



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIAS E TECNOLOGIAS
CURSO INTERDICIPILINAR EM TECNOLOGIA DA INFORMAÇÃO

JONATHAN RYAN DA SILVA PAIVA

**UM ESTUDO SOBRE O USO DE MODELOS DE LINGUAGEM DE GRANDE
ESCALA NA ATIVIDADE DE LEVANTAMENTO DE REQUISITOS**

PAU DOS FERROS - RN

2025

JONATHAN RYAN DA SILVA PAIVA

**UM ESTUDO SOBRE O USO DE MODELOS DE LINGUAGEM DE GRANDE
ESCALA NA ATIVIDADE DE LEVANTAMENTO DE REQUISITOS**

Monografia apresentada a Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Bacharel Interdisciplinar em Tecnologia da Informação.

Orientador: João Batista de Souza Neto,
Prof. Dr.

PAU DOS FERROS – RN

2025

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

P145e Paiva, Jonathan Ryan da Silva.
Um estudo sobre o uso de modelos de linguagem
de grande escala na atividade de levantamento de
requisitos / Jonathan Ryan da Silva Paiva. - 2025.
81 f. : il.

Orientador: João Batista de Souza Neto.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Tecnologia da
Informação, 2025.

1. Engenharia de requisitos. 2. Modelos de
linguagem de grande escala. 3. LLMs. 4.
Levantamento de requisitos. 5. Estudo de caso. I.
Souza Neto, João Batista de, orient. II. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade
com AACR2 e os dados fornecidos pelo(a) autor(a).

Biblioteca Campus Pau dos Ferros
Bibliotecária: Erlanda Maria Lopes da Silva
CRB: 15/966

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

JONATHAN RYAN DA SILVA PAIVA

**UM ESTUDO SOBRE O USO DE MODELOS DE LINGUAGEM DE GRANDE
ESCALA NA ATIVIDADE DE LEVANTAMENTO DE REQUISITOS**

Monografia apresentada a Universidade
Federal Rural do Semi-Árido como requisito
para obtenção do título de Bacharel
Interdisciplinar em Tecnologia da Informação.

Defendida em: 05 / 08 / 2025.

BANCA EXAMINADORA

João Batista de Souza Neto, Prof. Dr. (UFERSA)
Presidente

Geiser Chalco Chalco, Prof. Dr. (UFERSA)
Membro Examinador

Reudismam Rolim de Sousa, Prof. Dr. (UFERSA)
Membro Examinador

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, por me conceder a força e a resiliência necessárias para superar os desafios e por iluminar meu caminho até a conclusão desta importante etapa da minha vida.

À minha família, minha base e meu porto seguro, expresso minha imensa gratidão. Vocês sempre acreditaram no meu sonho de ser um estudante universitário e me apoiaram com todos os recursos para que eu pudesse chegar até aqui.

Deixo um agradecimento que transborda o peito à minha avó, a pessoa-chave que tornou tudo isso possível. Vó, sua força foi minha inspiração e seus ensinamentos são o alicerce sobre o qual construo meu futuro. Muito obrigado por tudo.

À minha namorada, dedico um agradecimento especial. Seu amor, paciência e apoio incondicional foram meu refúgio, especialmente, nos momentos mais difíceis. Obrigado por nunca me abandonar nas piores horas e por ser minha maior incentivadora. Estendo minha gratidão à sua família, que sempre me acolheu com tanto carinho.

Agradeço à Universidade Federal Rural do Semi-Árido (UFERSA) pela oportunidade de um ensino público de qualidade e pelas experiências enriquecedoras que levarei para toda a vida.

Por fim, agradeço ao meu orientador, Prof. Dr. João Batista de Souza Neto, pela honra de ser seu orientando. Sua orientação, apoio e conhecimento foram essenciais para que este trabalho alcançasse um excelente resultado. Muito obrigado pela confiança e pela parceria.

RESUMO

A Engenharia de Requisitos ER é uma fase crítica no desenvolvimento de software, porém suscetível a desafios como comunicação ineficaz, ambiguidades e alto custo na correção de erros tardios. Recentemente, os Modelos de Linguagem de Grande Escala LLMs surgiram como uma tecnologia promissora com potencial para otimizar essas atividades. Este trabalho investiga a aplicação de LLMs como ferramenta de apoio na etapa de levantamento de requisitos. Para isso, foi realizada uma revisão da literatura para identificar o estado da arte, seguida pela elaboração de um conjunto de diretrizes práticas para o uso de LLMs em tarefas específicas de elicitación, como compreensão de domínio, identificação de stakeholders e extração de requisitos funcionais e não-funcionais. A eficácia dessas diretrizes foi avaliada empiricamente por meio de um estudo de caso, aplicando-as no contexto do desenvolvimento de um Sistema de Tutoria Inteligente STI-Física. Os resultados indicam que os LLMs, quando utilizados com os *prompts* estruturados propostos, são ferramentas eficazes para acelerar a formalização de artefatos, estruturar informações e reduzir a ambiguidade inicial. Conclui-se que o LLM atua como um assistente de produtividade para o engenheiro de requisitos, otimizando tarefas repetitivas, mas não substitui a necessidade de validação e do pensamento crítico do especialista humano, que permanece central para garantir a qualidade e a adequação dos requisitos.

Palavras-chave: engenharia de requisitos; modelos de linguagem de grande escala; LLMs; levantamento de requisitos; estudo de caso.

ABSTRACT

Requirements Engineering RE is a critical phase in software development, yet it is susceptible to challenges such as ineffective communication, ambiguities, and the high cost of late error correction. Recently, Large Language Models LLMs have emerged as a promising technology with the potential to optimize these activities. This work investigates the application of LLMs as a support tool in the requirements elicitation stage. To this end, a literature review was conducted to identify the state-of-the-art, followed by the development of a set of practical guidelines for using LLMs in specific elicitation tasks, such as domain understanding, stakeholder identification, and the extraction of functional and non-functional requirements. The effectiveness of these guidelines was empirically evaluated through a case study, applying them in the context of developing an Intelligent Tutoring System ITS-Physics. The results indicate that LLMs, when used with the proposed structured prompts, are effective tools for accelerating artifact formalization, structuring information, and reducing initial ambiguity. It is concluded that the LLM acts as a productivity assistant for the requirements engineer, optimizing repetitive tasks, but it does not replace the need for validation and critical thinking by the human expert, which remains central to ensuring the quality and adequacy of the requirements.

Keywords: requirements engineering; large language models; LLMs; requirements elicitation; case study.

LISTA DE FIGURAS

Figura 1	–	Custo relativos para correção de defeitos conforme o ponto de encontro	16
Figura 2	–	Etapas da Engenharia de Requisitos	19
Figura 3	–	Técnicas de Elicitação de Requisitos	25

LISTA DE TABELAS

Quadro 1	– Prompt genérico para compreensão do domínio de negócio.....	38
Quadro 2	– Prompt genérico para identificação de Stakeholder	41
Quadro 3	– Prompt genérico para requisitos funcionais da entrevista.....	44
Quadro 4	– Prompt genérico para identificação de requisitos Não-Funcionais.....	47
Quadro 5	– Prompt genérico para especificação de caso de uso.....	51
Quadro 6	– Prompt genérico para especificação de regras de negócio.....	54
Quadro 7	– Prompt genérico para matriz de rastreabilidade de requisitos.....	59
Quadro 8	– Resultado obtido do prompt.....	61
Quadro 9	– Stakeholders e responsabilidades.....	64

LISTA DE ABREVIATURAS E SIGLAS

AI	Artificial Intelligence (Inteligência Artificial)
BERT	Bidirectional Encoder Representations from Transformers
CPRE	Certified Professional for Requirements Engineering
DER	Documento de Especificação de Requisitos
ER	Engenharia de Requisitos
GPT	Generative Pre-trained Transformer
IA	Inteligência Artificial
ISBN	International Standard Book Number
LLaMA	Large Language Model Meta AI
LLM	<i>Large Language Model</i> (Modelo de Linguagem de Grande Escala)
PLN	Processamento de Linguagem Natural
PMI	<i>Project Management Institute</i>
REAI	<i>Requirements Engineering for Artificial Intelligence</i>

SUMÁRIO

1 INTRODUÇÃO	11
1.1. Problemática	12
1.2. Objetivos.....	13
1.2.1. Objetivos Geral.....	13
1.2.2. Objetivos Específicos	13
1.3 Organização do Trabalho	14
2. FUNDAMENTAÇÃO TEÓRICA.....	15
2.1. Engenharia de Requisitos	15
2.2. Processo de Engenharia de Requisitos.....	15
2.2.1 Elicitação de Requisitos.....	16
2.2.2. Análise de Requisitos	17
2.2.3. Especificação de Requisitos	18
2.2.4. Validação de Requisitos	18
2.2.5. Gerenciamento de Requisitos	19
2.3. Desafios tradicionais da Engenharia de Requisitos.....	19
2.4. Modelos de Linguagem de Grande Escala LLMs.....	24
2.4.1 Definição e Funcionamento.....	24
2.4.2 Evolução e Capacidade.....	25
2.4.3 Aplicação em Engenharia de Software.....	26
3. TRABALHOS RELACIONADOS	27
3.1 How Mature is Requirements Engineering for AI-based Systems? A systematic Mapping Study on Practices, Challenges, and Future Research Directions.....	27
3.2 Uso de LLMs em atividades de Engenharia de Requisitos.....	28
3.3 Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs.....	28

3.4 Enhancing Requirements Engineering Practices Using Large Language Models. University of Gothenburg	29
3.5. Discussões	30
4. METODOLOGIA.....	31
4.1. Revisão Bibliográfica.....	31
4.2. Elaboração de Diretrizes para Uso de LLMs.....	31
4.3 Experimentação Prática.....	32
4.4 Análise dos Resultados	32
5 DIRETRIZES PARA O USO DE LLMs NO LEVANTAMENTO DE REQUISITOS	34
5.1. Compreensão do Domínio do Negócio	34
5.1.1. Resultados Esperados	35
5.1.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.....	36
5.1.3. Resposta do LLM	36
5.2. Identificação de <i>Stakeholders</i>.....	37
5.2.1. Resultados Esperados	38
5.2.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.....	38
5.2.3. Resposta do LLM	39
5.3. Análise de transcrições de entrevista	40
5.3.1. Resultados esperados.....	41
5.3.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.....	42
5.3.3. Resposta do LLM	42
5.4. Identificação de requisitos não funcionais.....	43
5.4.1. Resultados esperados.....	45
5.4.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.....	45
5.4.3. Resposta do LLM	46

5.5. Especificação de caso de uso	47
5.5.1. Resultados esperados.....	48
5.6. Especificação de regras de negócio	51
5.7. Matriz de Rastreabilidade de requisitos.....	55
5.7.1. Resultados esperados.....	56
5.7.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.....	57
5.7.3. Resposta do LLM	58
6 ESTUDO DE CASO	60
6.1. Projeto do Sistema de Tutoria Inteligente para o Ensino de Física.....	60
6.2. Experimentos com LLMs no Levantamento de Requisitos do STI-Física.....	62
6.3. Análise dos Resultados	69
6.4. Discussão dos Resultados e Limitações da Abordagem	71
7. CONSIDERAÇÕES FINAIS.....	73
7.1. Principais contribuições.....	73
7.2. Trabalhos Futuros	76
REFERÊNCIAS.....	78

1 INTRODUÇÃO

O desenvolvimento de sistemas de software é uma atividade marcada por desafios intrínsecos, como a complexibilidade crescente que acompanha seu desenvolvimento. Sommerville (2016) ressalta que os sistemas de software são “abstratos e intangíveis”, podendo ganhar proporções de complexidade tão rapidamente que se tornam incompreensíveis em pouco tempo, elevando os custos com manutenção do software.

Nesse cenário, a Engenharia de Requisitos ER surge como uma disciplina fundamental, responsável por estabelecer a ponte entre as necessidades dos *stakeholders* e a equipe de desenvolvimento, dando início ao processo de criação do software.

A importância crítica da Engenharia de Requisitos é evidenciada pelo alto custo associado a falhas em suas fases iniciais. Conforme destacado por Brooks (1975), corrigir um erro nos requisitos pode custar “100 vezes mais se identificado tardiamente”, o que posiciona o levantamento de requisitos como uma etapa crucial para o sucesso de qualquer projeto de software.

Reinehr (2020), destaca o processo de engenharia de requisitos em cinco etapas principais: elicitação, análise, especificação, validação e gerenciamento. Essas etapas organizam e direcionam a equipe no desenvolver das atividades e diminui a chance de erros iniciais do projeto.

No entanto, mesmo com metodologias estruturadas, hoje ainda persistem grandes lacunas que comprometem a eficiência do processo de ER. Sommerville (2016) destaca que a maioria dos problemas na engenharia de requisitos está diretamente relacionado a comunicação, seja ela de forma direta ou indiretamente.

Os Modelos de Linguagem de Grande Escala LLMs representam uma tecnologia de Inteligência Artificial IA que tem ganhado destaque em diversas áreas do conhecimento. Conforme descrevem Zhao (2023), os LLMs são modelos massivos treinados em vastos volumes de dados textuais, o que lhes confere uma capacidade notável de compreender, sumarizar, traduzir e gerar textos com alta fluência e coerência.

A arquitetura fundamental por trás da maioria desses modelos é a de *Transformers*, introduzida por Vaswani (2017). Essa arquitetura, baseada em mecanismos de atenção (*attention mechanisms*), permite que o modelo processe palavras em relação a todas as outras em uma sequência, capturando contextos complexos e dependências de longa distância. É essa capacidade que habilita os LLMs a produzir respostas úteis e contextualmente relevantes em linguagem natural (Zhao, 2023).

Com os grandes avanços nos LLMs, sua utilização em atividades tem se expandido em vários segmentos. Segundo Habiba (2024), observa-se um aumento significativo no uso de Inteligência Artificial nas atividades de ER.

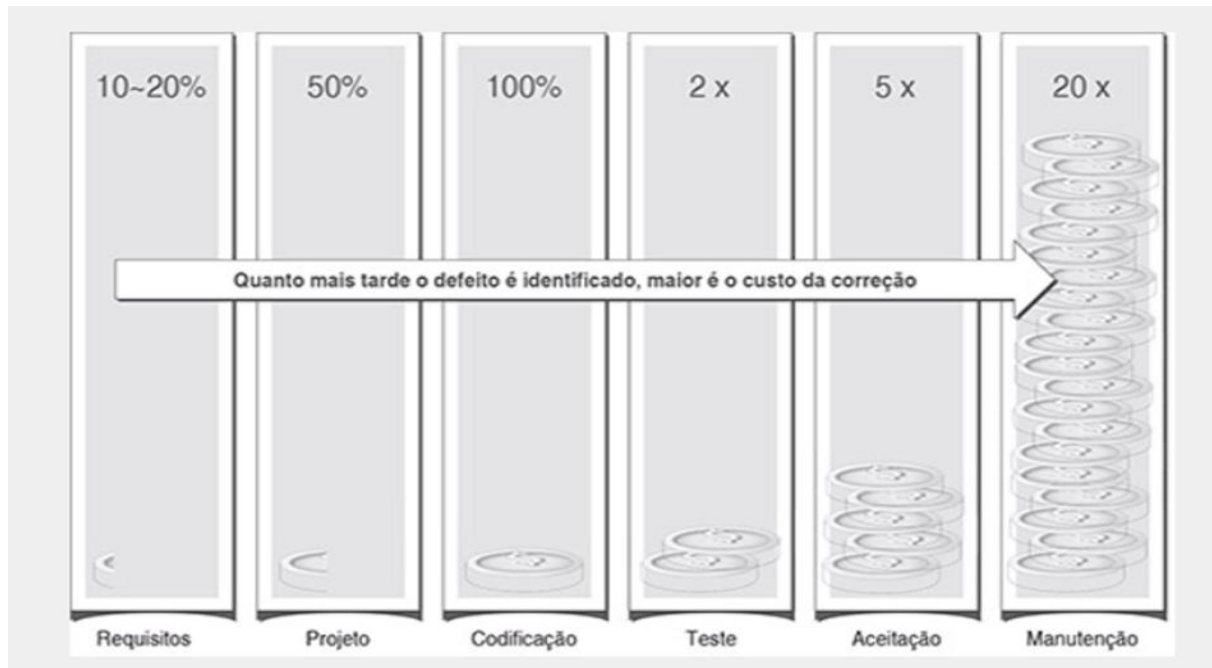
1.1. Problemática

O processo de levantamento de requisitos é um pilar crítico para o sucesso de projetos de software, mas sua execução enfrenta desafios persistentes. Como é destacado por Sommerville (2016), a comunicação frequentemente marcada por ambiguidade e lacunas podem trazer grandes prejuízos quando não se é percebida de imediato, e resolvê-las podem custar caro, a medida que o tempo passa as partes seguirão suas atividades seguros de que a mensagem foi compreendida. Sommerville (2016), ainda destaca que esse desencontro de comunicação somente será notado quando o produto final for entregue ou tiver na fase de teste de aceitação, trazendo grande prejuízos para partes interessadas do projeto.

