Avaliação do Impacto da Engenharia de Prompt na Elicitação e Documentação de Requisitos com LLMs

Álvaro Soares dos Santos[0009-0008-1866-9809] and Lyrene Fernandes da Silva[0000-0003-1772-6062]

Universidade Federal do Rio Grande do Norte, UFRN, Brasil
alvaroass2110@gmail.com, lyrene.silva@ufrn.br

Resumo. A Engenharia de Requisitos é uma etapa fundamental no desenvolvimento de sistemas, responsável por identificar, compreender e documentar as necessidades dos *stakeholders* a partir da análise do contexto do sistema. Avanços recentes em *Large Language Models* (LLMs) têm impulsionado seu uso como ferramenta de apoio à Engenharia de Requisitos, embora ainda existam desafios relacionados à qualidade, consistência e confiabilidade dos resultados gerados. Este estudo avalia o impacto da Engenharia de Prompt nesse contexto, comparando diferentes abordagens de elicitação e escrita de requisitos. Para isso, foi definido um pipeline de *prompts* dividido em contextualização, elicitação (entrevista e seis chapéus do pensamento) e documentação (estruturada e não estruturada), cujos resultados foram avaliados com base em critérios de qualidade de requisitos. Os resultados indicam que *prompts* estruturados produzem requisitos mais consistentes e rastreáveis, enquanto *prompts* não estruturados apresentam maior flexibilidade e cobertura do domínio. Além disso, a técnica dos seis chapéus do pensamento apresentou melhores resultados nos critérios avaliados, enquanto entrevistas geraram informações mais focadas e alinhadas ao contexto do sistema. Conclui-se que a Engenharia de Prompt e as técnicas de elicitação utilizadas influenciam significativamente a qualidade dos artefatos gerados por LLMs.

Keywords: Engenharia de Requisitos, Elicitação de requisitos, Documentação de requisitos, Large Language Models, Engenharia de Prompt.

1 Introdução

A definição de requisitos é uma etapa crucial, responsável por compreender e registrar as necessidades dos *stakeholders* a partir da coleta e análise de informações do contexto do sistema. Sua importância é crítica, pois requisitos mal definidos podem comprometer todo o ciclo de desenvolvimento, resultando em retrabalho, aumento de custos e insucesso em atender às necessidades dos *stakeholders* [5].

Com os avanços recentes da Inteligência Artificial Generativa, os *Large Language Model*s (LLMs) têm se destacado pela capacidade de compreender, gerar e transformar textos de forma sofisticada [12]. Diante dessas capacidades, observa-se um crescente interesse no uso de LLMs como ferramenta de apoio às atividades da Engenharia de Requisitos. Estudos recentes têm explorado LLMs nas mais diversas tarefas associadas a Engenharia de Requisitos (ER) e demonstrado potencial para automatizar tarefas repetitivas e organizar informações [6].

Proceedings of the 29th Workshop on Requirements Engineering (WER2026),
August 11-13, 2026, La Plata, Argentina.
https://doi.org/10.29327/1884517.29-19

Porém, apesar do seu potencial, a literatura também aponta que o uso de LLMs na Engenharia de Requisitos ainda apresenta desafios relevantes, especialmente no que se refere à qualidade, consistência e confiabilidade dos artefatos gerados. Estudos indicam que requisitos produzidos por LLMs podem apresentar ambiguidades, incompletude, inconsistências semânticas e perda de informações relevantes, exigindo validação humana [1][2].

A literatura sugere que parte dessas limitações das LLMs está diretamente ligada à forma como os *prompts* são estruturados e utilizados. A ausência do uso de padrões sistemáticos da Engenharia de Prompts faz com que os artefatos gerados apresentem alta variabilidade, baixa reprodutibilidade e inconsistência. Diante disso, considera-se que a Engenharia de Prompt é um elemento importante para auxiliar a utilização das LLMs, fazendo com que os artefatos gerados sejam mais controlados e previsíveis [3].

Diante desse cenário, este trabalho tem como objetivo avaliar o impacto da Engenharia de Prompt na elicitação e escrita de requisitos utilizando LLMs. Para isso, definimos e comparamos duas abordagens de elicitação e duas abordagens de escrita. Quanto à elicitação, definimos *prompts* para as técnicas de entrevista e a técnica dos seis chapéus do pensamento. Quanto à escrita de requisitos, definimos um *prompt* estruturado, com técnicas de output customization, e um *prompt* não estruturado. Esses *prompts* são encadeados em pipelines cujos resultados são então comparados qualitativamente e por meio dos critérios de qualidade de requisitos: Correção, completude, consistência, não ambiguidade, verificabilidade, rastreabilidade, modularidade, modificabilidade, clareza e compreensibilidade.

