*. ACM, 2003. p. 355–358. Disponível em: <<https://doi.org/10.1145/940071.940120>>.
- 250 GOMAA, H. *Designing software product lines with UML - from use cases to pattern-based software architectures*. [S.l.]: ACM, 2005. ISBN 978-0-201-77595-2.
- 251 BRAGANÇA, A.; MACHADO, R. J. Deriving software product line 's architectural requirements from use cases : an experimental approach. In: . [s.n.], 2005. Disponível em: <<https://api.semanticscholar.org/CorpusID:17901744>>.
- 252 ERIKSSON, M.; BÖRSTLER, J.; BORG, K. The PLUSS approach - domain modeling with features, use cases and use case realizations. In: OBBINK, J. H.; POHL, K. (Ed.). *Software Product Lines, 9th International Conference, SPLC 2005, Rennes, France, September 26-29, 2005, Proceedings*. Springer, 2005. (Lecture Notes in Computer Science, v. 3714), p. 33–44. Disponível em: <https://doi.org/10.1007/11554844_5>.
- 253 GALLINA, B.; GUELFY, N. A template for requirement elicitation of dependable product lines. In: SAWYER, P.; PAECH, B.; HEYMANS, P. (Ed.). *Requirements Engineering: Foundation for Software Quality, 13th International Working Conference, REFSQ 2007, Trondheim, Norway, June 11-12, 2007, Proceedings*. Springer, 2007. (Lecture Notes in Computer Science, v. 4542), p. 63–77. Disponível em: <https://doi.org/10.1007/978-3-540-73031-6_5>.
- 254 CHOI, W.; KANG, S.; CHOI, H.; BAIK, J. Automated generation of product use case scenarios in product line development. In: WU, Q.; HE, X.; NGUYEN, Q. V.; JIA, W.; HUANG, M. L. (Ed.). *Proceedings of 8th IEEE International Conference on Computer and Information Technology, CIT 2008, Sydney, Australia, July 8-11, 2008*. IEEE Computer Society, 2008. p. 760–765. Disponível em: <<https://doi.org/10.1109/CIT.2008.4594770>>.
- 255 ANTHONYSAMY, P.; SOMÉ, S. S. Aspect-oriented use case modeling for software product lines. In: WHITTLE, J.; MUSSBACHER, G. (Ed.). *Proceedings of the 2008 AOSD workshop on Early aspects, EA '08, Brussels, Belgium, March 31, 2008*. ACM, 2008. p. 5:1–5:8. Disponível em: <<https://doi.org/10.1145/1404946.1404951>>.
- 256 JIRAPANTHONG, W. Analysis on relationships among software models through traceability activity. In: PAPASRATORN, B.; CHUTIMASKUL, W.; PORKAEW, K.; VANIJJA, V. (Ed.). *Advances in Information Technology - Third International Conference, IAIT 2009, Bangkok, Thailand, December 1-5, 2009. Proceedings*. Springer, 2009. (Communications in Computer and Information Science, v. 55), p. 71–80. Disponível em: <https://doi.org/10.1007/978-3-642-10392-6_8>.
- 257 BONIFÁCIO, R.; BORBA, P. Modeling scenario variability as crosscutting mechanisms. In: SULLIVAN, K. J.; MOREIRA, A.; SCHWANNINGER, C.; GRAY, J. (Ed.). *Proceedings of the 8th International Conference on Aspect-Oriented Software Development, AOSD 2009*,

Charlottesville, Virginia, USA, March 2-6, 2009. ACM, 2009. p. 125–136. Disponível em: <<https://doi.org/10.1145/1509239.1509258>>.

258 OLIVEIRA, R. P. de; INSFRÁN, E.; ABRAHÃO, S.; GONZALEZ-HUERTA, J.; BLANES, D.; COHEN, S.; ALMEIDA, E. S. de. A feature-driven requirements engineering approach for software product lines. In: *2013 VII Brazilian Symposium on Software Components, Architectures and Reuse, Brasília, DF, Brazil, September 29 - October 4, 2013*. IEEE Computer Society, 2013. p. 1–10. Disponível em: <<https://doi.org/10.1109/SBCARS.2013.11>>.

259 DARDENNE, A.; LAMSWEERDE, A. van; FICKAS, S. Goal-directed requirements acquisition. *Sci. Comput. Program.*, v. 20, n. 1-2, p. 3–50, 1993. Disponível em: <[https://doi.org/10.1016/0167-6423\(93\)90021-G](https://doi.org/10.1016/0167-6423(93)90021-G)>.