O Project Management Institute (PMI) apresenta uma constatação de que 47% dos projetos que fracassam, são causados por uma deficiência na atividade de engenharia de requisitos.

Boehm (2001) aponta que o esforço para corrigir um defeito após a entrega do software pode custar um valor consideravelmente cem vezes maior que corrigir este defeito na fase inicial, enquanto ainda se está trabalhando no desenvolvimento dos requisitos do software. Conforme mostra a figura os custos relativos para correções de defeitos aumentam conforme o desenvolvimento do projeto.

Figura 1 – Custo relativos para correção de defeitos conforme o ponto de encontro.



Fonte: Sommerville (2016).

Diante desse cenário, os LLMs emergem como uma ferramenta promissora para solucionar alguns desses desafios enfrentados pela ER.

Apesar do crescimento exponencial dos LLMs como o DeepSeek, (DeepSeek AI), 2024, GPT-4, (OpenAI, 2023), e outros, sua aplicação na ER ainda é incipiente e carece de estudos. Conforme destacado por Marques (2023), a maioria das pesquisas sobre LLMs concentra-se em tarefas genéricas de processamento de linguagem, com poucas investigações sistemáticas sobre seu impacto em etapas críticas do desenvolvimento de software, como a elicitação de requisitos. Essa lacuna acadêmica é corroborada por Habiba (2024), que apontam a necessidade de mais evidências empíricas sobre a eficácia desses modelos em contextos reais de ER.

1.2. Objetivos

1.2.1. Objetivos Geral

O objetivo deste trabalho é explorar o uso de LLMs no auxílio de tarefas de levantamento de requisitos, com foco na etapa de elicitação.

1.2.2. Objetivos Específicos

Para alcançar o objetivo geral dessa proposta de pesquisa, foram identificados os seguintes objetivos específicos.

- Analisar os desafios tradicionais da ER.
- Estudo do estado da arte sobre o uso de LLMs na ER.
- Mapear aplicação de LLMs em ER.
- Propor diretrizes para uso de LLMs na elicitación de requisitos.
- Avaliar a eficiência dos LLMs por meio de um estudo empírico.

1.3 Organização do Trabalho

Este trabalho está estruturado da seguinte forma para apresentar a pesquisa de maneira clara e lógica:

- Capítulo 2 apresenta a Fundamentação Teórica, detalhando os conceitos essenciais da Engenharia de Requisitos, seu processo, desafios tradicionais e, em seguida, introduz os Modelos de Linguagem de Grande Escala, sua definição e funcionamento.
- Capítulo 3 revisa os Trabalhos Relacionados, analisando estudos recentes sobre a aplicação de LLMs e IA na Engenharia de Requisitos, o que ajuda a contextualizar a presente pesquisa e a identificar lacunas na literatura.
- Capítulo 4 descreve a Metodologia, que guiou a pesquisa, incluindo a revisão da literatura, a elaboração das diretrizes, a condução da experimentação prática e a análise dos resultados.
- Capítulo 5 detalha as Diretrizes para Uso de LLMs, que constituem a principal contribuição prática deste trabalho, oferecendo um guia com *prompts* estruturados para diversas atividades de levantamento de requisitos.
- Capítulo 6 apresenta o Estudo de Caso, em que as diretrizes foram aplicadas em um projeto real STI-Física, descrevendo os experimentos realizados e os resultados obtidos em cada atividade.
- Capítulo 7 finaliza o trabalho com as Considerações Finais, resumindo as principais contribuições, validando o alcance dos objetivos e sugerindo direções para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos essenciais para a compreensão do presente trabalho, abordando a Engenharia de Requisitos ER e os Modelos de Linguagem de Grande Escala LLMs, bem como suas intersecções no contexto do desenvolvimento de software de qualidade.

2.1. Engenharia de Requisitos

A Engenharia de Requisitos ER é uma disciplina fundamental no ciclo de vida do desenvolvimento de software, responsável por definir, documentar e gerenciar as necessidades e restrições de um sistema. Ela atua como uma ponte crucial entre as expectativas dos stakeholders e a equipe de desenvolvimento, garantindo que o produto final atenda aos objetivos propostos. Sua importância reside no fato que requisitos bem definidos são essenciais para o sucesso do projeto, evitando retrabalho, custos elevados, atrasos e insatisfação do cliente como destaca Sommerville (2016). A etapa de elicitação por sua vez, é uma das mais críticas, pois nela são identificadas as reais necessidades dos usuários e do negócio.

2.2. Processo de Engenharia de Requisitos

O processo de Engenharia de requisitos é iterativo que utiliza métodos e práticas para identificar, documentar e manter os requisitos de um sistema, sempre com foco nos objetivos do negócio e na qualidade do produto final (Avazquez e Simões, 2016). Ela serve como base para o design e desenvolvimento do software, sendo fundamental para garantir que o sistema atenda às reais necessidades do cliente (Pressman, 2021).

Segundo Reinehr (2020), muitos problemas em projetos de software estão relacionados a falhas nos requisitos. Quando esses não são bem compreendidos ou gerenciados, podem causar baixo desempenho do sistema, retrabalho e aumento de custos.

De forma geral, a Engenharia de Requisitos é composta por duas fases principais: o desenvolvimento e o gerenciamento dos requisitos (Reinehr, 2020). A fase de desenvolvimento inclui atividades como elicitação, análise, especificação e validação. Já o

gerenciamento está presente ao longo de todo o processo, com foco no controle das mudanças e no rastreamento dos requisitos, facilitando decisões futuras.

Figura 2 – Etapas da Engenharia de Requisitos.



Fonte: Reinehr (2020).

A Figura 2 ilustra as principais etapas da Engenharia de Requisitos, essenciais para assegurar que o software desenvolvido esteja alinhado aos objetivos do negócio, minimizando falhas e evitando custos desnecessários. A seguir, cada uma dessas etapas será detalhada. Vale destacar que essas atividades são estruturadas de forma iterativa e incremental, seguindo um modelo em espiral Sommerville (2020).

2.2.1 Elicitação de Requisitos

A elicitação de requisitos é a primeira e uma das etapas mais críticas da Engenharia de Requisitos. Trata-se do processo de descobrir e entender o que os stakeholders como clientes, usuários finais, gerentes e desenvolvedores realmente necessitam do sistema. Mais do que simplesmente coletar informações, essa etapa exige compreensão profunda do contexto de negócio, dos processos envolvidos e das expectativas em relação ao software a ser desenvolvido (Reinehr, 2020).

De acordo com Sommerville (2020), a elicitação é um processo colaborativo que pode ser afetado por diversos fatores, como a falta de clareza nos objetivos dos stakeholders, a comunicação inadequada ou a dificuldade dos usuários em expressar o que realmente

desejam. Por isso, o analista de requisitos precisa desenvolver habilidades interpessoais e ter sensibilidade para lidar com diferentes perfis de usuários.

Reinehr (2020) destaca que a comunicação ineficaz é uma das principais causas de requisitos mal definidos, incompletos ou incorretos. Esses problemas, se não forem identificados a tempo, podem gerar retrabalho, atrasos e elevação dos custos do projeto, especialmente quando impactam partes estruturais do sistema.

Pressman (2021) reforça que compreender as metas de negócio dos stakeholders é essencial na elicitação, pois essas metas ajudam a orientar a priorização dos requisitos e a resolver eventuais conflitos entre os envolvidos. A elicitação não deve focar apenas nos aspectos técnicos, mas também considerar o valor que o sistema precisa entregar.

Diversas técnicas podem ser utilizadas para realizar a elicitação de forma eficaz. Entre elas incluem entrevistas, workshops, questionários, análise de documentos e observação.

2.2.2. Análise de Requisitos

Após a elicitação, os requisitos, muitas vezes, se encontram em um estado bruto: uma coleção de ideias, necessidades e restrições expressas em linguagem natural, potencialmente ambíguas, incompletas e até mesmo contraditórias. A etapa de Análise de Requisitos é o processo investigativo que transforma essa informação bruta em um modelo coeso e compreensível. Seu objetivo principal é refinar, estruturar e priorizar os requisitos elicitados.

Segundo Pressman (2021), a análise de requisitos permite ao engenheiro de software identificar e resolver conflitos entre as necessidades de diferentes stakeholders. Por exemplo, o departamento de marketing pode desejar um sistema com muitas funcionalidades visuais, enquanto a equipe de TI pode priorizar o desempenho e a segurança. A análise é o momento de negociar essas diferenças e encontrar um equilíbrio que atenda aos objetivos estratégicos do projeto.

Nesta fase, os requisitos são classificados, agrupados e detalhados. Conforme destaca Sommerville (2020), atividades comuns na análise incluem:

- **Classificação:** Organizar os requisitos em diferentes categorias, como funcionais (o que o sistema faz), não funcionais (como o sistema se comporta) e de domínio (regras do negócio).
- **Modelagem:** Criar representações gráficas do sistema, como diagramas de caso de uso, modelos de dados ou diagramas de fluxo, para facilitar a compreensão e a comunicação entre a equipe e os stakeholders.

- **Negociação e Priorização:** Resolver conflitos e definir a ordem de importância dos requisitos, garantindo que os recursos de desenvolvimento sejam focados nas funcionalidades de maior valor para o negócio.
- **Verificação de Consistência:** Garantir que os requisitos não entrem em conflito uns com os outros e que formem um conjunto lógico e viável.

Em suma, a análise de requisitos é uma atividade de pensamento crítico que adiciona clareza e estrutura, preparando o terreno para a etapa de Especificação, onde os requisitos, já refinados e acordados, serão formalmente documentados.

2.2.3. Especificação de Requisitos

A especificação de requisitos consiste em documentar formalmente os requisitos levantados e analisados de forma clara, concisa e livre de ambiguidades. O objetivo é criar um artefato que sirva de base tanto para o desenvolvimento quanto para a validação do sistema (Pressman, 2021).

Segundo Sommerville (2020), a especificação deve descrever de forma detalhada o que o sistema deve fazer, e não como ele fará. Essa documentação pode assumir diferentes formas, como o Documento de Especificação de Requisitos (DER), histórias de usuário ou modelos formais, dependendo do tipo de projeto e do nível de formalidade exigido.

Reinehr (2020) reforça que o uso de linguagens padronizadas, como a UML (Unified Modeling Language), facilita a compreensão dos requisitos por diferentes perfis de profissionais, além de permitir que se identifiquem falhas mais rapidamente. Uma boa especificação também favorece a rastreabilidade dos requisitos e o acompanhamento de sua implementação ao longo do projeto.

2.2.4. Validação de Requisitos

A validação de requisitos tem como objetivo garantir que os requisitos documentados estejam corretos, completos, consistentes e realmente representem as necessidades dos stakeholders. Pressman (2021) destaca que essa etapa é vital para evitar retrabalho e prejuízos decorrentes de erros que, se descobertos mais adiante, são muito mais custosos de corrigir.

Reinehr (2020) afirma que, para validar requisitos, podem ser utilizadas diversas técnicas, como revisões formais, prototipagem, simulações, inspeções, reuniões com stakeholders e testes de aceitação. Essas atividades ajudam a confirmar se os requisitos estão

alinhados com os objetivos do projeto e se são compreendidos de forma unificada por todos os envolvidos.

Sommerville (2020) complementa que a validação deve ocorrer de forma contínua e não apenas ao final da etapa de especificação, permitindo a identificação precoce de falhas. O envolvimento dos stakeholders na validação é indispensável, pois são eles que têm o conhecimento de negócio necessário para julgar a adequação dos requisitos.

2.2.5. Gerenciamento de Requisitos

O gerenciamento de requisitos acompanha todas as fases do ciclo de vida do projeto e visa garantir que os requisitos sejam controlados, rastreados e adaptados conforme necessário. Reinehr (2020) define o gerenciamento como o processo de manter a integridade dos requisitos ao longo do tempo, lidando com mudanças inevitáveis e controlando seu impacto no sistema.

Segundo Pressman (2021), o gerenciamento envolve a identificação de cada requisito, o controle de versões, a análise de impacto de alterações e o rastreamento desde sua origem até a implementação. A rastreabilidade é uma característica essencial, pois permite saber quais partes do sistema são afetadas por mudanças em determinado requisito.

Sommerville (2020) observa que, em ambientes ágeis, o gerenciamento pode ser menos formal, mas ainda assim é necessário um controle rigoroso das mudanças para garantir a qualidade e a entrega de valor contínuo ao cliente. Ferramentas específicas, como repositórios de requisitos ou sistemas de controle de mudanças, auxiliam nesse processo, especialmente em projetos de grande porte.

2.3. Desafios tradicionais da Engenharia de Requisitos

Apesar de sua importância no desenvolvimento de software, a Engenharia de Requisitos enfrenta uma série de desafios que podem comprometer a qualidade final e a satisfação dos stakeholders. Vazques e Simões (2016), apontam alguns dos principais obstáculos enfrentados por analistas e equipe de desenvolvimento durante esse processo.

Comunicação Ineficaz: A comunicação é um pilar no desenvolvimento de software, atuando como a ponte que conecta as ideias dos stakeholders com a equipe de desenvolvimento. Falhas nessa comunicação são uma das principais e mais comuns causas de problemas em projetos. Reinehr (2020) destaca que a falha na comunicação entre stakeholders

e a equipe técnica é uma das razões mais frequentes para retrabalho e mudanças tardias no projeto. A dificuldade pode surgir da tradução de conceitos técnicos para um público não especializado, do uso de termos com significados diferentes para cada grupo, ou de barreiras linguísticas e culturais.

Uma comunicação ineficaz leva a mal-entendidos e requisitos que não refletem as reais necessidades do cliente. Sommerville (2020) reforça que os analistas precisam não apenas ouvir os stakeholders, mas também garantir que as informações obtidas foram corretamente compreendidas e registradas. Para mitigar este desafio, é crucial estabelecer um diálogo constante e claro, utilizando técnicas como workshops, prototipagem e documentação detalhada. A escuta ativa, por parte dos analistas, é uma habilidade fundamental para compreender verdadeiramente o que está sendo solicitado e evitar interpretações equivocadas que comprometem o resultado final.

Requisitos Ambíguos e Incompletos: A ambiguidade e a incompletude são inimigas da precisão na Engenharia de Requisitos. A utilização da linguagem natural, embora comum, é um fator que contribui diretamente para a ambiguidade. Termos genéricos ou mal definidos, como "amigável" ou "rápido", podem ser interpretados de formas diferentes por cada membro da equipe, gerando divergências no entendimento do que deve ser implementado (Vazques e Simões, 2016). Um requisito incompleto, por sua vez, deixa lacunas que serão preenchidas pelos desenvolvedores com base em suas próprias suposições.

Pressman (2021) explica que requisitos incompletos — aqueles que não descrevem plenamente as funcionalidades esperadas — frequentemente resultam em funcionalidades ausentes ou mal implementadas no produto final. Esses problemas geram retrabalho e custos desnecessários. Para combater a ambiguidade, os requisitos devem ser específicos, mensuráveis e testáveis. A clareza e a completude são, portanto, aspectos essenciais para o sucesso do projeto, exigindo um processo de análise iterativo para identificar e corrigir inconsistências e omissões.

Requisitos Voláteis: A mudança é uma constante em projetos de software, tornando os requisitos "voláteis". Essa volatilidade está relacionada à evolução das necessidades do negócio, dos objetivos dos stakeholders ou do ambiente organizacional. Segundo Sommerville (2020), essa é uma realidade inevitável em muitos projetos, especialmente em

contextos de alta competitividade. As mudanças podem ser impulsionadas por um melhor entendimento do sistema pelo cliente, novas regulamentações ou avanços tecnológicos.

O principal desafio não é evitar as mudanças, mas sim gerenciá-las de forma controlada. Reinehr (2020) aponta que um processo de gerenciamento de mudanças bem estruturado pode mitigar os efeitos negativos da volatilidade. Esse processo envolve a formalização de solicitações, a análise de impacto no escopo, custo e cronograma, e a manutenção da rastreabilidade. Metodologias ágeis são particularmente adequadas para lidar com a volatilidade, pois seus ciclos curtos e iterativos permitem a adaptação contínua às novas demandas.

Alto Custo de Erros: Um dos princípios mais conhecidos na Engenharia de Software é que o custo para corrigir um erro aumenta exponencialmente à medida que o projeto avança. De acordo com Pressman (2021), quanto mais tarde um erro na fase de requisitos é descoberto, maior será o seu custo para ser corrigido, podendo comprometer significativamente o orçamento e o cronograma do projeto. Uma falha no requisito inicial pode levar ao desenvolvimento de uma funcionalidade inteiramente errada, propagando o retrabalho por todas as etapas subsequentes: design, codificação, testes e implantação.