O restante deste artigo está organizado da seguinte maneira: A Seção 2 apresenta os conceitos e técnicas utilizados neste trabalho, bem como os trabalhos relacionados. A Seção 3 descreve o método para criação e aplicação de *prompts* a serem utilizados para gerar requisitos de software. A Seção 4 apresenta os resultados obtidos com a aplicação dos *prompts* e a avaliação dos resultados gerados por LLM. A Seção 5 discute os pontos positivos e negativos da abordagem proposta. Por fim, a Seção 6 resume as contribuições e trabalhos futuros.

2 Background

A Engenharia de Requisitos (ER) é a fase do desenvolvimento de software responsável por identificar, analisar, documentar e validar as necessidades dos stakeholders [5]. Dentre suas atividades, destaca-se a elicitação de requisitos, responsável por levantar e compreender as necessidades, restrições e informações do domínio dos *stakeholders* em relação a um software em desenvolvimento [5]. Essa atividade é realizada por meio de técnicas variadas como entrevistas, questionários, *workshops* e análise de documentos [5]. Embora eficazes, essas abordagens são altamente dependentes das habilidades dos engenheiros de requisitos, da disponibilidade de fontes de informação e da colaboração dos *stakeholders*. Nesse contexto, analisar a qualidade dos requisitos se torna um fator essencial. De acordo com as recomendações da IEEE 830, os requisitos com qualidade devem possuir características como correção, completude, consistência, não ambiguidade, verificabilidade e rastreabilidade [4].

Entre as diferentes abordagens utilizadas para explorar as necessidades dos *stakeholders*, a entrevista se destaca por possibilitar um diálogo direcionado entre o engenheiro de requisitos e os participantes envolvidos no sistema. Por meio dessa interação, é possível aprofundar a compreensão do domínio do problema, esclarecer dúvidas e explorar percepções, expectativas e restrições relacionadas ao software [4]. Além de métodos tradicionais de interação, técnicas que

estruturam o processo de reflexão também podem contribuir para ampliar a exploração das ideias durante a elicitação. Nesse contexto, destaca-se a técnica dos Seis Chapéus do Pensamento, proposta por Edward de Bono, que organiza a análise de um problema em diferentes perspectivas cognitivas representadas por chapéus. Cada chapéu corresponde a um modo específico de pensamento, como análise de fatos e informações, avaliação crítica, percepção emocional, identificação de benefícios e geração criativa de ideias [14]. Ao estimular a análise estruturada de um problema sob múltiplos pontos de vista, essa técnica favorece a exploração mais ampla de possíveis necessidades, problemas e funcionalidades relacionadas ao sistema.

LLMs são modelos de Inteligência Artificial treinados em grandes volumes de dados textuais, capazes de gerar, compreender e transformar textos, podendo ser aplicados em diversas tarefas de processamento de linguagem natural [12].

O uso de LLMs tem se destacado como uma abordagem promissora para apoiar atividades da Engenharia de Requisitos, trazendo benefícios relevantes em diversas etapas do processo de ER. No contexto da elicitação, esses modelos podem auxiliar na identificação de necessidades dos *stakeholders* [2], na geração de perguntas e na extração de requisitos a partir de descrições textuais, demonstrando capacidade de lidar com informações implícitas e pouco estruturadas [6]. Outro ponto positivo é a automação de tarefas, apoio ao refinamento de requisitos, estruturação e documentação, contribuindo para maior qualidade, completude e eficiência no processo de ER [8].

Entretanto, apesar desses avanços, a literatura também evidencia limitações importantes que impactam a aplicação das LLMs na ER. Entre os principais desafios, destaca-se a ambiguidade nas respostas geradas, uma vez que os modelos podem produzir diferentes interpretações para a mesma entrada [1]. Problemas de inconsistências em interações prolongadas ou em tarefas que exigem coerência entre múltiplos requisitos [2]. Além disso, as respostas podem apresentar superficialidade, não capturando adequadamente a complexidade do domínio ou os detalhes necessários para especificações de alta qualidade [8].

Diante dessas limitações, a engenharia de *prompt* se torna uma abordagem fundamental para melhorar a qualidade das respostas geradas pelas LLMs. A engenharia de *prompt* consiste na definição estruturada de instruções fornecidas ao modelo, com objetivo de orientar o comportamento de LLMs e influenciar diretamente o resultado produzido [3]. Estudos indicam que a forma como o *prompt* é elaborado impacta em aspectos como clareza, precisão e completude das respostas [10]. Além disso, diretrizes específicas para construção dos *prompts*, como uso de contexto, exemplos e restrições explícitas, se mostraram eficazes na melhora da qualidade das respostas [11].