260 WANDERLEY, F.; SILVA, A.; ARAÚJO, J.; SILVEIRA, D. S. da. Snapmind: A framework to support consistency and validation of model-based requirements in agile development. In: MOREIRA, A.; SÁNCHEZ, P.; MUSSBACHER, G.; ARAÚJO, J. (Ed.). *IEEE 4th International Model-Driven Requirements Engineering Workshop, MoDRE 2014, 25 August, 2014, Karlskrona, Sweden*. IEEE Computer Society, 2014. p. 47–56. Disponível em: <<https://doi.org/10.1109/MoDRE.2014.6890825>>.

261 MISHRA, A. *IOS code testing test-driven development and behavior-driven development with swift*. [S.l.]: Apress, 2017.

262 MAGUIRE, M. Using human factors standards to support user experience and agile design. In: STEPHANIDIS, C.; ANTONA, M. (Ed.). *Universal Access in Human-Computer Interaction. Design Methods, Tools, and Interaction Techniques for eInclusion*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. p. 185–194. ISBN 978-3-642-39188-0.

263 ISO/IEC/IEEE International Standard - Information technology - Security techniques - Privacy framework. *ISO/IEC/IEEE 29100:2011*, p. 1–21, 2011.

264 ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes – Requirements engineering. *ISO/IEC/IEEE 29148:2018(E)*, p. 1–104, 2018.

265 IEEE Recommended Practice for Software Requirements Specifications. *IEEE Std 830-1998*, p. 1–40, 1998.

266 YU, E. S. K. *Modelling strategic relationships for process reengineering*. Tese (Doutorado) — University of Toronto, Canada, 1995. Disponível em: <https://librarysearch.library.utoronto.ca/permalink/01UTORONTO_INST/14bjeso/alma991106020988606196>.

267 MAHMUD, I.; VENEZIANO, V. Mind-mapping: An effective technique to facilitate requirements engineering in agile software development. In: *14th International Conference on Computer and Information Technology (ICCIT 2011)*. [S.l.: s.n.], 2011. p. 157–162.

268 GABOROVIĆ, A.; KARIĆ, K.; BLAGOJEVIĆ, M.; PLAŠIĆ, J. Comparative analysis of iso/iec and iee standards in the field of internet of things. University of Kragujevac, Faculty of Technical Sciences, Čačak, 2022.

269 SCHOUTEN, D. G. M.; MASSINK, P.; DONKER, S. F.; NEERINCX, M. A.; CREMERS, A. H. M. Using scaffolding to formalize digital coach support for low-literate learners. *User Model. User Adapt. Interact.*, v. 31, n. 2, p. 183–223, 2021. Disponível em: <<https://doi.org/10.1007/s11257-020-09278-0>>.

270 TIAN, J.; TU, Z.; WANG, Z.; XU, X.; LIU, M. User intention recognition and requirement elicitation method for conversational AI services. In: *2020 IEEE International Conference on Web Services, ICWS 2020, Beijing, China, October 19-23, 2020*. IEEE, 2020. p. 273–280. Disponível em: <<https://doi.org/10.1109/ICWS49710.2020.00042>>.