O alto custo se deve ao efeito cascata que um erro inicial provoca. Sommerville (2020) defende que revisões frequentes, validação contínua e prototipagem são estratégias eficazes para detectar falhas ainda nas fases iniciais. Portanto, investir tempo e recursos na validação e verificação de requisitos no início do ciclo de vida é uma das práticas mais eficazes para a economia do projeto, aumentando a qualidade do produto final e a satisfação do cliente.

Identificação de Stakeholders: A identificação de todos os stakeholders relevantes é um passo inicial crítico, mas muitas vezes desafiador, especialmente em sistemas complexos. Reinehr (2020) alerta que a exclusão de um stakeholder importante pode resultar em lacunas nos requisitos ou na definição de prioridades que não refletem o interesse de todos os envolvidos. Stakeholders são todos os indivíduos ou grupos que podem afetar ou ser afetados pelo projeto, e suas expectativas podem ser conflitantes.

A falha em mapear esse universo de influências pode levar a requisitos não descobertos e falta de apoio em momentos cruciais. Segundo Pressman (2021), é responsabilidade do analista de requisitos mapear todas as partes interessadas, compreender suas expectativas e garantir que essas perspectivas sejam consideradas desde o início. A

criação de um plano de comunicação e gerenciamento de stakeholders ajuda a garantir que as pessoas certas sejam consultadas nos momentos certos, minimizando riscos e aumentando as chances de sucesso do projeto.

Levantamento de Requisitos: O levantamento de requisitos, também conhecido como elicitación, é a fase inicial e uma das mais críticas da Engenharia de requisitos. Trata-se do processo de pesquisa, descoberta e documentação das necessidades dos usuários e stakeholders para um sistema. Segundo Pressman (2016), o levantamento de requisitos é uma atividade essencial, pois representa a base para a definição do escopo do projeto e o estabelecimento de um entendimento comum entre os envolvidos sobre o que deve ser desenvolvido. Seu objetivo não é apenas coletar uma lista de desejos, mas sim compreender profundamente os problemas de negócio, os objetivos do projeto e o contexto em que o software será utilizado.

Para realizar a investigação, se utiliza de diversas técnicas, em que podem ser aplicadas de forma isolada ou combinada. A escolha da abordagem correta depende do contexto do negócio, da disponibilidade dos stakeholders e da complexidade do sistema. A Figura 3 apresenta as técnicas de Elicitación de Requisitos seguindo a classificação apresentada por Glinz et al. (2024). Dentre as técnicas mais conhecidas e aplicadas, destacam-se as entrevistas, os questionários e a observação, estas serão mais detalhadas a seguir.

Figura 3 – Técnicas de Elicitação de Requisitos.



Fonte: Glinz (2024).

Entrevista: Considerada a técnica mais tradicional e interativa, a entrevista consiste em uma conversa direta com os stakeholders para explorar suas necessidades. Elas podem ser estruturadas, com um roteiro fixo de perguntas, ou semiestruturadas, permitindo maior flexibilidade para aprofundar tópicos relevantes que surgem durante o diálogo. Segundo Vazques e Simões (2016), as entrevistas são extremamente eficazes para obter informações qualitativas, capturar detalhes e nuances, e construir um bom relacionamento com o cliente, o que facilita a colaboração ao longo do projeto.

Questionários: Esta técnica é ideal para coletar informações de um grande número de pessoas de forma padronizada, especialmente quando os stakeholders estão geograficamente dispersos. Pressman (2021) aponta que os questionários são úteis para obter dados quantitativos e identificar tendências gerais, permitindo que um grande volume de respostas seja coletado de forma eficiente. No entanto, eles oferecem pouca flexibilidade para explorar respostas inesperadas e não permitem o esclarecimento de dúvidas em tempo real, sendo mais adequados para validar hipóteses do que para a descoberta inicial de requisitos complexos.

Observação: A observação consiste em o analista se inserir no ambiente de trabalho do usuário para observar seus processos, ferramentas e interações em primeira mão. Para Sommerville (2020), essa técnica é poderosa para descobrir requisitos implícitos — aquelas necessidades tão enraizadas na rotina do usuário que ele mesmo não consegue verbalizá-las por considerá-las óbvias. Ao observar o "mundo real", o analista pode identificar ineficiências e oportunidades de melhoria que não seriam mencionadas em uma entrevista, garantindo que o sistema proposto se adeque ao fluxo de trabalho real.

2.4. Modelos de Linguagem de Grande Escala LLMs

Os Modelos de Linguagem de Grande Escala LLMs representam um avanço significativo no campo da Inteligência Artificial (IA) e do Processamento de Linguagem Natural (PLN). São modelos de aprendizado profundo, treinados em vastos volumes de dados textuais, que demonstram uma notável capacidade de compreender, gerar e manipular a linguagem humana de forma coerente e contextualmente relevante. A escala desses modelos, que podem conter centenas de bilhões de parâmetros, permite que eles capturem padrões, sintaxe e semântica da linguagem com uma profundidade sem precedentes transformando a maneira como interagimos com a tecnologia (Brown et al., 2020).

Através de um processo de treinamento auto-supervisionado em corpus massivos de texto — como livros, artigos e o conteúdo da internet — os LLMs aprendem a prever a próxima palavra em uma sentença. Embora pareça uma tarefa simples, é essa capacidade preditiva, operando em uma escala massiva, que resulta em habilidades emergentes impressionantes (Brown et al., 2020). Essas habilidades incluem, mas não se limitam a tradução automática, resumo de documentos, resposta a perguntas, geração de conteúdo criativo e até mesmo a escrita de código de programação, tornando-os ferramentas flexíveis e de propósito geral.

2.4.1 Definição e Funcionamento

Um LLM é um tipo de rede neural artificial, geralmente baseada na arquitetura *Transformer*, que é capaz de aprender padrões complexos em dados de texto (Vaswani et al., 2017). Essa arquitetura permite que o modelo processe sequências de palavras de forma eficiente, capturando relações de longo alcance e dependências contextuais. A fase principal de sua criação é o pré-treinamento, na qual o modelo é exposto a uma quantidade massiva de dados não rotulados (textos da internet, livros, artigos), aprendendo a prever a próxima

palavra em uma sequência. Ao ser treinado em bilhões de palavras, o LLM adquire um conhecimento enciclopédico e a capacidade de gerar texto coerente e contextualmente relevante.

Posteriormente, para otimizar ou especializar o modelo para tarefas específicas, podem ser aplicadas técnicas como a engenharia de prompts (a arte de construir entradas de texto que guiam o modelo para a resposta desejada) ou o ajuste fino (*fine-tuning*). O *fine-tuning* é um processo no qual o modelo pré-treinado passa por um treinamento adicional em um conjunto de dados menor e mais específico, adaptando suas capacidades para um domínio particular e melhorando seu desempenho em tarefas especializadas.

2.4.2 Evolução e Capacidade

A evolução dos LLMs tem sido exponencial, impulsionada por avanços em poder computacional, disponibilidade de dados e otimizações na arquitetura *Transformer*. Modelos como GPT (*Generative Pre-trained Transformer*) da OpenAI (Radford et al., 2018), BERT (*Bidirectional Encoder Representations from Transformers*) do Google (Devlin et al., 2019) e LLaMA (*Large Language Model Meta AI*) da Meta AI (Touvron et al., 2023) são exemplos proeminentes dessa evolução, cada um introduzindo inovações em treinamento e arquitetura.

As capacidades atuais dos LLMs, frequentemente descritas como habilidades emergentes de modelos em grande escala, são diversas e impressionantes (Brown et al., 2020). Incluem:

Geração de Texto: Produção de artigos, resumos, e-mails, roteiros e até mesmo código de programação.

Tradução: Tradução de textos entre diferentes idiomas com alta fluidez e precisão.

Sumarização: Geração de resumos concisos de documentos longos, mantendo as informações essenciais.

Resposta a Perguntas: Compreensão de perguntas em linguagem natural e fornecimento de respostas informativas e relevantes.

Análise de Sentimento: Identificação e classificação do tom emocional de um texto (positivo, negativo, neutro).

Classificação de Texto: Categorização de documentos ou frases com base em seu conteúdo.

2.4.3 Aplicação em Engenharia de Software

O potencial dos LLMs se estende a diversas áreas da Engenharia de Software, transformando a forma como os desenvolvedores interagem com o código e os processos. Algumas aplicações notáveis incluem:

Geração de Código: LLMs podem gerar trechos de código, funções ou até mesmo classes inteiras a partir de descrições em linguagem natural, acelerando o desenvolvimento.

Refatoração de Código: Sugestão de melhorias no código existente para otimizar desempenho, legibilidade e manutenibilidade.

Geração de Testes: Criação de casos de teste e scripts de teste automatizados com base em especificações ou código-fonte.

Documentação de Código: Geração automática de documentação para funções, classes e módulos, facilitando a compreensão e manutenção do software.

Análise de Logs e Erros: Auxílio na identificação de padrões em logs de sistema e na sugestão de soluções para erros e exceções.

Essas aplicações demonstram o crescente impacto dos LLMs na produtividade e na qualidade do desenvolvimento de software, abrindo caminho para novas abordagens e ferramentas.

3. TRABALHOS RELACIONADOS

Este capítulo apresenta uma revisão da literatura sobre trabalhos relacionados ao uso de Modelos de Linguagem de Grande Escala (LLMs) na Engenharia de Requisitos (ER), com foco na elicitac  o de requisitos. A an  lise dos estudos selecionados visa identificar as tend  ncias atuais, os desafios e as oportunidades na aplica  o de LLMs neste campo, fornecendo uma base para o desenvolvimento do presente trabalho. O cap  tulo    organizado em se   es, onde cada se   o aborda um trabalho diferente.

3.1 How Mature is Requirements Engineering for AI-based Systems? A systematic Mapping Study on Practices, Challenges, and Future Research Directions

Habiba et al. (2024), realizaram um estudo de mapeamento sistem  tico sobre Engenharia de Requisitos para Sistemas Baseados em Intelig  ncia Artificial (RE4AI). O objetivo foi fornecer uma vis  o abrangente da RE4AI para pesquisadores e profissionais, identificando as pr  ticas dispon  veis, lacunas e desafios de pesquisa que ainda precisam ser abordados. A pesquisa foi motivada pelos novos desafios na engenharia de requisitos para IA, como a dificuldade em especificar e validar requisitos para IA ou considerar novos requisitos de qualidade devido a implica   es   ticas emergentes.

Para isso, os autores conduziram um estudo de mapeamento sistem  tico combinando busca por sequ  ncia de consulta e snowballing extensivo. O estudo incluiu a classifica  o da literatura existente e a caracteriza  o de pr  ticas em ferramentas, t  cnicas, modelos e processos. O objetivo foi oferecer uma vis  o sobre o estado atual da Engenharia de Requisitos, identificar lacunas e propor fundamentos para avan  os nesta   rea de estudo.

Os resultados do estudo destacaram tend  ncias significativas na Engenharia de Requisitos para Intelig  ncia Artificial, evidenciando a ado  o de pr  ticas tradicionais, como elicitac  o e valida  o de requisitos, e a incorpora  o de abordagens inovadoras, como m  todos colaborativos e an  lise de dados em tempo real. No entanto, desafios como a inconsist  ncia terminol  gica, a complexidade dos sistemas de IA e quest  es   ticas e de explicabilidade foram identificados.

O estudo de Habiba et al. (2024) concluiu que a Engenharia de Requisitos est   em um processo de crescente evolu  o, com diversifica  o de pr  ticas e abordagens. Embora tenha havido avan  os significativos, o estudo apontou a necessidade de adapta  o das pr  ticas tradicionais para enfrentar os desafios da Intelig  ncia Artificial, com   nfase em abordagens

centradas no ser humano e na integração das necessidades dos stakeholders. Além disso, foi sugerido o desenvolvimento de novas metodologias para melhorar a explicabilidade e transparência dos sistemas de IA abordando requisitos não funcionais como justiça e acessibilidade, e explorando conceitos emergentes como a modelagem de hipóteses.

3.2 Uso de LLMs em atividades de Engenharia de Requisitos

Equiton (2025), realizou um estudo empírico com o objetivo de avaliar a eficiência e a aplicabilidade dos Modelos de Linguagem de Grande Escala (LLMs) em atividades de Engenharia de Requisitos (ER). A pesquisa destaca que a ER é uma área fundamental da Engenharia de Software, e que estudos recentes têm apontado os LLMs como ferramentas promissoras e inovadoras neste campo. No entanto, o autor ressalta a necessidade de validar a eficiência e a aplicabilidade dos LLMs ao longo das diferentes etapas da Engenharia de Requisitos, bem como examinar se essas ferramentas podem ser reconhecidas como eficazes e confiáveis no contexto da ER.

A metodologia adotada por Equiton (2025) incluiu um estudo bibliográfico e a realização de um curso ministrado para estudantes universitários, o que permitiu a coleta de dados relevantes para a análise dos resultados obtidos. Os resultados evidenciam o potencial dos LLMs na Engenharia de Requisitos, bem como a eficácia das técnicas analisadas. Contudo, também foram identificadas limitações, principalmente relacionadas à infraestrutura dessas ferramentas, evidenciando a necessidade de pesquisas adicionais que explorem soluções mais abrangentes e avançadas.

Em conclusão, o trabalho de Equiton (2025) reforça o potencial dos LLMs para auxiliar nas atividades de Engenharia de Requisitos, mas também aponta para a importância de superar as limitações de infraestrutura e de realizar mais pesquisas para otimizar a aplicação dessas ferramentas no campo da ER.

3.3 Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs

Arora et al. (2024), realizaram uma investigação sobre o potencial dos Modelos de Linguagem de Grande Escala (LLMs) para impulsionar os processos de Engenharia de Requisitos (ER), com o objetivo de melhorar a eficiência e precisão das tarefas relacionadas aos requisitos. O estudo aborda a Engenharia de Requisitos como uma fase crítica no

desenvolvimento de software que inclui a elicitaco, anlise, especificaco e validaco de requisitos de software.

A pesquisa destaca que, apesar da importncia da ER, ela permanece um processo desafiador devido s complexidades de comunicaco, incerteza nos estgios iniciais e suporte inadequado de automaco. Os autores propem direces-chave e anlise SWOT para pesquisa e desenvolvimento no uso de LLMs para ER, focando no potencial para elicitaco, anlise, especificaco e validaco de requisitos.

Os resultados do estudo indicam que os LLMs tm mostrado promessa significativa em diversos domnios, incluindo processamento de linguagem natural, gerao de cdigo e compreenso de programas. O trabalho explora especificamente o potencial dos LLMs em impulsionar os processos de ER, visando melhorar a eficincia e preciso das tarefas relacionadas aos requisitos. Os autores apresentaram resultados de uma avaliao preliminar neste contexto, demonstrando o potencial dos LLMs para aprimorar vrias tarefas de ER.

O estudo de Arora et al. (2024) concluiu que os LLMs possuem potencial para melhorar significativamente os processos de Engenharia de Requisitos, especialmente na automaco de tarefas tradicionalmente manuais e na melhoria da comunicaco entre stakeholders. No entanto, os autores tambm identificaram desafios relacionados  preciso, confiabilidade e necessidade de validaco humana dos resultados gerados pelos LLMs.

3.4 Enhancing Requirements Engineering Practices Using Large Language Models. University of Gothenburg

Ronanki (2024), investigou o uso de Modelos de Linguagem de Grande Escala (LLMs) na Engenharia de Requisitos (ER), abordando suas capacidades tcnicas e as restries ticas e regulatrias. A pesquisa destaca que, embora os LLMs ofeream interao em linguagem natural, insights tcnicos e automaco de tarefas, sua integrao sistemtica nos processos de ER apresenta desafios e limitaes nicos. O estudo enfatiza a necessidade de desenvolver e compreender mecanismos para integrar LLMs de forma eficiente e responsvel.

Para alcanar seus objetivos, Ronanki (2024) empregou uma abordagem multimtodo, combinando quase-experimentos, entrevistas, uma pesquisa e um estudo de caso para coletar dados empricos sobre as habilidades tcnicas dos LLMs e as experincias dos usurios em tarefas e processos de ER. Alm disso, foi realizada uma reviso terciria seguida de uma meta-anlise da literatura sobre diretrizes e estruturas ticas de IA para identificar e

compreender as restrições envolvidas no uso de LLMs para tarefas e processos de ER na prática.

Os resultados dos experimentos empíricos revelaram que os LLMs são capazes de realizar tarefas técnicas como geração de requisitos, avaliação da qualidade de histórias de usuário, classificação binária de requisitos e rastreamento de requisitos interdependentes, com diferentes níveis de desempenho. A análise comparativa das diretrizes e estruturas éticas de IA revelou as restrições e requisitos envolvidos no uso de LLMs na prática. O estudo de caso industrial e a pesquisa resultaram em dez recomendações para a integração responsável e sistemática de LLMs para ER na prática.