Nesse contexto, uma das estratégias utilizadas na engenharia de *prompt* é o uso de padrões de *prompt*, que consistem em estruturas reutilizáveis de instruções projetadas para orientar o comportamento de modelos de linguagem durante a geração de respostas. Esses padrões ajudam a organizar a forma como o contexto, as instruções e os exemplos são apresentados. Entre esses padrões, destacam-se aqueles classificados como Output Customization, cujo objetivo é personalizar o formato ou estilo das respostas geradas. Dentre essa categoria, encontram-se o Persona Pattern, que define um papel ou perfil específico que o modelo deve assumir durante a interação e o Template Pattern, que estabelece uma estrutura predefinida para organizar a saída gerada pelo modelo[3].

Trabalhos recentes têm investigado o uso de LLMs como apoio às atividades da Engenharia de Requisitos, especialmente na etapa de elicitação. O estudo de Ronanki, Berger e Horkoff [7]

analisa o potencial do ChatGPT para auxiliar no processo de elicitação, destacando que os modelos possuem boa capacidade para gerar informações relevantes a apoiar a exploração inicial de requisitos, porém ainda apresentam limitações relacionadas à profundidade e precisão das respostas. De forma complementar, os trabalhos de Portugal et al. [1][2] investigam o uso de LLMs tanto no apoio à elicitação de conhecimento para a construção de catálogos de requisitos não funcionais quanto na geração automática de requisitos a partir de entrevistas de elicitação, evidenciando que esses modelos podem auxiliar na transformação de informações textuais em especificações de requisitos. Outros estudos concentram-se especificamente no papel da engenharia de *prompt* para melhorar o desempenho dos modelos. Ren, Nakagawa e Tsuchiya [16] demonstram que o uso de exemplos nos *prompts* pode melhorar a qualidade dos requisitos gerados pelos modelos. De maneira semelhante, Ronanki, Arvidsson e Axell. [11] apresentam diretrizes para estruturar *prompts* de forma a orientar melhor o comportamento dos modelos em tarefas da Engenharia de Requisitos. Além disso, Razinskas [20] investiga o uso de engenharia de *prompt* para facilitar a elicitação de requisitos a partir de múltiplos formatos de entrada, evidenciando que a forma como o *prompt* é estruturado influencia diretamente na qualidade das respostas.

Diante desse cenário, observa-se que a maioria dos trabalhos investiga o uso de LLMs ou da engenharia de *prompts* de forma isolada, sem considerar, de maneira integrada, a combinação de diferentes técnicas de elicitação e estratégias de documentação de requisitos.

Nesse contexto, este trabalho propõe investigar o impacto da engenharia de *prompt* na qualidade dos requisitos gerados por LLMs. Especificamente, busca comparar o uso de *prompts* estruturados e não estruturados, avaliando seus efeitos com base nos critérios de qualidade de requisitos, como correção, completude, consistência, não ambiguidade, verificabilidade, rastreabilidade, modularidade, modificabilidade, clareza e compreensibilidade. Assim, este estudo contribui para a literatura ao oferecer uma análise comparativa e qualitativa do uso de LLMs em conjunto com engenharia de prompt na elicitação de requisitos.

3 Método

A abordagem utilizada neste trabalho envolveu a utilização de LLM em conjunto com a engenharia de *prompt* para apoiar a elicitação de requisitos. Neste trabalho, todos os testes e execuções usaram o LLM ChatGPT, versão 5.2. Inicialmente, *prompts* de trabalhos relacionados [7] [9] [11] foram identificados e experimentados para identificarmos padrões e melhorias. Com isso percebe-se a necessidade de separar e estruturar a definição de requisitos em um pipeline que inclui 3 *prompts* com objetivos complementares: Contextualização, Elicitação e Escrita de Requisitos.

As primeiras versões dos *prompts* foram desenvolvidas a partir de padrões de *prompts* consolidados na literatura [3]. Em seguida, foram realizadas diversas rodadas de experimentação e refinamento, nas quais as estruturas dos *prompts* foram ajustadas com base nas experiências dos usuários durante a execução e na análise dos artefatos gerados por LLM. Esse processo iterativo resultou em 1 *prompt* para contextualização, 2 *prompts* para elicitação e 2 *prompts* para escrita de requisitos. Os *prompts* de elicitação definem o uso de duas técnicas: entrevista e seis chapéus do pensamento. Os *prompts* de escrita definem uma abordagem estruturada seguindo um template com campos bem definidos e uma abordagem não estruturada.