- 271 IHSANI, N. S.; BAIZAL, Z. K. A.; IKHSAN, N. Conversational recommender system based on functional requirements and technical specifications. In: *2021 International Conference on Data Science and Its Applications (ICoDSA)*. IEEE, 2021. Disponível em: <<http://dx.doi.org/10.1109/ICoDSA53588.2021.9617539>>.
- 272 SPILIOTOPOULOS, D.; STAVROPOULOU, P.; KOUROUPETROGLOU, G. Spoken dialogue interfaces: Integrating usability. In: HOLZINGER, A.; MIESENBERGER, K. (Ed.). *HCI and Usability for e-Inclusion, 5th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2009, Linz, Austria, November 9-10, 2009 Proceedings*. Springer, 2009. (Lecture Notes in Computer Science, v. 5889), p. 484–499. Disponível em: <https://doi.org/10.1007/978-3-642-10308-7_36>.
- 273 LEWIS, L. Avatars and robots as social companions in healthcare: Requirements, engineering, adoption and ethics. *Int. J. Enterp. Inf. Syst.*, v. 10, n. 2, p. 21–39, 2014. Disponível em: <<https://doi.org/10.4018/ijeis.2014040102>>.
- 274 GNEWUCH, U.; MORANA, S.; MÄDCHE, A. Towards designing cooperative and social conversational agents for customer service. In: KIM, Y. J.; AGARWAL, R.; LEE, J. K. (Ed.). *Proceedings of the International Conference on Information Systems - Transforming Society with Digital Innovation, ICIS 2017, Seoul, South Korea, December 10-13, 2017*. Association for Information Systems, 2017. Disponível em: <<http://aisel.aisnet.org/icis2017/HCI/Presentations/1>>.
- 275 CASTALDO, N.; DANIEL, F.; MATERA, M.; ZACCARIA, V. Conversational data exploration. In: BAKAEV, M.; FRASINCAR, F.; KO, I. (Ed.). *Web Engineering - 19th International Conference, ICWE 2019, Daejeon, South Korea, June 11-14, 2019, Proceedings*. Springer, 2019. (Lecture Notes in Computer Science, v. 11496), p. 490–497. Disponível em: <https://doi.org/10.1007/978-3-030-19274-7_34>.
- 276 GILSON, F.; ANNAND, S.; STEEL, J. Recording software design decisions on the fly. In: WADHWA, B.; CHAWLA, S.; GAN, B.; OUH, E. L.; MUENCHAISRI, P.; TIWARI, S.; RATHORE, S. S. (Ed.). *Joint Proceedings of SEED & NLPaSE co-located with 27th Asia Pacific Software Engineering Conference 2020, Singapore [Virtual], December 1, 2020*. CEUR-WS.org, 2020. (CEUR Workshop Proceedings, v. 2799), p. 53–66. Disponível em: <https://ceur-ws.org/Vol-2799/Paper7_NLPaSE.pdf>.
- 277 SPILIOTOPOULOS, D.; KOTIS, K.; VASSILAKIS, C.; MARGARIS, D. Semantics-driven conversational interfaces for museum chatbots. In: RAUTERBERG, M. (Ed.). *Culture and Computing - 8th International Conference, C&C 2020, Held as Part of the 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19-24, 2020, Proceedings*. Springer, 2020. (Lecture Notes in Computer Science, v. 12215), p. 255–266. Disponível em: <https://doi.org/10.1007/978-3-030-50267-6_20>.
- 278 HAN, H. J.; MENDU, S.; JAWORSKI, B. K.; OWEN, J. E.; ABDULLAH, S. Ptsdialogue: Designing a conversational agent to support individuals with post-traumatic stress disorder. In: DORYAB, A.; LV, Q.; BEIGL, M. (Ed.). *UbiComp/ISWC '21: 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and 2021 ACM International Symposium on Wearable Computers, Virtual Event, September 21-25, 2021*. ACM, 2021. p. 198–203. Disponível em: <<https://doi.org/10.1145/3460418.3479332>>.
- 279 LIE, M. F.; KVALVIK, P. Virtual assistant: A multi-paradigm dialog workflow system for visitor registration during a pandemic situation. In: AHAM, T. Z.; TAÏAR, R.; GROFF, F. (Ed.). *Human Interaction, Emerging Technologies and Future Applications IV - Proceedings of the 4th International Conference on Human Interaction and Emerging Technologies: Future Applications (IHiet - AI 2021), April 28-30, 2021, Strasbourg, France*. Springer, 2021. (Advances in Intelligent Systems and Computing, v. 1378), p. 162–169. Disponível em: <https://doi.org/10.1007/978-3-030-74009-2_21>.