Em conclusão, o estudo de Ronanki (2024) observou a necessidade de uma estrutura de delegação de tarefas humano-LLM, destacou a importância de validar LLMs em ambientes semelhantes ao mundo real e trouxe à tona os aspectos humanos e organizacionais subestimados. O trabalho futuro precisa aprofundar a identificação e mitigação dos desafios associados à adoção e integração de LLMs, incluindo a necessidade de estudar as barreiras organizacionais, os fatores decisivos e os artefatos que influenciam e permitem a adoção responsável e sistemática de LLMs.

3.5. Discussões

A análise da literatura, abrangendo os estudos de Habiba et al. (2024), Equiton (2025), Arora et al. (2024) e Ronanki (2024), revela um consenso claro: a aplicação de LLMs na Engenharia de Requisitos é uma área emergente e de alto potencial, mas que ainda carece de maturidade e de validação empírica estruturada.

Os trabalhos convergem ao apontar que os LLMs se destacam na automação de tarefas manuais e na melhoria da comunicação, como a geração de artefatos e o processamento de linguagem natural. No entanto, todos os autores ressaltam, com igual ênfase, os desafios relacionados à precisão, confiabilidade e à indispensável necessidade de validação humana. A pesquisa de Ronanki (2024), em particular, destaca a importância de desenvolver mecanismos de delegação e integração responsável, um ponto que ecoa nos demais estudos.

A principal lacuna identificada é a falta de diretrizes práticas e detalhadas que orientem os engenheiros de requisitos sobre *como* interagir com os LLMs de forma eficaz para cada atividade específica da elicitación. Embora os estudos validem o potencial, eles não oferecem um guia de uso estruturado com exemplos de *prompts* e análises de resultados por tarefa.

4. METODOLOGIA

A pesquisa proposta será estruturada em quatro etapas interligadas, cada uma com objetivos específicos que contribuem para a validação do uso de Modelos de Linguagem de Grande Escala (LLMs) na Engenharia de Requisitos (ER). A abordagem metodológica visa garantir a robustez e a relevância dos resultados, combinando revisão bibliográfica, elaboração de diretrizes, experimentação prática e análise qualitativa. As etapas são detalhadas a seguir:

4.1. Revisão Bibliográfica

A etapa inicial desta pesquisa consistirá em uma revisão da literatura para investigar a aplicação de LLMs na Engenharia de Requisitos ER. O objetivo refere-se a identificar e analisar trabalhos relevantes que abordam o uso de LLMs na área, a fim de compreender as melhores práticas, oportunidades e limitações dessa integração.

Para tanto, serão consultadas bases de dados acadêmicas de relevância na área de Engenharia de Software e Inteligência Artificial. As buscas serão realizadas utilizando combinações de palavras-chave como "Large Language Models", "LLMs", "Requirements Engineering", "Elicitation", "Artificial Intelligence", "Software Engineering", entre outras, em português e inglês, para garantir uma cobertura ampla da literatura.

Os artigos encontrados serão analisados com base em sua relevância para o tema, contribuições específicas e aplicabilidade prática. A partir dessa análise, será possível identificar as abordagens mais comuns, as ferramentas emergentes e os principais desafios na utilização de LLMs para apoiar as atividades de ER.

Esta revisão disponibiliza a base teórica para a pesquisa, permitindo identificar lacunas e tendências que orientarão as etapas seguintes do trabalho. Além disso, o conhecimento adquirido será fundamental para a elaboração das diretrizes e para a análise dos resultados da experimentação prática, garantindo que o estudo esteja alinhado ao cenário científico atual da área.

4.2. Elaboração de Diretrizes para Uso de LLMs

Com base nos achados da revisão bibliográfica, esta etapa se dedicará à elaboração de um conjunto de diretrizes e boas práticas para o uso eficaz de LLMs em atividades de Engenharia de Requisitos, com foco especial na etapa de elicitação. O conhecimento

adquirido sobre as abordagens existentes, os desafios e as oportunidades identificadas na literatura será fundamental para a construção dessas diretrizes.

As diretrizes serão formuladas de maneira clara e prática, visando auxiliar engenheiros de requisitos e desenvolvedores no aproveitamento do potencial dos LLMs. Isso incluirá a criação de *prompts* estruturados, que são instruções específicas e otimizadas para interagir com os LLMs e obter resultados relevantes para a elicitación de requisitos. Serão abordados aspectos como a formulação de perguntas, a especificação do contexto, a definição do formato de saída desejado e a iteração com o modelo para refinar os resultados.

4.3 Experimentação Prática

A experimentação prática constitui uma etapa crucial desta pesquisa, visando avaliar empiricamente a aplicação e os resultados das diretrizes propostas para o uso de LLMs na elicitación de requisitos. Esta fase é conduzida por meio de um estudo de caso, no qual as diretrizes e os *prompts* desenvolvidos serão aplicados em um cenário prático de Engenharia de Requisitos.

O estudo de caso envolverá um projeto de desenvolvimento de software, em que um conjunto de requisitos será elicitado com o auxílio de um LLM, seguindo o método proposto neste trabalho. O foco refere-se a colocar em prática as diretrizes e observar a eficácia do modelo para gerar, refinar e documentar requisitos a partir de descrições iniciais do problema.

O objetivo principal desta experimentação é coletar dados que permitam analisar a qualidade e a utilidade dos artefatos gerados com o apoio do LLM. A avaliação dos resultados esteve focada na qualidade, clareza, consistência e abrangência dos requisitos elicitados, conforme os critérios detalhados na próxima seção. Esta etapa consiste em evidências empíricas sobre o valor agregado e as limitações da abordagem proposta em um contexto prático, validando a aplicabilidade das diretrizes desenvolvidas.

4.4 Análise dos Resultados

A etapa final da metodologia consiste na análise aprofundada dos resultados obtidos na experimentação prática. O principal método de análise será a comparação qualitativa dos requisitos produzidos pelas duas abordagens de elicitación: aquela que utilizou LLMs e a que empregou métodos tradicionais. Esta comparação será realizada com base em um conjunto de critérios de qualidade de requisitos, visando uma avaliação sistemática e objetiva.

Os critérios de avaliação incluirão, clareza, consistência, ambiguidade, abrangência e rastreabilidade. A análise será conduzida por especialistas, se possível, para minimizar vieses, e os resultados serão triangulados com as observações e feedbacks coletados durante a experimentação prática.

Além da comparação qualitativa, serão analisados os dados coletados sobre o processo de elicitación, a percepção dos participantes sobre a facilidade de uso e a utilidade dos LLMs, e os desafios específicos enfrentados. Esta análise permitirá não apenas validar a hipótese de que os LLMs podem auxiliar na elicitación de requisitos, mas também identificar as áreas onde seu impacto é mais significativo e onde melhorias são necessárias. Os resultados desta análise fornecerão insights valiosos para as conclusões do trabalho e para futuras pesquisas na área.

5 DIRETRIZES PARA O USO DE LLMs NO LEVANTAMENTO DE REQUISITOS

Este capítulo apresenta diretrizes e boas práticas para a utilização de LLMs no auxílio às atividades de Engenharia de Requisitos, com foco especial na etapa de elicitação. As diretrizes aqui propostas são baseadas em uma revisão da literatura sobre a aplicação de LLMs na Engenharia de Requisitos, conforme detalhado no Capítulo de Trabalhos Relacionados, e enriquecidas por experimentos empíricos e explorações práticas realizadas com essas tecnologias. O objetivo é fornecer um arcabouço estruturado para otimizar a interação com LLMs, transformando-os em assistentes eficazes para engenheiros de requisitos.

O guia está organizado em seções, cada uma dedicada a uma tarefa específica no processo de elicitação de requisitos que pode ser auxiliada com o uso de LLMs. Para cada tarefa, será apresentada uma descrição detalhada, seus objetivos, os resultados esperados e um *prompt* genérico que pode ser adaptado para diferentes contextos. Além disso, exemplos práticos serão fornecidos para ilustrar a aplicação dos *prompts* e os tipos de resultados que podem ser obtidos. Para os exemplos utilizamos o DeepSeek. A abordagem visa capacitar os profissionais a integrarem os LLMs de forma eficiente e responsável em suas práticas de Engenharia de Requisitos, maximizando os benefícios e mitigando os desafios inerentes a essas tecnologias.

5.1. Compreensão do Domínio do Negócio

A compreensão do domínio do negócio é uma etapa fundamental na elicitação de requisitos, pois estabelece a base para todo o processo de desenvolvimento de software. Envolve a imersão no ambiente do cliente, a identificação de seus processos atuais, dores, necessidades e objetivos estratégicos. A dificuldade reside em extrair informações completas e precisas de *stakeholders*, que podem não ter clareza sobre suas próprias necessidades ou que utilizam uma linguagem específica do negócio, diferente da linguagem técnica da equipe de desenvolvimento.

O principal objetivo desta tarefa é gerar perguntas essenciais que permitam ao engenheiro de requisitos compreender profundamente o contexto de negócio do cliente. Isso inclui:

- Identificar os processos de negócio existentes

- Detectar pontos de dor e ineficiências nos processos atuais.
- Compreender as necessidades e expectativas dos stakeholders.
- Estabelecer uma base de conhecimento sólida para a elicitação de requisitos detalhados.

Quadro 1 – Prompt genérico para compreensão do domínio de negócio

Aspecto	Descrição
Objetivo	Gerar perguntas essenciais para compreender o contexto de negócio.
Papel do LLM	Analista de Requisitos experiente.
Contexto	Projeto de desenvolvimento de software para [informar o domínio].
Tarefa	Gerar 10 perguntas essenciais para compreender o contexto de negócio, processo atuais, dores e necessidades.
Formato de Saída	Lista numerada de perguntas, agrupadas por categorias (contexto organizacional, processos atuais, problemas e necessidades).
Prompt completo	Aja como um Analista de requisitos experiente. Estou iniciando um projeto de desenvolvimento de software para [descrever brevemente o domínio]. Tarefa: com base no domínio informado, gere uma lista de 10 perguntas essenciais que eu deveria fazer aos stakeholders para compreender adequadamente o contexto de negócio, processos atuais, dores e necessidades. Formato de saída: Lista numerada de perguntas, agrupadas por categoria (ex: contexto organizacional, processos atuais, problemas e necessidades).

Fonte: Fonte própria (2025).

5.1.1. Resultados Esperados

Espera-se obter uma lista numerada de perguntas bem formuladas, agrupadas por categorias relevantes (ex: contexto organizacional, processos atuais, problemas e necessidades). Essas perguntas servirão como um roteiro para entrevistas e discussões com os

stakeholders, garantindo que nenhum aspecto crítico do domínio do negócio seja negligenciado.

5.1.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.

Aja como um Analista de requisitos experiente. Estou iniciando um projeto de desenvolvimento de software para um sistema de gerenciamento de estoque para uma pequena livraria. Tarefa: com base no domínio informado, gere uma lista de 10 perguntas essenciais que eu deveria fazer aos *stakeholders* para compreender adequadamente o contexto de negócio, processos atuais, dores e necessidades, sob a perspectiva de um engenheiro de requisitos. Formato de saída: Lista numerada de perguntas, agrupadas por categoria (ex: processos de negócio, dados, interfaces, desempenho, segurança).

5.1.3. Resposta do LLM

Processos de Negócio:

1. Quais são os passos exatos para registrar a entrada de um novo livro no estoque, desde o recebimento até a disponibilização para venda?
2. Descreva o fluxo de trabalho completo para a realização de um pedido de compra de livros aos fornecedores, incluindo aprovações e prazos.

Dados:

1. Quais informações essenciais precisam ser armazenadas para cada livro no estoque (ex: ISBN, título, autor, editora, preço de custo, preço de venda, quantidade, localização)?
2. Existem dados de clientes ou histórico de vendas que precisam ser integrados ou considerados pelo sistema de estoque?

Interfaces:

1. O novo sistema de estoque precisará se integrar com algum sistema existente (ex: sistema de vendas, sistema financeiro, plataforma de e-commerce)? Se sim, quais e como?
2. Quais tipos de usuários interagirão com o sistema e quais são suas principais necessidades de interface (ex: tela de vendas, tela de cadastro de produtos, relatórios)?

Desempenho e Segurança:

1. Qual o volume médio e máximo de transações de estoque esperadas por dia (entradas, saídas, consultas)?
2. Quais são os requisitos de tempo de resposta para as operações críticas do sistema (ex: consulta de estoque, registro de venda)?
3. Quais são as preocupações de segurança em relação aos dados do estoque e quem terá acesso a quais informações?
4. Existem requisitos específicos de auditoria ou conformidade regulatória para o registro de movimentações de estoque?

5.2. Identificação de *Stakeholders*

A identificação de stakeholders é um passo crítico no início de qualquer projeto de software. Sommerville (2011), destaca que *stakeholders* são indivíduos, grupos ou organizações afetados pelo projeto ou que podem influenciar seu resultado. A falha em identificar *stakeholders* chaves ou em compreender suas expectativas e necessidades pode levar a requisitos incompletos, conflitos e, em última instância, ao fracasso do projeto. O desafio é mapear todas as partes interessadas, suas responsabilidades, seu nível de influência e as informações que podem fornecer para o processo de elicitação de requisitos.

O objetivo desta tarefa é identificar os *stakeholders* relevantes para o processo de elicitação de requisitos, garantindo que todas as perspectivas importantes sejam consideradas incluindo:

- Listar os tipos de stakeholders potenciais
- Explicar a importância de cada tipo de stakeholder para o projeto
- Detalhar as informações específicas que cada pode fornecer
- Sugerir as técnicas de elicitação mais adequadas para cada tipo de stakeholder.

Quadro 2 – Prompt genérico para identificação de Stakeholder

Aspecto	Descrição
Objetivo	Identificar <i>stakeholders</i> relevantes para o processo de elicitação.
Papel do LLM	Gerente de projetos com experiência em Engenharia de requisitos.
Contexto	Projeto para desenvolver [tipo do sistema] para [empresa].
Tarefa	Identificar <i>stakeholders</i> potenciais, explicando sua importância, informações que podem fornecer e técnicas de elicitação adequada.
Formato de Saída	Lista de <i>stakeholders</i> com as informações solicitadas.
Prompt completo	Aja como um Gerente de Projetos com experiência em Engenharia de Requisitos. Contexto: Estou iniciando um projeto para desenvolver [tipo de sistema] para [setor]. Tarefa: Identifique os potenciais stakeholders que deveriam ser consultados durante a fase de elicitação de requisitos. Para cada tipo de stakeholder, explique brevemente: 1. Por que este stakeholder é importante para o projeto 2. Quais informações específicas este stakeholder pode fornecer 3. Quais técnicas de elicitação seriam mais adequadas para este stakeholder. Formato de saída: Lista de stakeholders com as informações solicitadas.

Fonte: Fonte própria (2025).

5.2.1. Resultados Esperados

Espera-se obter uma lista organizada de *stakeholders*, com uma breve explicação de sua relevância, o tipo de informação que podem oferecer e as técnicas de elicitação mais eficazes para interagir com eles. Essa lista servirá como um guia para o planejamento das atividades de elicitação, assegurando que todas as vozes importantes sejam ouvidas.

5.2.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria.

Aja como um Gerente de Projetos com experiência em Engenharia de Requisitos. Contexto: Estou iniciando um projeto para desenvolver um sistema de gerenciamento de

estoque para uma livraria. Tarefa: Identifique os potenciais *stakeholders* que deveriam ser consultados durante a fase de elicitação de requisitos. Para cada tipo de *stakeholder*, explique brevemente: 1. Por que este *stakeholder* é importante para o projeto 2. Quais informações específicas este *stakeholder* pode fornecer 3. Quais técnicas de elicitação seriam mais adequadas para este *stakeholder*. Formato de saída: Lista de *stakeholders* com as informações solicitadas.

5.2.3. Resposta do LLM

1. Proprietário/Gerente da Livraria:

- **Importância:** Detém a visão estratégica do negócio, define objetivos de alto nível para o estoque e aprova investimentos.
- **Informações:** Metas de negócio (ex: redução de perdas, otimização de compras), visão de longo prazo para o gerenciamento de estoque, restrições orçamentárias e de tempo, expectativas de retorno sobre o investimento.
- **Técnicas de Elicitação:** Entrevistas de alto nível, *workshops* de visão de produto.

2. Funcionários do Estoque/Vendedores:

- **Importância:** São os usuários diretos do sistema, lidam com o dia a dia do estoque (recebimento, organização, vendas) e conhecem os processos operacionais.
- **Informações:** Fluxo de trabalho atual de entrada e saída de livros, problemas diários com o controle manual, necessidades de funcionalidades para registro de vendas, devoluções, inventário e localização de produtos.
- **Técnicas de Elicitação:** Entrevistas, observação de processos, questionários, *workshops* de brainstorming.

3. Contador/Responsável Financeiro:

- **Importância:** Garante a conformidade fiscal e contábil, e precisa de informações precisas sobre o valor do estoque e movimentações financeiras.

- **Informações:** Requisitos para relatórios financeiros (custo de mercadoria vendida, valor do estoque), integração com sistemas contábeis, regras fiscais aplicáveis à venda de livros.
- **Técnicas de Elicitação:** Entrevistas, análise de documentos financeiros e fiscais.