Posteriormente, esses 5 *prompts* foram avaliados considerando 4 cenários distintos: i) Contextualização+Entrevista+Escrita Estruturada; ii) Contextualização+Entrevista+Escrita Não-Estruturada; iii) Contextualização+6Chapéus+Escrita Estruturada; e iv) Contextualização+6Chapéus+Escrita Não-Estruturada. Por fim, os requisitos gerados em cada cenário foram avaliados individualmente e em conjunto, com base nos seguintes critérios de qualidade: correção, completude, consistência, não ambiguidade, verificabilidade, rastreabilidade, modularidade, modificabilidade, clareza e compreensibilidade.

A avaliação foi realizada por 7 pessoas que analisaram, cada uma, dois cenários, contendo uma lista de requisitos estruturados e outra de requisitos não estruturados. Para organizar a distribuição dos cenários, foram elaborados quatro formulários distintos, contendo os resultados gerados por LLM. Além disso, ao final de cada formulário há 3 questões subjetivas para o participante relatar os pontos positivos e negativos de cada cenário analisado, qual dos dois conjuntos de requisitos ele teve alguma preferência e sugestões e críticas sobre a pesquisa.

Os *prompts* desenvolvidos neste trabalho compartilham dois padrões estruturais base: o padrão Persona, que orienta o modelo a assumir papéis específicos em cada etapa, e o padrão Template, que define campos padronizados para as saídas geradas. A exceção é o *prompt* de documentação não estruturada (Fig. 5), que não utiliza nenhum padrão, funcionando como baseline comparativo. As particularidades de cada *prompt* são descritas a seguir.

3.1 Prompt de Contextualização

A Fig. 1 apresenta o prompt de contextualização, responsável por estabelecer a base para as etapas subsequentes. O modelo assume o papel de Engenheiro de Requisitos Sênior e recebe campos parametrizáveis de escopo, objetivo e *stakeholders*.

> Assuma a persona de um Engenheiro de Requisitos Sênior altamente experiente. Sua missão é me auxiliar a elicitar e documentar requisitos para um novo projeto de software. Eu fornecerei os detalhes do projeto e, em seguida, escolheremos uma técnica de elicitação para gerar os requisitos. Analise e absorva completamente o seguinte contexto do projeto:
> Escopo do problema/projeto:**{ESCOPO_DO_SISTEMA}**
> Objetivo principal do projeto:**{OBJETIVO}**
> Stakeholder envolvidos e suas prioridades:**{STAKEHOLDERS}**
> Sua tarefa nesta etapa é apenas confirmar se você entendeu o contexto e se está pronto para prosseguir para a fase de elicitação. Se sim, responda apenas com: "Contexto absorvido. Estou pronto para iniciar a elicitação. Qual técnica utilizaremos?"

Fig. 1. Prompt do contexto do sistema

3.2 Prompt de Entrevista

A Fig. 2 apresenta o *prompt* de entrevista, cujo diferencial é a adoção de múltiplas personas: o modelo assume simultaneamente os papéis de entrevistador e dos *stakeholders*, simulando uma coleta de informações a partir de suas perspectivas. O número de perguntas é configurável via parâmetro {N}

Vamos iniciar o levantamento de requisitos utilizando entrevistas estruturadas como técnica de elicitação.

O papel: Você assumirá, simultaneamente, as personas dos stakeholders que já foram definidos no nosso projeto e a persona entrevistador. Se convier, alguns stakeholders podem participar da mesma entrevista.

A dinâmica: O entrevistador irá formular perguntas estratégicas para descobrir as necessidades do sistema buscando complementar ou detalhar informações já coletadas. Já os stakeholders irão responder de acordo com suas perspectivas, expectativas e necessidades.

O formato: Para cada entrevista com no mínimo **{N}** perguntas, a saída será: “Pergunta” : [Nome do Stakeholder ou Papel]: “Resposta da pergunta”.

Por favor, comece a simulação, listando primeiro os stakeholders (ou grupos de stakeholders) que participarão e, em seguida, prossiga com a entrevista e armazene as respostas para que possam ser acessados posteriormente.

Fig. 2. Prompt de técnica de entrevista

3.3 Prompt da Técnica dos Seis Chapéus do Pensamento

A Fig. 3 apresenta o *prompt* baseado na técnica dos Seis Chapéus do Pensamento. Aqui o modelo também assume múltiplas personas sendo o facilitador e todos os *stakeholders*, conduzindo rodadas temáticas correspondentes a cada chapéu, com pelo menos {N} ideias por participante. O objetivo é simular uma dinâmica de discussão baseada na técnica dos seis chapéus na qual diferentes perspectivas são exploradas para apoiar a elicitação de requisitos. Neste *prompt* a geração de ideias ocorre em rodadas, nas quais todos os *stakeholders* analisam o problema utilizando um chapéu específico da técnica: fatos e dados, benefícios, precauções, criatividade, sentimentos e processo.Ao final, produz um resumo destacando pontos de concordância e conflitos entre as perspectivas.