- 280 MULGUND, P.; SHARMAN, R.; RIFKIN, D.; MARRAZZO, S. Design, development, and evaluation of a telemedicine platform for patients with sleep apnea (ognoMy): Design science research approach. *JMIR Formative Research*, JMIR Publications Inc., v. 5, n. 7, p. e26059, jul. 2021. ISSN 2561-326X. Disponível em: <<http://dx.doi.org/10.2196/26059>>.
- 281 RIVERO, L.; PORTELA, C.; BOARO, J. M. C.; SANTOS, P. T. C. dos; REGO, V.; JUNIOR, G. B.; PAIVA, A. C. de; ALVES, E. W. B. A. L.; OLIVEIRA, M. S. L.; MORAES, R. U. M. e; MENDES, M. Lessons learned from applying requirements and design techniques in the development of a machine learning system for predicting lawsuits against power companies. In: YAMAMOTO, S.; MORI, H. (Ed.). *Human Interface and the Management of Information. Information Presentation and Visualization - Thematic Area, HIMI 2021, Held as Part of the 23rd HCI International Conference, HCII 2021, Virtual Event, July 24-29, 2021, Proceedings, Part I*. Springer, 2021. (Lecture Notes in Computer Science, v. 12765), p. 227–243. Disponível em: <https://doi.org/10.1007/978-3-030-78321-1_18>.
- 282 VÖLKEL, S. T.; BUSCHEK, D.; EIBAND, M.; COWAN, B. R.; HUSSMANN, H. Eliciting and analysing users' envisioned dialogues with perfect voice assistants. In: KITAMURA, Y.; QUIGLEY, A.; ISBISTER, K.; IGARASHI, T.; BJØRN, P.; DRUCKER, S. M. (Ed.). *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*. ACM, 2021. p. 254:1–254:15. Disponível em: <<https://doi.org/10.1145/3411764.3445536>>.
- 283 WAMBSGANSS, T.; HÖCH, A.; ZIERAU, N.; SÖLLNER, M. Ethical design of conversational agents: Towards principles for a value-sensitive design. In: *Innovation durch Informationssysteme - WI als zukunftsweisende Wissenschaft, 16. Internationale Tagung Wirtschaftsinformatik (WI 2021), March 09-11, 2021, Universität Duisburg-Essen, Germany*. AISEL, 2021. Disponível em: <<https://aisel.aisnet.org/wi2021/ZMethods/Track01/2>>.
- 284 ABBAS, T.; GADIRAJU, U.; KHAN, V.; MARKOPOULOS, P. Understanding user perceptions of response delays in crowd-powered conversational systems. *Proc. ACM Hum. Comput. Interact.*, v. 6, n. CSCW2, p. 1–42, 2022. Disponível em: <<https://doi.org/10.1145/3555765>>.
- 285 KHOSRAWI-RAD, B.; SCHLIMBACH, R.; STROHMANN, T.; ROBBA-BISSANTZ, S. Design knowledge for virtual learning companions. In: *Proceedings of the AIS SIGED 2022 Conference*. [S.l.: s.n.], 2022.
- 286 PALESE, B.; PICKARD, M.; BARTOSIAK, M. Can we help the bots? towards an evaluation of their performance and the creation of human enhanced artifact for emotions de-escalation. In: *55th Hawaii International Conference on System Sciences, HICSS 2022, Virtual Event / Maui, Hawaii, USA, January 4-7, 2022*. ScholarSpace, 2022. p. 1–10. Disponível em: <<http://hdl.handle.net/10125/79375>>.
- 287 STIEGLITZ, S.; HOFEDITZ, L.; BRÜNKER, F.; EHNIS, C.; MIRBABAIE, M.; ROSS, B. Design principles for conversational agents to support emergency management agencies. *Int. J. Inf. Manag.*, v. 63, p. 102469, 2022. Disponível em: <<https://doi.org/10.1016/j.ijinfomgt.2021.102469>>.
- 288 VOLKMANN, N.; SCHMIDT, S. Design pattern for conversational agents handling data-driven requests. In: *Gemeinschaften in Neuen Medien. Digitalität und Diversität. Mit digitaler Transformation Barrieren überwinden!?* Technische Universität Dresden, 2022. (GeNeMe'22). Disponível em: <<http://dx.doi.org/10.25368/2023.72>>.
- 289 GEBBING, P.; KHOSRAWI-RAD, B.; STROHMANN, T.; WINDOLF, C.; LATTEMANN, C. A virtual companion for lifelong learning – design principles for motivation, social learning, and exam preparation. In: *36th Bled eConference – Digital Economy and Society: The Balancing Act for Digital Innovation in Times of Instability: June 25 – 28, 2023, Bled, Slovenia, Conference Proceedings*. University of Maribor Press, 2023. (BeC). Disponível em: <<http://dx.doi.org/10.18690/um.fov.6.2023.13>>.