4. **Fornecedores de Livros/Editoras:**

- **Importância:** São parceiros externos que interagem com o processo de compra e recebimento de mercadorias.
- **Informações:** Formatos de dados para pedidos de compra, prazos de entrega, condições de devolução, informações sobre novos lançamentos e catálogos.
- **Técnicas de Elicitação:** Entrevistas, análise de contratos e acordos comerciais.

5. **Clientes da Livraria (potenciais):**

- **Importância:** Embora não usem o sistema de estoque diretamente, suas necessidades e a experiência de compra são impactadas pela eficiência do estoque.
- **Informações:** Expectativas sobre disponibilidade de livros, agilidade na localização de produtos, interesse em funcionalidades como consulta de estoque online ou reserva de livros.
- **Técnicas de Elicitação:** Pesquisas de satisfação, grupos focais (para entender a experiência de compra).

5.3. **Análise de transcrições de entrevista**

As entrevistas com *stakeholders* geram um volume significativo de dados em linguagem natural, muitas vezes não estruturados e contendo informações explícitas e implícitas. A análise dessas transcrições para extrair requisitos funcionais de maneira precisa e completa é um desafio. A identificação de requisitos implícitos, que não são diretamente declarados, mas inferidos do contexto da conversa, é particularmente complexa e exige um alto grau de interpretação e conhecimento do domínio.

O objetivo desta tarefa é auxiliar o engenheiro de requisitos na extração de requisitos funcionais a partir de transcrições de entrevistas. Isso inclui:

- Identificar e descrever requisitos funcionais de forma clara e objetiva.
- Distinguir entre requisitos explícitos e implícitos.
- Fornecer evidências textuais da transcrição para cada requisito identificado.

Quadro 3 – Prompt genérico para requisitos funcionais da entrevista.

Aspecto	Descrição
Objetivo	Extrair requisitos funcionais de transcrição de entrevista.
Papel do LLM	Analista de Requisitos experiente.
Contexto	Transcrição de entrevista com [cargo] sobre o sistema [nome do sistema].
Tarefa	Analisar transcrição e extrair requisitos funcionais, descrevendo cada um, indicando se é explícito ou implícito, e citando evidência.
Formato de Saída	Lista numerada de requisitos funcionais com as informações solicitadas.
Prompt completo	Aja como um Analista de Requisitos experiente. Contexto: A transcrição em anexo é de uma entrevista com [cargo do stakeholder] sobre o sistema [nome do sistema]. Tarefa: Analise a transcrição e extraia todos os requisitos funcionais mencionados explícita ou implicitamente. Para cada requisito: 1. Descreva-o em uma frase clara e objetiva 2. Indique se foi mencionado explicitamente ou está implícito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas). Transcrição: "" [Transcrição da entrevista] "". Formato de saída: Lista numerada de requisitos funcionais com as informações solicitadas.

Fonte: Fonte própria (2025).

5.3.1. Resultados esperados

Espera-se obter uma lista numerada de requisitos funcionais, cada um com sua descrição, indicação de ser explícito ou implícito, e a citação da parte relevante da transcrição que a evidencia. Essa organização facilita a validação dos requisitos e a rastreabilidade com a fonte original.

5.3.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria

5.3.2.1. Exemplo da entrevista

"Olha, o que a gente precisa mesmo é conseguir ter um controle exato do que entra e sai do estoque. Hoje, a gente faz tudo no papel e é uma bagunça. Seria ótimo se o sistema nos avisasse quando um livro está acabando, tipo, quando chega a 5 unidades, para a gente poder pedir mais. E claro, precisamos que ele registre todas as movimentações, sejam vendas, devoluções ou perdas. Isso é fundamental para o inventário. E, se possível, que a gente consiga gerar relatórios de vendas por gênero literário, isso nos ajudaria muito no planejamento de compras."

5.3.2.2. Exemplo do prompt

Aja como um Analista de Requisitos experiente. Contexto: A transcrição em anexo é de uma entrevista com o Gerente da Livraria sobre o Sistema de Gerenciamento de Estoque para Livraria. Tarefa: Analise a transcrição e extraia todos os requisitos funcionais mencionados explícita ou implicitamente. Para cada requisito: 1. Descreva-o em uma frase clara e objetiva 2. Indique se foi mencionado explicitamente ou está implícito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas). Transcrição: ""Olha, o que a gente precisa mesmo é conseguir ter um controle exato do que entra e sai do estoque. Hoje, a gente faz tudo no papel e é uma bagunça. Seria ótimo se o sistema nos avisasse quando um livro está acabando, tipo, quando chega a 5 unidades, para a gente poder pedir mais. E claro, precisamos que ele registre todas as movimentações, sejam vendas, devoluções ou perdas. Isso é fundamental para o inventário. E, se possível, que a gente consiga gerar relatórios de vendas por gênero literário, isso nos ajudaria muito no planejamento de compras."" Formato de saída: lista numerada de requisitos funcionais com as informações solicitadas.

5.3.3. Resposta do LLM

1. **Descrição:** O sistema deve permitir o controle exato das entradas e saídas do estoque.
 - **Tipo:** Explícito

- **Evidência:**

"...conseguir ter um controle exato do que entra e sai do estoque."

2. **Descrição:** O sistema deve notificar o usuário quando a quantidade de um livro no estoque atingir um limite mínimo (ex: 5 unidades).

- **Tipo:** Explícito

- **Evidência:**

"...seria ótimo se o sistema nos avisasse quando um livro está acabando, tipo, quando chega a 5 unidades..."

3. **Descrição:** O sistema deve registrar todas as movimentações de estoque, incluindo vendas, devoluções e perdas.

- **Tipo:** Explícito

- **Evidência:**

"...precisamos que ele registre todas as movimentações, sejam vendas, devoluções ou perdas."

4. **Descrição:** O sistema deve permitir a geração de relatórios de vendas por gênero literário.

- **Tipo:** Explícito

- **Evidência:**

"...que a gente consiga gerar relatórios de vendas por gênero literário..."

5.4. Identificação de requisitos não funcionais

Requisitos Não-Funcionais (RNFs) definem as qualidades e restrições de um sistema, como desempenho, segurança, usabilidade, confiabilidade e manutenibilidade. Diferentemente dos requisitos funcionais, que descrevem o que o sistema deve fazer, os RNFs especificam como o sistema deve se comportar. A identificação e elicitación de RNFs são frequentemente mais desafiadoras, pois são menos tangíveis e podem ser expressos de forma ambígua pelos *stakeholders*. Além disso, muitos RNFs são implícitos e requerem um engenheiro de requisitos experiente para serem descobertos e formalizados.

O objetivo dessa tarefa é auxiliar na identificação de requisitos não-funcionais a partir de transcrições de entrevistas ou outras fontes de informação.

- Detectar menções explícitas ou implícitas a atributos de qualidade do sistema.
- Classificar os RNFs em categorias padrão (ex: desempenho, segurança, usabilidade).
- Fornecer evidências textuais para cada RNF identificado.

Quadro 4 – Prompt genérico para identificação de requisitos Não-Funcionais

Aspecto	Descrição
Objetivo	Identificar requisitos não-funcionais em transcrição de entrevista.
Papel do LLM	Engenheiro de software especializado em requisitos não-funcionais.
Contexto	Transcrição de entrevista com [cargo] sobre o sistema [nome do sistema].
Tarefa	Analisar transcrição e identificar requisitos não-funcionais, categorizando-os segundo ISO/IEC 25010, descrevendo cada um, citando evidência e sugerindo métricas.
Formato de Saída	Requisitos não-funcionais agrupados por categoria, com as informações solicitadas.
Prompt completo	Aja como um Engenheiro de Software especializado em requisitos não-funcionais. Contexto: A transcrição abaixo é de uma entrevista com [cargo do stakeholder] sobre o sistema [nome/tipo do sistema]. Tarefa: Analise a transcrição e identifique todos os requisitos não-funcionais mencionados ou implícitos, categorizando-os segundo a ISO/IEC 25010 (usabilidade, desempenho, segurança, confiabilidade, etc.). Para cada requisito: 1. Descreva-o claramente 2. Indique a categoria do requisito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas) 4. Sugira uma métrica ou critério de aceitação para este requisito. Transcrição: "" [Transcrição da entrevista] "". Formato de saída: Requisitos não-funcionais agrupados por categoria, com as informações solicitadas.

Fonte: Fonte própria (2025).

5.4.1. Resultados esperados

Espera-se obter uma lista de requisitos não-funcionais, categorizados e com suas respectivas evidências. Essa lista é crucial para garantir que o sistema não apenas atenda às funcionalidades desejadas, mas também possua as qualidades necessárias para ser bem-sucedido em seu ambiente operacional.

5.4.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria

5.3.2.1. Exemplo da entrevista

"É fundamental que o sistema seja rápido ao consultar o estoque, os clientes não podem ficar esperando. Principalmente na hora da venda, o atraso tem que ser mínimo. E a segurança dos dados dos livros e dos clientes é nossa prioridade número um, não podemos ter perdas ou acessos indevidos. Além disso, o sistema precisa ser fácil de usar, mesmo para quem não tem muita familiaridade com sistemas de computador. E, claro, tem que estar disponível durante todo o horário de funcionamento da livraria, sem quedas. A gente também precisa conseguir fazer atualizações sem ter que parar as vendas por muito tempo."

5.3.2.2. Exemplo do *prompt*

Aja como um Engenheiro de software especializado em requisitos não-funcionais. Contexto: A transcrição em anexo é de uma entrevista com o Gerente da Livraria sobre o Sistema de Gerenciamento de Estoque para Livraria. Tarefa: Analise a transcrição e extraia todos os requisitos não-funcionais mencionados explícita ou implicitamente. Para cada requisito: 1. Descreva-o em uma frase clara e objetiva 2. Indique se foi mencionado explicitamente ou está implícito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas). Formato de saída: Lista numerada de requisitos não-funcionais com as informações solicitadas, agrupados por categoria (ex: Desempenho, Segurança, Usabilidade, Confiabilidade, Manutenibilidade).

5.4.3. Resposta do LLM

Desempenho

1. **Descrição:** O sistema deve ter um tempo de resposta rápido ao consultar o estoque.

- **Tipo:** Explícito
- **Evidência:**

"...é fundamental que o sistema seja rápido ao consultar o estoque, os clientes não podem ficar esperando."

2. **Descrição:** O atraso na hora da venda deve ser mínimo.

- **Tipo:** Explícito
- **Evidência:**

"...Principalmente na hora da venda, o atraso tem que ser mínimo."

Segurança

3. **Descrição:** A segurança dos dados dos livros e dos clientes deve ser a prioridade número um, evitando perdas ou acessos indevidos.

- **Tipo:** Explícito
- **Evidência:**

"...E a segurança dos dados dos livros e dos clientes é nossa prioridade número um, não podemos ter perdas ou acessos indevidos."

Usabilidade

4. **Descrição:** O sistema deve ser fácil de usar, mesmo para usuários com pouca familiaridade com sistemas de computador.

- **Tipo:** Explícito
- **Evidência:**

"...o sistema precisa ser fácil de usar, mesmo para quem não tem muita familiaridade com sistemas de computador."

Confiabilidade

5. **Descrição:** O sistema deve estar disponível durante todo o horário de funcionamento da livraria, sem quedas.

- **Tipo:** Explícito
- **Evidência:**

“...tem que estar disponível durante todo o horário de funcionamento da livraria, sem quedas.”

Manutenibilidade

6. Descrição: O sistema deve permitir atualizações sem a necessidade de parar as vendas por muito tempo.

- **Tipo:** Implícito
- **Evidência:**

“...A gente também precisa conseguir fazer atualizações sem ter que parar as vendas por muito tempo.”

5.5. Especificação de caso de uso

A especificação de caso de uso é uma técnica fundamental na Engenharia de Requisitos para descrever as interações entre os usuários (atores) e o sistema, detalhando as funcionalidades que o sistema deve oferecer. O desafio reside em criar especificações claras, completas e não ambíguas que capturem todos os cenários possíveis, incluindo fluxos alternativos e de exceção. A elaboração manual de casos de uso detalhados pode ser um processo demorado e propenso a erros, especialmente em sistemas complexos

O objetivo desta tarefa é auxiliar na criação de especificações detalhadas de casos de uso a partir de descrições de funcionalidades ou interações.

- Definir o nome do caso de uso e os atores envolvidos.
- Descrever as pré-condições e pós-condições.
- Detalhar o fluxo principal de eventos e os fluxos alternativos/de exceção.
- Identificar requisitos especiais associados ao caso de uso.

Quadro 5 – Prompt genérico para especificação de caso de uso

Aspecto	Descrição
Objetivo	Criar especificação detalhada de caso de uso.
Papel do LLM	Engenheiro de Requisitos especializado em casos de uso.
Contexto	Documentação de requisitos para o sistema [nome do sistema].
Tarefa	Criar especificação de caso de uso detalhada com nome, atores, pré-condições, fluxos, pós-condições e requisitos especiais.
Formato de Saída	Especificação completa do caso de uso no formato estruturado.
Prompt completo	Aja como um Engenheiro de Requisitos especializado em casos de uso. Contexto: Estou documentando requisitos para o sistema [nome do sistema] e preciso criar especificações de casos de uso. Informações sobre as funcionalidades: ""[colar informações sobre as funcionalidades específicas]"". Tarefa: Crie uma especificação de caso de uso detalhada incluindo: 1. Nome do caso de uso 2. Atores principais e secundários 3. Pré-condições 4. Fluxo básico (seu passo a passo) 5. Fluxos alternativos 6. Fluxos de exceção 7. Pós-condições 8. Requisitos especiais (Nesse caso, requisitos não funcionais relacionados) 9. Frequência de ocorrência estimada. Formato de saída: Especificação completa do caso de uso no formato estruturado.

Fonte: Fonte própria (2025).

5.5.1. Resultados esperados

Espera-se obter uma especificação de caso de uso completa e estruturada, que sirva como base para o *design*, desenvolvimento e teste do sistema. A especificação deve ser compreensível tanto para stakeholders técnicos quanto não técnicos.

5.5.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria

5.3.2.1. Exemplo do prompt

Aja como um Engenheiro de Requisitos especializado em casos de uso. Contexto: Estou documentando os requisitos para o Sistema de Gerenciamento de Estoque para Livraria. Tarefa: Crie uma especificação de caso de uso detalhada para a funcionalidade "Registrar Entrada de Livro no Estoque", incluindo: Nome do Caso de Uso, Atores, Pré-condições, Fluxo Principal de Eventos, Fluxos Alternativos, Fluxos de Exceção, Pós-condições e Requisitos Especiais. Formato de saída: Texto estruturado com os itens solicitados.

5.5.3. Resposta do LLM

Nome do Caso de Uso: Registrar Entrada de Livro no Estoque

Atores: Funcionário do Estoque

Pré-condições:

- O Funcionário do Estoque está logado no sistema.
- O Livro a ser registrado está fisicamente presente na livraria.

Fluxo Principal de Eventos:

1. O Funcionário do Estoque acessa a funcionalidade de registro de entrada de livros.
2. O Funcionário do Estoque informa o Número Padrão Internacional de Livro.
3. O sistema verifica se o Livro já está cadastrado. Se sim, exibe os dados existentes. Se não, solicita o preenchimento dos dados do novo Livro (título, autor, editora, preço de custo, preço de venda etc.).
4. O Funcionário do Estoque informa a quantidade de exemplares que estão entrando no estoque.
5. O Funcionário do Estoque informa a localização física (prateleira, seção) onde os livros serão armazenados.
6. O Funcionário do Estoque confirma a entrada.

7. O sistema registra a entrada, atualiza a quantidade em estoque do Livro e a data da última entrada.
8. O sistema exibe a confirmação do registro e o total de exemplares em estoque.

Fluxos Alternativos:

- **A.1. ISBN não encontrado (novo livro):** No passo 3, se o ISBN não for encontrado, o sistema solicita que o Funcionário do Estoque preencha os dados completos do novo livro (título, autor, editora, preço de custo, preço de venda, etc.) antes de prosseguir.
- **A.2. Dados de livro existentes precisam ser atualizados:** No passo 3, se o livro já estiver cadastrado, mas houver necessidade de atualizar informações (ex: novo preço de custo), o sistema permite a edição antes da confirmação.

Fluxos de Exceção:

- **E.1. Erro de Conexão com o Banco de Dados:** Se ocorrer um erro de conexão durante o registro da entrada, o sistema exibe uma mensagem de erro e tenta novamente ou informa o Funcionário para tentar mais tarde.
- **E.2. ISBN Inválido:** Se o ISBN informado não seguir o padrão esperado, o sistema exibe uma mensagem de erro e solicita um ISBN válido.

Pós-condições:

- A quantidade em estoque do Livro é atualizada.
- Um registro de entrada é criado no histórico de movimentações do estoque.
- A data da última entrada do Livro é atualizada.