Vamos conduzir um exercício criativo e produtivo. Sua tarefa agora é simular uma sessão de elicitação de requisitos utilizando a técnica dos 6 chapéus do pensamento de Edward de Bono.

O papel: Você assumirá, simultaneamente, as personas de todos os stakeholders que já definimos no nosso projeto. Você será tanto o facilitador quanto todos os participantes da reunião.

A dinâmica: O objetivo é gerar o máximo de requisitos possíveis utilizando as perspectivas dos chapéus e dos stakeholders, sem filtro inicial. Em cada rodada um dos chapéus será usado por todos os stakeholders, até que todos os chapéus sejam usados.

O formato: A cada rodada a resposta será: quem "falou": [Nome do Stakeholder ou Papel]: "Ideia" Para cada stakeholder ou papel, liste pelo menos **{N}** ideias.

Por favor, comece a simulação, listando primeiro os stakeholders que participarão e, em seguida, prossiga com as rodadas. Ao final de todas as rodadas faça um apanhado geral do que foi levantado destacando os pontos de acordo e de conflito e então armazene as respostas para que possam ser acessados posteriormente.

Fig. 3. Prompt da Técnica dos Seis Chapéus do Pensamento

3.4 Prompt para Escrita Estruturada de Requisitos

A Fig. 4 apresenta o *prompt* para escrita de requisitos estruturados, funcionais e não funcionais. Diferentemente dos demais, não utiliza o padrão Persona e adota a abordagem Zero-shot, instruindo o modelo a gerar os requisitos diretamente a partir do contexto e das respostas

geradas anteriormente. O Template contém campos definidos para nome, descrição, entradas, saídas, restrições, dependências, critérios de aceitação e prioridade.

Com base nos dados absorvidos anteriormente, gere os requisitos para o meu sistema seguindo exatamente os templates abaixo. Não adicione conteúdo fora dos templates.

=== REQUISITOS FUNCIONAIS (RF) ===

PARA CADA REQUISITO, SIGA EXATAMENTE O FORMATO ABAIXO:
RF-{NÚMERO}
Nome do Requisito: {NOME_DO_REQUISITO}
Descrição:
{DESCRICAO_DETALHADA_DO_REQUISITO}
Entradas:
- {DADO_DE_ENTRADA_1}
- {DADO_DE_ENTRADA_N}
Saídas:
- {SAIDA_1}
- {SAIDA_N}
Comportamento do Sistema:
{DESCRICAO_DO_COMPORTAMENTO_ESPERADO_CONFORME_IEEE_830}
Restrições:
- {RESTRICAO_1}
- {RESTRICAO_N}
Dependências:
- {RF_RELACIONADO_1}
- {RF_RELACIONADO_N}
Critérios de Aceitação:
- {CRITERIO_1}
- {CRITERIO_N}
Prioridade: {ALTA | MÉDIA | BAIXA}

=== REQUISITOS NÃO FUNCIONAIS (RNF) ===
PARA CADA REQUISITO, SIGA EXATAMENTE O FORMATO ABAIXO:
RNF-{NÚMERO}
Categoria: {DESEMPENHO, USABILIDADE, SEGURANÇA, CONFIABILIDADE, MANUTENIBILIDADE, PORTABILIDADE, COMPATIBILIDADE, DENTRE OUTROS}
Nome do Requisito: {NOME_DO_RNF}
Descrição: {DESCRICAO_CLARA_E_ESPECIFICA_CONFORME_IEEE_830}
Critérios de Aceitação:
- {CRITERIO_1}
- {CRITERIO_N}
Restrições Relacionadas:
- {RESTRICAO_1}
- {RESTRICAO_N}
Dependências:
- {RF_OU_RNF_RELACIONADO_1}
- {RF_OU_RNF_RELACIONADO_N}
Prioridade: {ALTA | MÉDIA | BAIXA}

Fig. 4. Prompt de documentação de requisitos estruturado

3.5 Prompt (não estruturado) para Escrita Estruturada de Requisitos

A Fig. 5 apresenta o *prompt* não estruturado definido para a documentação de requisitos. Nesta etapa, o LLM é instruído apenas a gerar requisitos funcionais e não funcionais, sem a utilização de padrões de *prompts*.