- 290 SCHOUTEN, D. G. M.; DENEKA, A. A.; THEUNE, M.; NEERINCX, M. A.; CREMERS, A. H. M. An embodied conversational agent coach to support societal participation learning by low-literate users. *Univers. Access Inf. Soc.*, v. 22, n. 4, p. 1215–1241, 2023. Disponível em: <<https://doi.org/10.1007/s10209-021-00865-5>>.
- 291 SILVA, G. R. S.; RODRIGUES, G. N.; CANEDO, E. D. A modeling strategy for the verification of context-oriented chatbot conversational flows via model checking. *JUCS - Journal of Universal Computer Science*, Pensoft Publishers, v. 29, n. 7, p. 805–835, jul. 2023. ISSN 0948-695X. Disponível em: <<http://dx.doi.org/10.3897/jucs.91311>>.
- 292 SONNTAG, M.; MEHMANN, J.; TEUTEBERG, F. Deriving trust-supporting design knowledge for ai-based chatbots in customer service: A use case from the automotive industry. *J. Organ. Comput. Electron. Commer.*, v. 33, n. 3-4, p. 178–210, 2023. Disponível em: <<https://doi.org/10.1080/10919392.2023.2276631>>.
- 293 GÖKER, M. H.; THOMPSON, C. A. Personalized conversational case-based recommendation. In: BLANZIERI, E.; PORTINALE, L. (Ed.). *Advances in Case-Based Reasoning, 5th European Workshop, EWCBR 2000, Trento, Italy, September 6-9, 2000, Proceedings*. Springer, 2000. (Lecture Notes in Computer Science, v. 1898), p. 99–111. Disponível em: <https://doi.org/10.1007/3-540-44527-7_10>.
- 294 BRANTING, K.; LESTER, J.; MOTT, B. Dialogue management for conversational case-based reasoning. In: _____. *Advances in Case-Based Reasoning*. Springer Berlin Heidelberg, 2004. p. 77–90. ISBN 9783540286318. Disponível em: <http://dx.doi.org/10.1007/978-3-540-28631-8_7>.
- 295 GEBHARD, P.; KLESEN, M.; RIST, T. Coloring multi-character conversations through the expression of emotions. In: ANDRÉ, E.; DYBKJÆR, L.; MINKER, W.; HEISTERKAMP, P. (Ed.). *Affective Dialogue Systems, Tutorial and Research Workshop, ADS 2004, Kloster Irsee, Germany, June 14-16, 2004, Proceedings*. Springer, 2004. (Lecture Notes in Computer Science, v. 3068), p. 128–141. Disponível em: <https://doi.org/10.1007/978-3-540-24842-2_13>.
- 296 HIRTH, J.; SCHMITZ, N.; BERNS, K. Towards social robots: Designing an emotion-based architecture. *Int. J. Soc. Robotics*, v. 3, n. 3, p. 273–290, 2011. Disponível em: <<https://doi.org/10.1007/s12369-010-0087-2>>.
- 297 LI, J.; CHIGNELL, M. H. Communication of emotion in social robots through simple head and arm movements. *Int. J. Soc. Robotics*, v. 3, n. 2, p. 125–142, 2011. Disponível em: <<https://doi.org/10.1007/s12369-010-0071-x>>.
- 298 MEJIA, A. L. M. Robotic interfaces design - avatar and GUI competing for older user's attention. In: ZHOU, J.; SALVENDY, G. (Ed.). *Human Aspects of IT for the Aged Population. Design for Everyday Life - First International Conference, ITAP 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2-7, 2015. Proceedings, Part II*. Springer, 2015. (Lecture Notes in Computer Science, v. 9194), p. 300–310. Disponível em: <https://doi.org/10.1007/978-3-319-20913-5_28>.
- 299 RABELO, R. J.; ZAMBIASI, S. P.; ROMERO, D. Collaborative softbots: Enhancing operational excellence in systems of cyber-physical systems. In: CAMARINHA-MATOS, L. M.; AFSARMANESH, H.; ANTONELLI, D. (Ed.). *Collaborative Networks and Digital Transformation - 20th IFIP WG 5.5 Working Conference on Virtual Enterprises, PRO-VE 2019, Turin, Italy, September 23-25, 2019, Proceedings*. Springer, 2019. (IFIP Advances in Information and Communication Technology, v. 568), p. 55–68. Disponível em: <https://doi.org/10.1007/978-3-030-28464-0_6>.
- 300 ALEXIADIS, A.; NIZAMIS, A.; KOSKINAS, I.; IOANNIDIS, D.; VOTIS, K.; TZOVARAS, D. Applying an intelligent personal agent on a smart home using a novel dialogue generator. In: MAGLOGIANNIS, I.; ILIADIS, L.; PIMENIDIS, E. (Ed.). *Artificial Intelligence Applications and*