Requisitos Especiais:

- O sistema deve validar o formato do ISBN (10 ou 13 dígitos).
- O sistema deve permitir a impressão de etiquetas com código de barras para os novos livros.
- O sistema deve gerar um relatório diário das entradas de livros.

5.6. Especificação de regras de negócio

As regras de negócio são declarações que definem ou restringem algum aspecto do negócio, controlando o comportamento de um sistema. Elas são cruciais para garantir que o software reflita com precisão as políticas, procedimentos e leis que governam a organização. O desafio na especificação de regras de negócio reside em identificá-las de forma completa, expressá-las de maneira clara e não ambígua, e garantir que sejam consistentes entre si e com os requisitos funcionais. Regras de negócio mal definidas podem levar a sistemas que não atendem às necessidades do negócio ou que operam de forma incorreta.

O objetivo desta tarefa é auxiliar na documentação detalhada das regras de negócio que governam o comportamento do sistema. Isso inclui:

- Identificar regras de negócio a partir de requisitos funcionais ou descrições de processos.
- Descrever cada regra de forma precisa, incluindo suas condições e ações.
- Atribuir um identificador único a cada regra.
- Fornece exemplos práticos para ilustrar a aplicação da regra.

Quadro 6– Prompt genérico para especificação de regras de negócio

Aspecto	Descrição
Objetivo	Documentar regras de negócio detalhadas para o sistema.
Papel do LLM	Analista de Requisitos especializado em regras de negócio.
Contexto	Sistema [nome do sistema] com requisitos funcionais estabelecidos.
Tarefa	Identificar e detalhar regras de negócio detalhadas, incluindo definições, condições, ações exceções e exemplos.
Formato de Saída	Catálogo estruturado de regras de negócio.

Prompt completo	Aja como um Analista de requisitos experiente em regras de negócio. Contexto: Estamos desenvolvendo um sistema [nome do sistema] com os seguintes requisitos funcionais: ""[colar os requisitos funcionais do sistema] """. Tarefa: Identifique e documente as regras de negócio detalhadas que governam o comportamento do sistema. Para cada regra de negócio: 1. Atribua um identificador único (RN-XX) 2. Forneça um nome descritivo 3. Classifique o tipo de regra (cálculo, validação, processo, etc.) 4. Descreva a regra em linguagem natural precisa 5. Especifique condições de aplicação 6. Detalhe as ações ou resultados esperados 7. Documente exceções ou casos especiais 8. Forneça exemplos concretos de aplicação 9. Indique requisitos funcionais relacionados 10. Especifique a origem da regra (política da empresa, regulamentação, etc.) Formato de saída: Catálogo estruturado de regras de negócio, organizado por categorias lógicas, com todas as informações solicitadas para cada regra.
------------------------	--

Fonte: Fonte própria (2025).

5.6.1. Resultados esperados

Espera-se obter um catálogo estruturado de regras de negócio, que sirva como um documento de referência para desenvolvedores, testadores e *stakeholders*. Esse catálogo deve ser completo, consistente e de fácil compreensão, garantindo que o sistema implemente corretamente as políticas do negócio.

5.6.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria

5.3.2.1. Exemplo dos requisitos funcionais

- O sistema deve permitir o registro de entrada de livros no estoque.
- O sistema deve permitir o registro de saída de livros do estoque (venda, perda, devolução ao fornecedor).
- O sistema deve notificar quando o estoque de um livro atingir um nível mínimo.
- O sistema deve permitir a realização de inventário.

5.3.2.2. Exemplo do *prompt*

Aja como um Analista de requisitos experiente em regras de negócio. Contexto: Estamos desenvolvendo um Sistema de Gerenciamento de Estoque para Livraria com os seguintes requisitos funcionais: ""O sistema deve permitir o registro de entrada de livros no estoque. O sistema deve permitir o registro de saída de livros do estoque (venda, perda, devolução ao fornecedor). O sistema deve notificar quando o estoque de um livro atingir um nível mínimo. O sistema deve permitir a realização de inventário."" . Tarefa: Identifique e documente as regras de negócio detalhadas que governam o comportamento do sistema. Para cada regra de negócio: 1. Atribua um identificador único (RN-XX) 2. Descreva a regra de forma clara e concisa 3. Especifique as condições que ativam a regra 4. Descreva as ações que resultam da regra 5. Indique possíveis exceções ou alternativas 6. Forneça um exemplo prático. Formato de saída: Catálogo estruturado de regras de negócio.

5.6.3. Resposta do LLM

Catálogo de Regras de Negócio - Sistema de Gerenciamento de Estoque para Livraria

RN-01: Regra de Nível Mínimo de Estoque

- **Descrição:** O sistema deve alertar o funcionário responsável quando a quantidade em estoque de um livro for igual ou inferior ao seu nível mínimo configurado.
- **Condições:**
 - Quantidade em estoque de um livro $\leq$ Nível mínimo configurado para o livro.
 - Uma movimentação de saída (venda, perda) ocorreu ou o nível foi verificado manualmente.
- **Ações:**
 - Gerar uma notificação visual no sistema para o funcionário do estoque.
 - Opcionalmente, enviar um e-mail ou mensagem para o responsável por compras.
 - Registrar o alerta no histórico do livro.
- **Exceções/Alternativas:** O nível mínimo pode ser configurado por livro. Alertas podem ser desativados temporariamente para livros em promoção ou fora de linha.

- **Exemplo:** O livro "A Arte da Guerra" tem nível mínimo de 5 unidades. Após uma venda, o estoque cai para 4. O sistema exibe um alerta para o funcionário do estoque e sugere a necessidade de reposição.

RN-02: Regra de Registro de Venda

- **Descrição:** Ao registrar uma venda, a quantidade do livro vendido deve ser subtraída do estoque e o histórico de vendas atualizado.
- **Condições:**
 - Venda de um livro é confirmada no sistema.
 - Livro existe no estoque e possui quantidade suficiente.
- **Ações:**
 - Decrementar a quantidade do livro no estoque.
 - Registrar a transação de venda no histórico.
 - Gerar recibo de venda.
- **Exceções/Alternativas:** Se a quantidade em estoque for insuficiente, a venda não pode ser concluída. Pode haver a opção de registrar uma venda parcial ou de reservar o restante para posterior reposição.
- **Exemplo:** Um cliente compra 2 exemplares de "Dom Casmurro". O sistema subtrai 2 do estoque de "Dom Casmurro" e registra a venda.

RN-03: Regra de Inventário Físico

- **Descrição:** Durante o processo de inventário físico, o sistema deve permitir o ajuste das quantidades em estoque para refletir a contagem real, registrando as divergências.
- **Condições:**
 - Funcionário do estoque inicia o processo de inventário para um livro ou seção.
 - Contagem física é inserida no sistema.
- **Ações:**
 - Comparar a contagem física com a quantidade registrada no sistema.
 - Ajustar a quantidade em estoque para a contagem física.
 - Registrar a divergência (sobra ou falta) e a data do inventário.
- **Exceções/Alternativas:** O sistema pode exigir uma justificativa para grandes divergências. Relatórios de divergência podem ser gerados para análise.

- **Exemplo:** O sistema indica 10 unidades de "O Pequeno Príncipe", mas a contagem física revela 9. O sistema ajusta o estoque para 9 e registra uma perda de 1 unidade.

5.7. Matriz de Rastreabilidade de requisitos

A rastreabilidade de requisitos é um aspecto crítico da Engenharia de Requisitos que envolve a capacidade de rastrear a origem, evolução e implementação de cada requisito ao longo do ciclo de vida do projeto. A matriz de rastreabilidade é uma ferramenta que estabelece e mantém essas relações, permitindo o gerenciamento eficaz de mudanças, a verificação de que todos os requisitos foram implementados e testados, e a análise de impacto de modificações. O desafio reside em criar e manter uma matriz abrangente que capture todas as relações relevantes sem se tornar excessivamente complexa ou difícil de gerenciar.

O objetivo dessa atividade é auxiliar na criação de uma matriz de rastreabilidade completa que relacione requisitos com suas origens, dependências, casos de uso, casos de teste e componentes de implementação.

- Identificar a origem de cada requisito.
- Mapear dependências entre requisitos.
- Relacionar requisitos com casos de uso e outros artefatos.
- Estabelecer vínculos com casos de teste e componentes de implementação.

Quadro 7 – Prompt genérico para matriz de rastreabilidade de requisitos

Aspecto	Descrição
Objetivo	Criar matriz de rastreabilidade para gerenciamento de requisitos.
Papel do LLM	Analista de Requisitos especializado em rastreabilidade.
Contexto	Conjunto de requisitos, caso de uso e outros artefatos do projeto.
Tarefa	Desenvolver matriz de rastreabilidade relacionando requisitos com suas origens, dependência, casos de teste e componentes de implementação.
Formato de Saída	Matriz de rastreabilidade no formato tabular.
Prompt completo	Aja como um Analista de Requisitos especializado em rastreabilidade. Contexto: Abaixo estão os requisitos, casos de uso e outros artefatos do projeto [nome do sistema]. Requisitos: "" [Cole aqui a lista de requisitos] "". Casos de uso: "" [Cole aqui os casos de uso] "". Outros artefatos (se aplicável): "" [Cole aqui outros artefatos relevantes] "". Tarefa: Desenvolva uma matriz de rastreabilidade completa que: 1. Relacione cada requisito com sua origem (stakeholder, documento, entrevista) 2. Identifique dependências entre requisitos 3. Mapeie requisitos para casos de uso 4. Associe requisitos a componentes de implementação (se conhecidos) 5. Vincule requisitos a casos de teste (ou indique necessidade de criação) Formato de saída: Matriz de rastreabilidade em formato tabular, com colunas para ID do requisito, descrição, origem, dependências, casos de uso relacionados, componentes, casos de teste.

Fonte: Fonte própria (2025).

5.7.1. Resultados esperados

Espera-se obter uma matriz de rastreabilidade estruturada em formato tabular que facilite o gerenciamento de requisitos, a análise de impacto de mudanças e a verificação de cobertura de testes e implementação. A matriz deve ser clara, completa e de fácil manutenção.

5.7.2. Exemplo prático para um sistema de gerenciamento de estoque para uma pequena livraria

5.7.2.1. Exemplo dos requisitos funcionais

- RF-01: O sistema deve permitir o registro de entrada de livros no estoque.
- RF-02: O sistema deve permitir o registro de saída de livros do estoque.
- RF-03: O sistema deve notificar quando o estoque de um livro atingir um nível mínimo.
- RNF-01: O sistema deve ter tempo de resposta inferior a 3 segundos para consultas de estoque.

5.7.2.2. Exemplo de caso de uso

- UC-01: Registrar Entrada de Livro
- UC-02: Registrar Saída de Livro
- UC-03: Gerenciar Nível Mínimo de Estoque

5.7.2.3. Exemplo do prompt

Aja como um Analista de Requisitos especializado em rastreabilidade. Contexto: Abaixo estão os requisitos, casos de uso e outros artefatos do projeto Sistema de Gerenciamento de Estoque para Livraria. Requisitos: ""RF-01: O sistema deve permitir o registro de entrada de livros no estoque. RF-02: O sistema deve permitir o registro de saída de livros do estoque. RF-03: O sistema deve notificar quando o estoque de um livro atingir um nível mínimo. RNF-01: O sistema deve ter tempo de resposta inferior a 3 segundos para consultas de estoque."" . Casos de uso: ""UC-01: Registrar Entrada de Livro. UC-02: Registrar Saída de Livro. UC-03: Gerenciar Nível Mínimo de Estoque."" . Tarefa: Desenvolva uma matriz de rastreabilidade completa que: 1. Relacione cada requisito com sua origem (stakeholder, documento, entrevista) 2. Identifique dependências entre requisitos 3. Mapeie requisitos para casos de uso 4. Sugira casos de teste relacionados 5. Indique componentes de sistema que implementarão cada requisito. Formato de saída: Matriz de rastreabilidade no formato tabular.

5.7.3. Resposta do LLM

Quadro 8 – Resultado obtido do prompt

ID Requisito	Descrição	Origem	Dependências	Casos de Uso	Casos de Teste Sugeridos	Componentes de Implementação
RF-01	O sistema deve permitir o registro de entrada de livros no estoque	Entrevista com Funcionário do Estoque	-	UC-01	CT-01: Entrada de livro novo CT-02: Entrada de livro existente CT-03: Entrada com ISBN inválido	Módulo de Estoque Interface de Entrada Banco de Dados
RF-02	O sistema deve permitir o registro de saída de livros do estoque	Entrevista com Vendedor	RF-01 (livro deve estar no estoque)	UC-02	CT-04: Saída por venda CT-05: Saída por perda CT-06: Saída de livro inexistente	Módulo de Vendas Módulo de Estoque Interface de Saída
RF-03	O sistema deve	Gerente da Livraria	RF-01, RF-02	UC-03	CT-07: Notificação	Módulo de Alertas

ID Requisito	Descrição	Origem	Dependências	Casos de Uso	Casos de Teste Sugeridos	Componentes de Implementação
	notificar quando o estoque de um livro atingir um nível mínimo				de estoque baixo CT-08: Configuração de nível mínimo	Módulo de Estoque Configurações
RNF-01	O sistema deve ter tempo de resposta inferior a 3 segundos para consultas de estoque	Requisito de Desempenho	Todos os RF	Todos os UC	CT-09: Teste de performance de consulta CT-10: Teste de carga em consultas	Otimização de Banco Cache de Dados Arquitetura do Sistema

Fonte: Fonte própria (2025).

6 ESTUDO DE CASO

Este capítulo apresenta um estudo de caso que foi realizado para avaliar as diretrizes para o uso de LLMs em tarefas de levantamento de requisitos propostas neste trabalho. O capítulo é organizado da seguinte forma: primeiro vai ser apresentado o projeto em que o estudo foi realizado. Em seguida, serão apresentados os resultados do caso de estudo. Por último, será apresentada uma análise dos resultados.

6.1. Projeto do Sistema de Tutoria Inteligente para o Ensino de Física

Este projeto consiste no desenvolvimento de um Sistema de Tutoria Inteligente para o ensino de Física (STI-Física). Trata-se de um projeto educacional em andamento realizado por docentes da Universidade Federal Rural do Semi-Árido (UFERSA) do Campus Pau dos Ferros, focado em desenvolver uma solução para auxiliar alunos e professores do ensino médio de escolas públicas brasileiras no ensino da disciplina de Física, especialmente, aquelas localizadas em áreas rurais e carentes que possuem limitações técnicas e de conectividade. O principal objetivo do STI-Física é mitigar os obstáculos enfrentados no ensino de Física nessas regiões, como a falta de infraestrutura tecnológica, baixa motivação estudantil e a dificuldade de personalização pedagógica em salas de aula numerosas.

Este sistema inovador utilizará Inteligência Artificial generativa para produzir automaticamente exercícios e *feedbacks* personalizados, proporcionando uma experiência de aprendizado adaptativa, interativa e acessível, mesmo em contextos com baixa ou nenhuma conectividade com a internet. O escopo do STI-Física abrange um aplicativo educacional desenvolvido para funcionar de forma *offline*, compatível com dispositivos Android de baixo custo, e destinado ao ensino de Física em escolas públicas de regiões rurais e socialmente vulneráveis.

As funcionalidades essenciais incluem diagnóstico inicial do aluno, geração de exercícios e *feedbacks* personalizados com uso de IA generativa que não dependem de conexão com a Internet. Além disso, o sistema permitirá que professores planejem e organizem aulas, elaborem atividades adaptadas ao perfil da turma e acompanhem o desempenho dos alunos por meio de relatórios detalhados. Para o engajamento estudantil, o sistema incorporará elementos de gamificação, como desafios em grupo, *rankings* e *badges*.

Os *stakeholders* envolvidos neste projeto incluem alunos, professores, gestores escolares e a equipe de desenvolvimento. Os requisitos do sistema são centrados na acessibilidade, adaptabilidade, engajamento e na capacidade de operar em ambientes com

recursos limitados. O contexto do projeto é o de um esforço contínuo para promover a equidade educacional e melhorar os resultados de aprendizagem em Física no Brasil. Na Tabela 10 podemos ver um resumo dos principais stakeholders do projeto.

Quadro 9 – *Stakeholders* e responsabilidades

Stakeholder	Descrição	Responsabilidades
Alunos	Estudantes de ensino médio que utilizam o STI para aprender física	• Acessar conteúdos didáticos e simuladores de Física disponíveis. • Realizar exercícios e avaliações. • Interagir com professores e colegas de turma. • Acompanhar seu progresso no aprendizado.
Professores	Educadores responsáveis por ministrar aulas de Física e utilizam o STI como ferramenta de ensino.	• Criar e gerenciar planos de aula. • Criar e compartilhar listas de exercícios e avaliações. • Monitorar o desempenho dos alunos pela qual é responsável. • Interagir com os alunos via fóruns ou vídeo aula.
Gestores escolar	Gestores educacionais que supervisionam o ensino de física nas escolas	• Acompanhar o uso do STI e de outras plataformas pelos professores e alunos. • Avaliar relatórios de desempenho acadêmico. • Garantir a adoção da plataforma nas escolas.