Com base nos dados absorvidos anteriormente, gere requisitos funcionais e requisitos não funcionais. Leve em consideração os critérios de qualidade de documentação expressos na norma IEEE.

Fig. 5. Prompt de documentação de requisitos não estruturado

4 Resultado

Nesta seção são apresentados os resultados da análise dos requisitos gerados nos 4 cenários Contextualização-Elicitação-Escrita, definidos na Seção 3. Em todos os cenários o *prompt* de Contextualização recebeu os mesmos dados referentes a um sistema para registro e acompanhamento de ocorrências em condomínios: *"Sistema de Registro de Ocorrências em Condomínios - O sistema tem como objetivo registrar e acompanhar ocorrências em condomínios, como problemas de manutenção ou reclamações. O escopo inclui criação de ocorrências, categorização, acompanhamento de status e comentários. Não inclui integração com prestadores de serviço externos. Os stakeholders são moradores, síndicos e administradores. O sistema deve assegurar transparência, controle de acesso por perfil e histórico completo das ocorrências, além de garantir confiabilidade e disponibilidade para acesso contínuo pelos usuários."*.

Ao todo, foram gerados 101 requisitos, sendo 57 requisitos funcionais e 44 requisitos não funcionais. Desses, 20 requisitos funcionais e 15 não funcionais foram produzidos a partir de *prompts* estruturados, enquanto os outros 37 requisitos funcionais e 29 não funcionais foram gerados por meio de *prompts* não estruturados. Dessa forma, o *prompt* não estruturado gerou um número maior de requisitos. Considerando os limites de tokens gerados por LLMs, a abordagem estruturada consome uma maior quantidade de tokens por requisito, então reduz a quantidade total de requisitos produzidos em uma única execução, enquanto o formato não estruturado permite descrições mais diretas e maior volume de itens.

A Tabela 1 resume as quantidades de requisitos escritos em cada cenário. Em todos os cenários o LLM considerou os 3 *stakeholders* na elicitação, a saber: morador, síndico e administrador. A Tabela 1 também apresenta quais participantes avaliaram os requisitos gerados em cada cenário, nomeados P1 a P7.

A análise foi feita de duas formas complementares: individual e global, sendo as avaliações dos critérios de qualidade realizadas por meio de uma escala de 4 pontos, onde 0 representa o pior caso e 3 representa o melhor caso. Na análise individual, cada requisito foi avaliado de forma isolada, permitindo identificar problemas específicos quanto aos critérios de correção, ambiguidade, verificabilidade, clareza, compreensibilidade e modularidade.

Tabela 1. Resumo quantitativo por cenário

Cenários	Avaliadores	RFs	RNFs

1) Entrevista + EscritaEstruturada	P1, P2, P5	10	8
2) Entrevista + EscritaNãoEstruturada	P1, P2, P6, P7	22	15
3) 6Chapéus + EscritaEstruturada	P3, P4, P6, P7	10	7
4) 6Chapéus + EscritaNãoEstruturada	P3, P4, P5	15	14

Na análise global, a avaliação considerou o conjunto de requisitos como um todo, analisando a coerência e a integridade do conjunto gerado, considerando os critérios de completude, consistência, rastreabilidade e modificabilidade.

Dessa forma, a abordagem adotada possibilita uma análise mais abrangente, contemplando tanto a qualidade pontual de cada requisito quanto à consistência e organização do conjunto completo de requisitos.

4.1 Análise Individual

A Tabela 2 apresenta as médias calculadas para cada cenário, considerando as pontuações atribuídas pelos avaliadores para cada requisito e critério, e o número de participantes que avaliaram aquele cenário.

Tabela 2. Avaliação individual requisitos funcionais

Cenários	Correção	Ambiguidade	Verificabilidade	Clareza	Compreensibilidade	Modularidade
1) Entrevista+EscritaEstruturada	1,90	1,77	1,96	2,24	2,24	2,20
2) Entrevista+EscritaNãoEstrutr.	2,80	2,21	2,46	2,50	2,60	2,47
3) 6Chapéus+EscritaEstruturada	2,96	2,28	2,40	2,51	2,82	2,59
4) 6Chapéus+EscritaNãoEstrutur.	2,75	2,28	2,57	2,74	2,83	2,71
Total geral	**2,64**	**2,15**	**2,36**	**2,50**	**2,64**	**2,50**

A análise individual dos requisitos evidencia diferenças entre os cenários avaliados, especialmente no que se refere à influência da técnica de elicitação e da estruturação dos *prompts*. De modo geral, observa-se que o cenário 4 apresentou o melhor desempenho global na maioria dos critérios, com destaque para verificabilidade, clareza e modularidade. Esse resultado sugere que a combinação entre uma técnica de elicitação baseada em múltiplas perspectivas e maior flexibilidade na escrita favorece a geração de requisitos mais claros e compreensíveis.