- Innovations - 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5-7, 2020, Proceedings, Part II*. Springer, 2020. (IFIP Advances in Information and Communication Technology, v. 584), p. 384–395. Disponível em: <https://doi.org/10.1007/978-3-030-49186-4_32>.
- 301 LAGO, A. S.; DIAS, J. P.; FERREIRA, H. S. Conversational interface for managing non-trivial internet-of-things systems. In: KRZHIZHANOVSKAYA, V. V.; ZÁVODSZKY, G.; LEES, M. H.; DONGARRA, J. J.; SLOOT, P. M. A.; BRISSOS, S.; TEIXEIRA, J. (Ed.). *Computational Science - ICCS 2020 - 20th International Conference, Amsterdam, The Netherlands, June 3-5, 2020, Proceedings, Part V*. Springer, 2020. (Lecture Notes in Computer Science, v. 12141), p. 384–397. Disponível em: <https://doi.org/10.1007/978-3-030-50426-7_29>.
- 302 XU, Y.; GU, S.; LI, H.; MIN, F. A hybrid approach to three-way conversational recommendation. *Soft Comput.*, v. 26, n. 24, p. 13885–13897, 2022. Disponível em: <<https://doi.org/10.1007/s00500-022-07416-x>>.
- 303 GHOSE, S.; BARUA, J. J. Toward the implementation of a topic specific dialogue based natural language chatbot as an undergraduate advisor. In: *2013 International Conference on Informatics, Electronics and Vision (ICIEV)*. [S.l.: s.n.], 2013. p. 1–5.
- 304 ALMAHRI, F. A. A. J.; BELL, D.; ARZOKY, M. Personas design for conversational systems in education. *Informatics*, v. 6, n. 4, p. 46, 2019. Disponível em: <<https://doi.org/10.3390/informatics6040046>>.
- 305 EDIRISOORIYA, M.; MAHAKALANDA, I.; YAPA, T. Generalised framework for automated conversational agent design via qfd. In: *2019 Moratuwa Engineering Research Conference (MERCCon)*. IEEE, 2019. Disponível em: <<http://dx.doi.org/10.1109/MERCCon.2019.8818945>>.
- 306 BERKA, J.; BALATA, J.; JONKER, C. M.; MIKOVEC, Z.; RIEMSDIJK, M. B. van; TIELMAN, M. L. Misalignment in semantic user model elicitation via conversational agents: A case study in navigation support for visually impaired people. *International Journal of Human–Computer Interaction*, Informa UK Limited, v. 38, n. 18–20, p. 1909–1925, abr. 2022. ISSN 1532-7590. Disponível em: <<http://dx.doi.org/10.1080/10447318.2022.2059925>>.
- 307 CHEN, L.; ZHANG, Z.; ZHANG, X.; ZHAO, L. A pilot study for understanding users’ attitudes towards a conversational agent for news recommendation. In: HALVEY, M.; FOSTER, M. E.; DALTON, J.; MUNTEANU, C.; TRIPPAS, J. (Ed.). *CUI 2022: 4th Conference on Conversational User Interfaces, Glasgow, United Kingdom, July 26 - 28, 2022*. ACM, 2022. p. 36:1–36:6. Disponível em: <<https://doi.org/10.1145/3543829.3544530>>.
- 308 LARBI, D.; DENECKE, K.; GABARRON, E. Usability testing of a social media chatbot for increasing physical activity behavior. *Journal of Personalized Medicine*, MDPI AG, v. 12, n. 5, p. 828, maio 2022. ISSN 2075-4426. Disponível em: <<http://dx.doi.org/10.3390/jpm12050828>>.
- 309 DENECKE, K. Framework for guiding the development of high-quality conversational agents in healthcare. *Healthcare*, MDPI AG, v. 11, n. 8, p. 1061, abr. 2023. ISSN 2227-9032. Disponível em: <<http://dx.doi.org/10.3390/healthcare11081061>>.
- 310 DRAGICEVIC, S.; CELAR, S.; NOVAK, L. Use of method for elicitation, documentation, and validation of software user requirements (medov) in agile software development projects. In: AL-DABASS, D.; AMETI, V.; SKENDERI, F.; HALILI, F. (Ed.). *Sixth International Conference on Computational Intelligence, Communication Systems and Networks, CICSyN 2014, Tetova, Macedonia, May 27-29, 2014*. IEEE, 2014. p. 65–70. Disponível em: <<https://doi.org/10.1109/CICSyN.2014.27>>.

311 KITCHENHAM, B. A.; PFLEEGER, S. L. Principles of survey research: part 3: constructing a survey instrument. *SIGSOFT Softw. Eng. Notes*, Association for Computing Machinery, New York, NY, USA, v. 27, n. 2, p. 20–24, mar 2002. ISSN 0163-5948.