Fonte: Autoria própria (2025)

6.2. Experimentos com LLMs no Levantamento de Requisitos do STI-Física

Nesta seção, serão apresentados os experimentos realizados utilizando LLMs para auxiliar nas atividades de levantamento de requisitos do STI-Física. As atividades foram conduzidas seguindo as diretrizes propostas no Capítulo anterior deste trabalho, adaptando os prompts genéricos para o contexto específico do STI-Física. Para cada atividade, será descrito o que foi gerado pelo LLM e, quando cabível, serão incluídos trechos dos resultados obtidos.

6.1.1. Compreensão do Domínio do Negócio

Atividade: Geração de perguntas essenciais para compreender o contexto de negócio, processos atuais, dores e necessidades do STI-Física, sob a perspectiva de um analista de negócios.

6.1.2. Prompt

Aja como um Analista de requisitos experiente. Estou iniciando um projeto de desenvolvimento de software para o desenvolvimento de um aplicativo de física que vai ajudar os alunos no estudo da disciplina. Tarefa: com base no domínio informado, gere uma lista de 10 perguntas essenciais que eu deveria fazer aos stakeholders para compreender adequadamente o contexto de negócio, processos atuais, dores e necessidades. Formato de saída: Lista numerada de perguntas, agrupadas por categoria (ex: contexto organizacional, processos atuais, problemas e necessidades).

6.1.3. Resultado gerado pelo LLM

O LLM gerou uma lista de 10 perguntas essenciais, divididas nas seguintes categorias:

- Contexto Educacional: Perguntas sobre o objetivo pedagógico central do aplicativo, alinhamento com diretrizes curriculares (BNCC) e políticas da rede pública, e recursos tecnológicos disponíveis nas escolas.
- Processos Atuais: Perguntas sobre como os alunos estudam física atualmente, como os professores planejam e conduzem as aulas, e como a avaliação do aprendizado é realizada.

- Problemas e Necessidades: Perguntas sobre os maiores desafios no ensino e aprendizado de física, lacunas no material didático existente, e como o aplicativo poderia resolver esses problemas.

Algumas das perguntas geradas são apresentadas abaixo:

1. Qual é o objetivo pedagógico central do aplicativo? (Ex.: Reforçar conceitos teóricos, substituir laboratórios físicos, preparar para o ENEM).
2. Como o aplicativo se alinha às diretrizes curriculares (ex.: BNCC) e às políticas da rede pública?
3. Quais são os recursos tecnológicos disponíveis nas escolas (ex.: internet, dispositivos) e as limitações críticas (ex.: conectividade irregular)?

6.1.4. Identificação dos *Stakeholders*

Atividade: Identificação de stakeholders relevantes para o processo de elicitação de requisitos do STI-Física, explicando sua importância, informações que podem fornecer e técnicas de elicitação adequadas.

6.1.5. *Prompt*

Aja como um Gerente de Projetos com experiência em Engenharia de Requisitos. Contexto: Estou iniciando um projeto para desenvolver um aplicativo de física que vai ajudar os alunos no estudo da disciplina. Tarefa: Identifique os potenciais *stakeholders* que deveriam ser consultados durante a fase de elicitação de requisitos. Para cada tipo de stakeholder, explique brevemente: 1. Por que este *stakeholder* é importante para o projeto 2. Quais informações específicas este *stakeholder* pode fornecer 3. Quais técnicas de elicitação seriam mais adequadas para este *stakeholder*. Formato de saída: Lista de *stakeholders* com as informações solicitadas.

6.1.6. Resultado gerado pelo LLM

O LLM identificou os seguintes *stakeholders* para o projeto STI-Física:

- Alunos do Ensino Médio:
 - Importância: Usuários finais, suas necessidades e *feedback* são cruciais para a usabilidade e eficácia do aplicativo.
 - Informações: Dificuldades de aprendizado, preferências de interação, feedback sobre exercícios e gamificação.
 - Técnicas de Elicitação: Questionários, grupos focais, observação de uso, entrevistas.
- Professores de Física do Ensino Médio:
 - Importância: Usuários-chave, responsáveis pela integração do aplicativo no currículo e acompanhamento do progresso dos alunos.
 - Informações: Necessidades pedagógicas, tipos de exercícios desejados, requisitos para relatórios de desempenho, integração com o plano de aula.
 - Técnicas de Elicitação: Entrevistas, workshops, prototipagem colaborativa.
- Coordenadores Pedagógicos/Diretores de Escola:
 - Importância: Responsáveis pela aprovação e implementação do aplicativo na escola, preocupados com o alinhamento pedagógico e impacto educacional.
 - Informações: Requisitos de alinhamento com a BNCC, políticas educacionais, infraestrutura tecnológica da escola, métricas de sucesso em nível institucional.
 - Técnicas de Elicitação: Entrevistas de alto nível, reuniões de apresentação.
- Especialistas em Conteúdo (Física):
 - Importância: Garantem a precisão e a qualidade do conteúdo didático e dos exercícios.
 - Informações: Validação de conceitos de física, sugestões de exercícios, correção de feedbacks gerados pela IA.
 - Técnicas de Elicitação: Revisão de conteúdo, reuniões técnicas.
- Desenvolvedores/Equipe Técnica:
 - Importância: Responsáveis pela implementação do sistema, precisam entender os requisitos técnicos e as limitações.

- Informações: Viabilidade técnica dos requisitos, estimativas de esforço, sugestões de arquitetura.
- Técnicas de Elicitação: Reuniões de requisitos, discussões técnicas.

Trecho do Resultado (Exemplo):

- Importância: Usuários finais, suas necessidades e feedback são cruciais para a usabilidade e eficácia do aplicativo.
- Informações: Dificuldades de aprendizado, preferências de interação, feedback sobre exercícios e gamificação.
- Técnicas de Elicitação: Questionários, grupos focais, observação de uso, entrevistas.

6.1.7. Requisitos Funcionais

Atividade: Extração de requisitos funcionais a partir de uma transcrição de entrevista com um stakeholder do projeto STI-Física.

6.1.8. Prompt

Aja como um Analista de Requisitos experiente. Contexto: A transcrição em anexo é de uma entrevista com [cargo do *stakeholder*] sobre o sistema [nome do sistema]. Tarefa: Analise a transcrição e extraia todos os requisitos funcionais mencionados explícita ou implicitamente. Para cada requisito: 1. Descreva-o em uma frase clara e objetiva 2. Indique se foi mencionado explicitamente ou está implícito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas). Transcrição: "" [Transcrição da entrevista] "". Formato de saída: Lista numerada de requisitos funcionais com as informações solicitadas.

6.1.9. Resultado gerado pelo LLM

O LLM extraiu diversos requisitos funcionais da transcrição fornecida, incluindo:

- Funcionalidade Offline: O sistema deve operar *offline*, permitindo acesso a conteúdo e simuladores sem conexão à internet.

- **Compatibilidade com Dispositivos Móveis:** O sistema deve ser acessível via celular, considerando a realidade de alunos que não possuem computadores.
- **Integração de Simuladores Interativos:** O sistema deve incluir simuladores de física (ex.: circuitos elétricos, mecânica) que funcionem *offline* e *online*.
- **Geração de Exercícios Personalizados:** O sistema deve gerar exercícios personalizados com base no desempenho do aluno.
- **Feedback Imediato e Detalhado:** O sistema deve fornecer *feedback* imediato e detalhado sobre as respostas dos alunos.
- **Gamificação:** O sistema deve incorporar elementos de gamificação para aumentar o engajamento dos alunos.

Alguns dos requisitos funcionais gerados são apresentados abaixo:

1. Funcionalidade *Offline*

- **Descrição:** O sistema deve operar *offline*, permitindo acesso a conteúdo e simuladores sem conexão à internet.
- **Tipo:** Explícito
- **Evidência:** "A ideia desse projeto é que a gente possa desenvolver uma aplicação que ela rode localmente [...] uma aplicação mobile. Que o aluno consiga ter acesso, mesmo em condições de baixa conectividade com a internet, ou até mesmo sem internet".

2. Compatibilidade com Dispositivos Móveis

- **Descrição:** O sistema deve ser acessível via celular, considerando a realidade de alunos que não possuem computadores.
- **Tipo:** Explícito
- **Evidência:** "E que possa rodar localmente com o celular. [...] um celular já é mais normal ter a possibilidade de fazer uso".

3. Banco de Questões Adaptáveis

- Descrição: O sistema deve oferecer um banco de questões filtradas por conteúdo, nível de dificuldade e contexto (ex.: ENEM), permitindo adaptações pelo professor.
- Tipo: Explícito
- Evidência: "Eu vou em sites [...] filtro por conteúdo, por nível de dificuldade, por banca. [...] faço adaptações para aproximar do nível do aluno".

4. Trilhas de Aprendizado Personalizadas

- Descrição: O sistema deve guiar o aluno por sequências lógicas de conceitos, identificando lacunas de conhecimento (ex.: matemática básica) e sugerindo atividades de reforço.
- Tipo: Explícito
- Evidência: "Uma ferramenta que ajude o aluno a trilhar consensualmente [...] se o aluno tem dificuldade em cálculo, sugere atividades sobre potenciação antes de avançar".

6.1.10. Requisitos Não-funcionais

Atividade: Extração de requisitos não-funcionais a partir de uma transcrição de entrevista com um stakeholder do projeto STI-Física.

6.1.11. *Prompt*

Aja como um Engenheiro de software especializado em requisitos não-funcionais. Contexto: A transcrição em anexo é de uma entrevista com [cargo do stakeholder] sobre o sistema [nome do sistema]. Tarefa: Analise a transcrição e extraia todos os requisitos não-funcionais mencionados explícita ou implicitamente. Para cada requisito: 1. Descreva-o em uma frase clara e objetiva 2. Indique se foi mencionado explicitamente ou está implícito 3. Cite a parte da transcrição que evidencia este requisito (entre aspas). Formato de saída: Lista numerada de requisitos não-funcionais com as informações solicitadas, agrupados por categoria (ex: Desempenho, Segurança, Usabilidade, Confiabilidade, Manutenibilidade).

6.1.12. Resultado gerado pelo LLM

O LLM extraiu diversos requisitos não-funcionais da transcrição fornecida, incluindo:

- Portabilidade: O aplicativo deve operar *offline* em dispositivos móveis (celulares), sem dependência de internet.
- Usabilidade: Interface intuitiva e engajadora, evitando designs excessivamente textuais (como slides) que causam desatenção.
- Desempenho: O sistema deve ter um tempo de resposta rápido para a geração de exercícios e feedbacks.
- Confiabilidade: O sistema deve ser estável e não apresentar falhas frequentes.
- Segurança: Os dados dos alunos e professores devem ser protegidos.

Alguns dos requisitos funcionais gerados são apresentados abaixo:

Portabilidade:

1. Descrição: O aplicativo deve operar *offline* em dispositivos móveis (celulares), sem dependência de internet.
 - Categoria: Portabilidade (Subcaracterística: Adaptabilidade)
 - Evidência: "Desenvolver uma aplicação que ela rode localmente [...] uma aplicação mobile. Que o aluno consiga ter acesso, mesmo em condições de baixa conectividade com a internet, ou até mesmo sem internet".
 - Métrica de Aceitação: Funcionalidade 100% operacional após 30 minutos de uso offline, sem degradação de desempenho.

Usabilidade:

2. Descrição: Interface intuitiva e engajadora, evitando designs excessivamente textuais (como slides) que causam desatenção.
 - Categoria: Usabilidade (Subcaracterística: Satisfação)
 - Evidência: "Eu não quero que seja só texto, tipo slide. Tem que ser algo que prenda a atenção, que seja fácil de usar."

Desempenho:

3. Descrição: Consumo mínimo de recursos (CPU, memória) para funcionar em celulares antigos ou de baixo custo.

- Categoria: Desempenho (Subcaracterística: Utilização de Recursos)
- Evidência: "Sempre existe um celular [...] em certas situações de vulnerabilidade social".
- Métrica de Aceitação: Consumo de RAM $\leq$ 150 MB em dispositivos com Android 8.0 (Oreo) ou superior.

- Confiabilidade:

4. Descrição: Funcionamento estável em redes instáveis ou intermitentes, sem perda de dados.

- Categoria: Confiabilidade (Subcaracterística: Tolerância a Falhas)
- Evidência: "A internet das escolas é horríveis [...] a gente prefere usar aqueles que funcionam offline".
- Métrica de Aceitação: 99% de sucesso em operações de salvamento automático durante simulações de queda de conexão.
- Compatibilidade:

5. Descrição: Compatibilidade com dispositivos pessoais dos alunos (celulares), mesmo em escolas com restrições a aparelhos.

- Categoria: Compatibilidade (Subcaracterística: Coexistência)
- Evidência: "Para fins pedagógicos, posso autorizar os alunos a levar celular [...] é interessante cada aluno ter seu dispositivo".
- Métrica de Aceitação: Funcionamento verificado em 95% dos modelos de celulares (Android/iOS) usados por alunos em 5 escolas piloto.

6.3. Análise dos Resultados

Para analisar os resultados da aplicação das diretrizes propostas, foi realizada uma avaliação qualitativa sobre o desempenho do Modelo de Linguagem de Grande Escala LLM nas tarefas de levantamento de requisitos do projeto STI-Física. O objetivo desta análise é discutir, com base nas observações do experimento, a eficácia, os desafios e o potencial da ferramenta. Para estruturar esta avaliação, foi utilizada uma Análise SWOT, focada exclusivamente no processo de utilização dos LLMs.

Após a análise individual de cada experimento, é possível consolidar os achados em uma discussão geral sobre a eficácia do LLMs como ferramenta de apoio à eliciação de requisitos, utilizando o framework SWOT.

1. Forças (*Strengths*): A principal força dos LLMs, observada consistentemente em todas as atividades, é sua capacidade de acelerar tarefas de estruturação e formalização. Ele atua como um assistente interativo que combate a síndrome da página em branco, fornecendo rapidamente listas de perguntas, mapas de *stakeholders*, rascunhos de requisitos e especificações de casos de uso. Sua eficiência no processamento de dados não estruturados, como transcrições de entrevistas, é um diferencial claro para a produtividade.

2. Fraquezas (*Weaknesses*): A principal fraqueza reside na falta de empatia, nuance e expertise de domínio real. O LLMs não substitui a intuição e a experiência de um analista de requisitos. Ele pode gerar respostas plausíveis, mas superficiais, e tem dificuldade em capturar requisitos implícitos complexos ou em compreender o contexto político e emocional de um projeto. A qualidade de sua saída é diretamente dependente da qualidade do *prompt*, exigindo habilidade do operador.

3. Oportunidades (*Opportunities*): A maior oportunidade é a utilização dos LLMs como uma ferramenta de apoio à documentação em tempo real e à promoção de uma abordagem mais holística. Ele pode padronizar e agilizar a criação de artefatos, permitindo que o engenheiro de requisitos se concentre em tarefas de maior valor, como a negociação com stakeholders e a análise crítica. Além disso, pode servir como uma ferramenta de treinamento para analistas menos experientes.

4. Ameaças (*Threats*): A ameaça mais significativa é a confiança excessiva e a subestimação dos riscos. Um uso acrítico da ferramenta pode levar à inclusão de requisitos incorretos ("alucinações"), à negligência de *insights* puramente humanos e a uma falsa sensação de segurança. Questões de confidencialidade dos dados inseridos no *prompt* também representam uma ameaça que precisa ser gerenciada.

Em suma, a experimentação prática no contexto do STI-Física revelou que o LLM é uma ferramenta de apoio extremamente promissora, mas que exige uma postura de ceticismo saudável e validação humana rigorosa. Seu papel é o de um assistente que potencializa a capacidade do analista, e não o de um agente autônomo.

6.4. Discussão dos Resultados e Limitações da Abordagem

A análise do desempenho do LLM, estruturada pela SWOT, permite extrair conclusões importantes sobre seu papel na Engenharia de Requisitos e reconhecer as limitações deste estudo.

6.4.1. Discussão dos Achados

Os resultados do estudo de caso demonstram que os LLMs, quando guiados por diretrizes claras, são ferramentas de grande potencial para otimizar o processo de levantamento de requisitos. A principal contribuição observada é a aceleração de tarefas de formalização e estruturação. A capacidade de transformar rapidamente transcrições de entrevistas em listas de requisitos ou de gerar perguntas contextuais (Forças) representa um ganho de produtividade significativo.

No entanto, o experimento também evidenciou que o LLM atua como um assistente qualificado, e não como um substituto para o engenheiro de requisitos. As Fraquezas identificadas, como a dependência da qualidade do *prompt* e a necessidade de validação humana constante, reforçam a ideia de que a expertise humana é insubstituível. O pensamento crítico do analista é crucial para filtrar "alucinações", refinar saídas genéricas e, principalmente, para identificar os requisitos implícitos que a máquina não consegue captar.