O cenário 3 também apresentou desempenho elevado, tendo o melhor nível em correção e mantendo resultados consistentes nos demais critérios. Esse comportamento mostra que a estruturação da escrita contribui especialmente para a precisão e consistência dos requisitos, embora, em alguns aspectos, possa reduzir a flexibilidade das descrições quando comparada à escrita não estruturada.

Por outro lado, o cenário 1 apresentou os menores valores na maioria dos critérios, como correção, ambiguidade e verificabilidade. Esse resultado indica que, embora a estruturação contribua para a organização, a técnica de entrevista pode limitar a exploração do domínio, resultando em requisitos com menor qualidade geral.

Nos cenários com escrita não estruturada, observa-se um comportamento distinto. O cenário 2 apresentou melhorias significativas em relação ao cenário 1, especialmente em correção e compreensibilidade, sugerindo que a flexibilidade na escrita favorece descrições mais naturais e de mais fácil entendimento. No entanto, seus resultados ainda permanecem inferiores aos cenários que utilizam a técnica dos seis chapéus.

Tabela 3. Avaliação individual por tipo de requisito

Cenário	*RF ou RNF*	Correção	Ambiguidade	Verificabilidade	Clareza	Compreensibilidade	Modularidade
Cenário1	RF	1,93	1,87	1,67	2,10	1,79	2,33
	RNF	1,88	1,67	2,10	1,79	2,33	2,13
Cenário2	RF	2,85	2,26	2,56	2,59	2,72	2,46
	RNF	2,72	2,13	2,32	2,38	2,45	2,48
Cenário3	RF	3,00	2,28	2,38	2,48	2,83	2,58
	RNF	2,89	2,29	2,43	2,57	2,82	2,61
Cenário4	RF	2,80	2,20	2,56	2,73	2,89	2,67
	RNF	2,71	2,38	2,60	2,76	2,79	2,76

A separação entre requisitos funcionais (RF) e não funcionais (RNF), apresentada na Tabela 3, revela que o padrão de desempenho observado na Tabela 2 se mantém de forma consistente em ambas as categorias, sem inversões expressivas entre os cenários. De modo geral, os RF e RNF apresentaram pontuações bastante semelhantes entre si dentro de cada cenário, com diferenças marginais na maioria dos critérios avaliados. Esse comportamento pode ser explicado pelo fato de que a geração dos requisitos foi mediada por uma LLM, que tende a aplicar o mesmo nível de cuidado linguístico e estrutural independentemente do tipo de requisito produzido. Assim, o fator que mais influenciou os scores não foi o tipo de requisito em si, mas sim a técnica de elicitação e o estilo de escrita empregados em cada cenário.

4.2 Análise Global

A Tabela 4 resume os scores obtidos por cenário considerando o mesmo procedimento de cálculo de médias descrito anteriormente na análise individual, considerando a agregação das pontuações por critério em cada cenário.

A análise global mostra que o cenário 3 apresentou um desempenho ligeiramente superior aos demais. Ainda assim, as diferenças entre os cenários foram pequenas, o que sugere que nenhuma combinação se sobressaiu de forma expressiva. Esses resultados indicam que a combinação entre exploração sistemática de múltiplas perspectivas e padronização na documentação tende a favorecer, ainda que de maneira modesta, a produção de conjuntos de

requisitos mais coesos, organizados e fáceis de evoluir.

Tabela 4. Avaliação global contexto

Cenário	Completude	Consistência	Rastreabilidade	Modificabilidade
1) Entrevista + EscritaEstruturada	1,67	2,00	2,00	2,00
2) Entrevista + EscritaNãoEstruturada	2,75	2,50	1,75	2,50
3) 6Chapéus + EscritaEstruturada	2,50	2,75	2,50	2,75
4) 6ChapéusEscrita + NãoEstruturada	2,67	2,33	2,33	2,67
Total geral	**2,43**	**2,43**	**2,14**	**2,50**

Em contraste, o cenário 1 apresentou desempenho inferior em completude, sugerindo que, embora a estrutura contribua para organização, a técnica de entrevista pode limitar a abrangência das informações coletadas.

O cenário 2 apresentou o maior valor em completude, indicando que a flexibilidade da escrita favorece a cobertura do domínio. No entanto, esse cenário apresentou o pior desempenho em rastreabilidade, evidenciando que a ausência de estrutura compromete a organização e a relação entre os requisitos. Esse resultado reforça a existência de um trade-off entre abrangência e organização.