A análise SWOT revelou que o valor do LLM não está em "dar a resposta certa", mas em oferecer um ponto de partida robusto. A ferramenta combate a "síndrome da página em branco", permitindo que o analista comece seu trabalho com uma base já estruturada, que pode então ser criticada, refinada e validada.

6.4.2. Limitações do trabalho

Este trabalho, embora demonstre o potencial dos LLMs, possui algumas limitações importantes que devem ser consideradas:

- Escopo do Estudo de Caso: A experimentação foi realizada no contexto de um único projeto (STI-Física). A aplicabilidade e a eficácia das diretrizes podem variar em domínios de negócio diferentes e com outras complexidades.
- Falta de Avaliação Quantitativa: A análise foi primariamente qualitativa, baseada na percepção do pesquisador. Uma avaliação mais abrangente envolveria métricas quantitativas, como o tempo economizado em cada tarefa ou a comparação da qualidade dos requisitos gerados (com e sem LLM) por especialistas independentes.
- Modelo de LLM Específico: Os experimentos foram conduzidos com um modelo de LLM específico. A performance e os resultados podem variar significativamente com o uso de outros modelos, que possuem diferentes arquiteturas e dados de treinamento.
- Validação Humana Centralizada: A validação dos resultados gerados pelos LLMs foi realizada pelo próprio pesquisador. Um processo mais robusto incluiria a validação por múltiplos especialistas em Engenharia de Requisitos e no domínio do projeto para reduzir vieses.

Essas limitações destacam a necessidade de futuras pesquisas para aprimorar e generalizar a integração dos LLMs no ciclo de vida da Engenharia de Requisitos.

7. CONSIDERAÇÕES FINAIS

Este trabalho de conclusão de curso investigou a aplicação de Modelos de Linguagem de Grande Escala LLMs na atividade de levantamento de requisitos em Engenharia de Software, com foco particular na etapa de elicitação. A pesquisa buscou abordar os desafios inerentes ao processo tradicional de levantamento de requisitos, como a comunicação ineficaz, requisitos ambíguos e voláteis, e o alto custo de erros identificados tardiamente. Em resposta a esses desafios, o estudo propôs e avaliou o potencial dos LLMs como ferramentas auxiliares para otimizar e aprimorar a qualidade da elicitação de requisitos.

7.1. Principais contribuições

As principais contribuições deste trabalho materializam-se em duas frentes distintas, porém complementares: a elaboração de um conjunto de diretrizes para o uso de LLMs no levantamento de requisitos e a validação empírica dessas diretrizes por meio de um estudo de caso prático.

7.1.1. Diretrizes para o Uso de LLMs no Levantamento de Requisitos

Um dos pilares deste estudo foi a criação de um guia detalhado, ou um conjunto de diretrizes, para a utilização eficaz de LLMs em diversas atividades do levantamento de requisitos. Baseadas em uma revisão bibliográfica abrangente e em explorações práticas, essas diretrizes foram estruturadas para auxiliar engenheiros de requisitos e desenvolvedores a maximizar o potencial dos LLMs. As diretrizes abordam aspectos cruciais como:

- **Compreensão do Domínio do Negócio:** Fornecendo *prompts* estruturados para gerar perguntas essenciais que permitam uma imersão profunda no contexto do cliente, identificando processos, dores e necessidades. Isso visa superar a dificuldade de extrair informações completas e precisas de stakeholders que podem não ter clareza sobre suas próprias necessidades.
- **Identificação de *Stakeholders*:** Oferecendo um método para mapear todas as partes interessadas, suas responsabilidades, nível de influência e as informações que podem fornecer, garantindo que todas as perspectivas relevantes sejam consideradas e mitigando o risco de requisitos incompletos.

- **Análise de Transcrições de Entrevistas:** Propondo técnicas para extrair requisitos funcionais de forma precisa e completa a partir de dados não estruturados, incluindo a distinção entre requisitos explícitos e implícitos e a citação de evidências textuais.
- **Identificação de Requisitos Não-Funcionais:** Auxiliando na detecção e formalização de atributos de qualidade e restrições do sistema (desempenho, segurança, usabilidade, confiabilidade, manutenibilidade), que são frequentemente mais desafiadores de elicitar por serem menos tangíveis.
- **Especificação de Casos de Uso:** Facilitando a criação de especificações claras, completas e não ambíguas de casos de uso, detalhando interações entre usuários e o sistema, incluindo fluxos alternativos e de exceção, o que pode ser um processo demorado e propenso a erros quando feito manualmente.
- **Especificação de Regras de Negócio:** Auxiliando na documentação detalhada de regras que governam o comportamento do sistema, garantindo que o software reflita com precisão as políticas, procedimentos e leis da organização.
- **Matriz de Rastreabilidade de Requisitos:** Oferecendo um método para criar e manter uma matriz abrangente que relacione requisitos com suas origens, dependências, casos de uso, casos de teste e componentes de implementação, crucial para o gerenciamento eficaz de mudanças e a verificação da cobertura.

Essas diretrizes representam um arcabouço prático que capacita os profissionais a integrarem os LLMs de forma eficiente e responsável em suas práticas de Engenharia de Requisitos, transformando-os em assistentes inteligentes capazes de otimizar tarefas tradicionalmente manuais e complexas. A abordagem detalhada para a construção de prompts e a especificação de resultados esperados para cada tarefa demonstra a aplicabilidade e o valor agregado dos LLMs no ciclo de vida do desenvolvimento de software.

7.1.2. Estudo de Caso: Sistema Tutor Inteligente (STI) para o Ensino de Física

A segunda contribuição fundamental deste trabalho foi a validação empírica das diretrizes propostas por meio de um estudo de caso aplicado ao desenvolvimento de um Sistema Tutor Inteligente (STI) para o ensino de Física. Este estudo de caso, focado em um cenário real de desenvolvimento de software, permitiu avaliar a eficácia e a aplicabilidade dos

LLMs na elicitação de requisitos em um contexto prático e desafiador, caracterizado pela necessidade de operar em ambientes com recursos limitados e baixa conectividade.

No estudo de caso, as diretrizes foram aplicadas para auxiliar em diversas atividades de levantamento de requisitos do STI-Física, incluindo:

- **Compreensão do Domínio do Negócio:** O LLM gerou perguntas essenciais sobre o objetivo pedagógico, alinhamento curricular e recursos tecnológicos disponíveis, demonstrando sua capacidade de auxiliar na imersão inicial no projeto.
- **Identificação de *Stakeholders*:** O LLM identificou *stakeholders* cruciais como alunos, professores, coordenadores pedagógicos e especialistas em conteúdo, detalhando sua importância, informações que poderiam fornecer e técnicas de elicitação adequadas. Isso validou a capacidade do LLM de mapear o universo de influências do projeto.
- **Extração de Requisitos Funcionais:** A partir de transcrições de entrevistas, o LLM foi capaz de extrair requisitos funcionais como funcionalidade *offline*, compatibilidade com dispositivos móveis, integração de simuladores interativos, geração de exercícios personalizados, *feedback* imediato, gamificação, banco de questões adaptáveis, trilhas de aprendizado personalizadas e relatórios de desempenho detalhados. Isso evidenciou a habilidade do LLM em processar linguagem natural e identificar funcionalidades essenciais.

Extração de Requisitos Não-Funcionais: O LLM também identificou requisitos não-funcionais importantes, como portabilidade, usabilidade, desempenho, confiabilidade e segurança, a partir das transcrições, reforçando sua utilidade na formalização de atributos de qualidade do sistema.

Os resultados do estudo de caso demonstraram que os LLMs, quando utilizados com as diretrizes propostas, podem ser ferramentas valiosas para otimizar o processo de levantamento de requisitos. Eles se mostraram eficazes na geração de insights, na identificação de informações cruciais e na estruturação de requisitos, acelerando o processo e potencialmente reduzindo a ambiguidade. A aplicação em um projeto com características específicas, como o STI-Física, reforça a adaptabilidade e o potencial dos LLMs em diferentes contextos de desenvolvimento de software.

7.2. Trabalhos Futuros

O presente trabalho abre diversas avenidas para futuras pesquisas e desenvolvimentos na área de aplicação de LLMs na Engenharia de Requisitos. Algumas direções promissoras incluem:

- **Validação em Larga Escala:** Realizar estudos empíricos com um número maior de participantes (engenheiros de requisitos, desenvolvedores) e em diferentes contextos de projeto para validar a robustez e a generalização das diretrizes propostas. Isso incluiria a comparação quantitativa da eficiência e qualidade dos requisitos elicitados com e sem o auxílio de LLMs.
- **Como desdobramento desta pesquisa,** propõe-se como trabalho futuro a realização de uma validação formal dos artefatos gerados durante o experimento com o STI junto à equipe do projeto. Esta etapa visa coletar percepções qualitativas e técnicas dos profissionais envolvidos, permitindo aferir a aplicabilidade prática, a clareza e a utilidade dos artefatos no contexto real de desenvolvimento. Tal validação é fundamental para consolidar os resultados obtidos e garantir que as contribuições propostas estejam alinhadas às necessidades e dinâmicas operacionais da equipe.
- **Experimentação com outros LLMs:** Conduzir novos experimentos utilizando outros Modelos de Linguagem de Grande Escala disponíveis no mercado (e.g., GPT-4, Gemini, etc.) para comparar seu desempenho e identificar qual modelo se adapta melhor a cada tipo de tarefa de elicitação de requisitos. Isso permitiria uma análise comparativa aprofundada das capacidades e limitações de diferentes arquiteturas de LLMs.
- **Análise de Impacto na Qualidade do Software:** Investigar o impacto direto da utilização de LLMs na fase de elicitação de requisitos na qualidade final do software desenvolvido, medindo métricas como número de defeitos relacionados a requisitos, tempo de desenvolvimento e satisfação do cliente.
- **Estudo de Caso em Novos Domínios:** Aplicar as diretrizes e realizar novos estudos de caso em domínios de software distintos (e.g., sistemas financeiros, saúde, e-commerce) para verificar a adaptabilidade dos LLMs e das diretrizes a diferentes complexidades e especificidades de negócio.
- **Considerações Éticas e de Viés:** Aprofundar a pesquisa sobre os vieses que podem ser introduzidos pelos LLMs no processo de elicitação de requisitos e desenvolver estratégias para mitigá-los, garantindo que os requisitos sejam justos, imparciais e representem todas as partes interessadas.

- Treinamento e Ajuste Fino de LLMs Específicos para ER: Investigar a possibilidade de treinar ou realizar o ajuste fino (*fine-tuning*) de LLMs em grandes volumes de dados específicos de Engenharia de Requisitos (documentos de requisitos, casos de uso, transcrições de entrevistas de projetos reais) para otimizar seu desempenho em tarefas de elicitación.

Essas futuras linhas de pesquisa contribuirão para consolidar o papel dos LLMs como ferramentas transformadoras na Engenharia de Requisitos, pavimentando o caminho para processos de desenvolvimento de software mais eficientes, precisos e adaptáveis às crescentes demandas do mercado.

REFERÊNCIAS

ANGELA, W. et al. **Large Language Models for Software Engineering: Survey and open Problems.** *arXiv*, 2023. Disponível em: <https://arxiv.org/pdf/2310.03533>. Acesso em: 11 mai. 2025.

ARORA, C.; GRUNDY, J.; ABDELRAZEK, M. **Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs.** In: **Generative AI for Effective Software Development.** Springer, 2024. p. 129-148. Disponível em: https://link.springer.com/chapter/10.1007/978-3-031-55642-5_6. Acesso em: 19 jun. 2025.

BROWN, T. B. et al. **Language Models are Few-Shot Learners.** In: **ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS**, 33., 2020, [S. l.]. Anais [...]. [S. l.: s. n.], 2020. p. 1877-1901.

CHEN, T. et al. **Large Language Models for Software Engineering: A Systematic Literature Review.** *arXiv*, 2023. Disponível em: <https://arxiv.org/abs/2308.10620>. Acesso em: 3 jul. 2025.

DEEPSEEK AI et al. **DeepSeek LLM: Scaling Open-Source Language Models with Longtermism.** [S. l.]: DeepSeek, 2024. Disponível em: <https://huggingface.co/collections/Presidentlin/deepseek-papers-674c536aa6acddd9bc98c2ac>. Acesso em: 29 jul. 2025.

DEVLIN, J. et al. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.** In: **CONFERENCE OF THE NORTH AMERICAN CHAPTER OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS: HUMAN LANGUAGE TECHNOLOGIES, 2019, Minneapolis. Proceedings.** Minneapolis: Association for Computational Linguistics, 2019. p. 4171-4186.

GLINZ, M. et al. **Certified Professional for Requirements Engineering (CPRE) - Foundation Level: Guia de Estudo.** ISQI, 2024. Disponível em: https://isqi.org/media/b2/9c/54/1710758989/cpre_foundationlevel_handbook_BR_v1.1.1.pdf. Acesso em: 30 jul. 2025.

HABIBA, U. et al. **Quão madura é a Engenharia de Requisitos para Sistemas Baseados em IA? Um Estudo de Mapeamento Sistemático sobre Práticas, Desafios e Direções Futuras de Pesquisa.** Springer Nature, 2024.

LEVENTAMENTO de requisitos 2.4. **Carreira em TI.** Disponível em: <https://carreiraemti.com.br/glossary/levantamento-de-requisitos/>. Acesso em: 1 ago. 2025.

MARQUES, R. R. S.; BERNARDINO, J. **Using ChatGPT in Software Requirements Engineering: A Comprehensive Review.** ResearchGate, 2024. Disponível em: https://www.researchgate.net/publication/380777017_Using_ChatGPT_in_Software_Requirements_Engineering_A_Comprehensive_Review. Acesso em: 12 mai. 2025.

OPENAI. **GPT-4 Technical Report.** OpenAI, 2023. Disponível em: <https://arxiv.org/pdf/2303.08774>. Acesso em: 31 jul. 2025.

PRESSMAN, R. S. **Engenharia de Software: Uma Abordagem Profissional**. 9. ed. Porto Alegre: AMGH, 2021.

RADFORD, A. et al. **Improving Language Understanding by Generative Pre-Training**. [S. l.]: OpenAI, 2018.

REINEHR, S. **Engenharia de requisitos**. [S. l.]: Scribd, 2020. Disponível em: <https://pt.scribd.com/document/636508267/Untitled>. Acesso em: 30 abr. 2025.

RONANKI, K. **Enhancing Requirements Engineering Practices Using Large Language Models**. 2024. Tese (Doutorado em Engenharia de Software) – University of Gothenburg, Gotemburgo, 2024. Disponível em: <https://gupea.ub.gu.se/handle/2077/83054>. Acesso em: 23 jun. 2025.

SILVA, E. G. **Uso de LLMs em atividades de Engenharia de Requisitos: Um Estudo Empírico**. 2025.

SILVA, M. L. da. **Aplicação de LLMs na modelagem de requisitos: melhorando a criação de diagramas UML e estimulando a criatividade em novos casos de uso**. 2024. Trabalho de Conclusão de Curso (Graduação em Engenharia de Software) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2024. Disponível em: <https://lume.ufrgs.br/handle/10183/284220>. Acesso em: 3 jul. 2025.

SOMMERVILLE, I. **Engenharia de software**. 9. ed. São Paulo: Pearson, 2011. E-book.

SOMMERVILLE, I. **Engenharia de Software**. 10. ed. São Paulo: Pearson Education do Brasil, 2020.

TOUVRON, H. et al. **LLaMA: Open and Efficient Foundation Language Models**. [S. l.]: arXiv, 2023. Disponível em: <https://arxiv.org/abs/2302.13971>. Acesso em: 1 ago. 2025.

VASWANI, A. et al. **Attention Is All You Need**. In: **ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS**, 30., 2017, Long Beach. Proceedings [...]. Long Beach: NIPS, 2017. p. 5998-6008. Disponível em: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>. Acesso em: 31 jul. 2025.

VAZQUEZ, C. E.; SIMÕES, G. S. **Engenharia de requisitos: software orientado ao negócio**. 1. ed. Rio de Janeiro: Brasport, 2016. E-book.

WANG, X. et al. **Exploring the Use of Large Language Models in Requirements Engineering**. In: **INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING AND ARTIFICIAL INTELLIGENCE**, 3., 2024, New York. **Proceedings**. New York: ACM, 2024. p. 11-18. Disponível em: <https://dl.acm.org/doi/10.1145/3701625.3701687>. Acesso em: 3 jul. 2025.

ZHANG, X.; LI, Y.; WANG, H. **Large Language Model for Requirements Engineering: A Systematic Literature Review**. [S. l.]: ResearchGate, 2024. Disponível em:

https://www.researchgate.net/publication/386905816_Large_Language_Model_for_Requirements_Engineering_A_Systematic_Literature_Review. Acesso em: 3 jul. 2025.

ZHAO, W. X. et al. **A survey of large language models**. [S. l.]: arXiv, 2023. Disponível em: <https://arxiv.org/abs/2303.18223>. Acesso em: 31 jul. 2025.