Por fim, o cenário 4 apresentou resultados com desempenho equilibrado entre os critérios, mas inferior ao cenário estruturado equivalente. Isso indica que, embora a técnica dos seis chapéus contribua para a completude e diversidade das informações, a ausência de estrutura na documentação limita a qualidade global percebida do conjunto de requisitos.

4.3 Limitações e Ameaças

Apesar dos resultados promissores, o trabalho apresenta limitações que devem ser consideradas na interpretação dos achados. A restrição a um único modelo de LLM impede que os resultados sejam generalizados para outros modelos e outras versões, cujos comportamentos frente aos *prompts* avaliados podem diferir substancialmente. A avaliação qualitativa foi conduzida sobre um conjunto restrito de cenários, e um único contexto de domínio, o que limita a diversidade dos requisitos analisados e pode ter sido favorecido pelo contexto usado. Adicionalmente, cada cenário foi executado uma única vez, sem análise de estabilidade dos resultados, o que impede afirmar se os requisitos gerados são reprodutíveis ou se refletem variações de uma execução isolada. O número reduzido e a distribuição heterogênea de avaliadores introduzem assimetria na comparação entre grupos, efeito reforçado pela divergência de perfis observada entre os participantes. Enquanto alguns demonstraram uma postura avaliativa mais positiva, atribuindo pontuações mais altas, outros adotaram um perfil mais crítico e rigoroso, resultando em pontuações consideravelmente mais baixas, o que pode ter influenciado os resultados de forma desproporcional entre os cenários avaliados. Além disso, a quantidade total de avaliadores envolvidos no estudo é insuficiente para permitir a generalização dos resultados, uma vez que

um conjunto tão restrito de participantes não é representativo de uma população mais ampla de especialistas.

5 Discussão

Os resultados indicam que a engenharia de *prompt* e a técnica de elicitação impactam diretamente a qualidade dos requisitos gerados por LLMs. Esse efeito depende da combinação entre ambas as abordagens.

De forma geral, a técnica dos seis chapéus do pensamento apresentou desempenho superior à entrevista, especialmente nos critérios globais de consistência, rastreabilidade e modificabilidade, sugerindo que a exploração sistemática de múltiplas perspectivas favorece a geração de conjuntos de requisitos mais completos e coesos. No entanto, esse resultado não é isolado, sendo influenciado pela forma como os requisitos são documentados.

Em relação à documentação, *prompts* estruturados apresentaram melhor desempenho em critérios relacionados à organização e consistência, devido à padronização fornecida pelos templates. Por outro lado, essa abordagem pode introduzir limitações, como preenchimento superficial de campos e aumento da complexidade dos requisitos.

Já os *prompts* não estruturados favoreceram maior flexibilidade e cobertura do domínio, resultando em descrições mais naturais e, em alguns casos, mais informativas. Entretanto, essa abordagem apresentou limitações como menor padronização, ausência de elementos essenciais e maior dependência de contexto, dificultando a validação e o uso direto dos requisitos.

Por fim, os resultados evidenciam um trade-off entre padronização e flexibilidade, enquanto a escrita estruturada favorece a organização e o controle das informações, a escrita não estruturada amplia a cobertura do domínio, porém com menor controle e consistência.

6 Conclusão

Este trabalho analisou o impacto da engenharia de *prompt* na elicitação de requisitos com o uso de LLMs, a partir da definição e comparação de diferentes abordagens de elicitação e documentação organizadas em pipelines estruturados.

Os resultados obtidos demonstram que a qualidade dos requisitos gerados é influenciada pela combinação entre a técnica de elicitação e a estratégia de escrita adotada. De modo geral, observou-se que abordagens baseadas na técnica dos seis chapéus do pensamento apresentaram melhor desempenho global, especialmente nos critérios de completude, consistência, rastreabilidade e modificabilidade, evidenciando o impacto da exploração sistemática de múltiplas perspectivas. Em relação à estruturação dos *prompts*, os resultados indicam que sua contribuição está principalmente associada à melhoria de aspectos relacionados à organização e consistência, embora seu efeito não seja isolado, sendo dependente da técnica de elicitação utilizada. Além disso, foi identificado um trade-off entre padronização e flexibilidade, no qual a escrita estruturada favorece a organização e o controle das informações, enquanto a escrita não estruturada tende a ampliar a cobertura do domínio e a naturalidade das descrições.

Como principal contribuição, este estudo apresenta uma análise comparativa entre diferentes estratégias da engenharia de *prompt* aplicadas à Engenharia de Requisitos, evidenciando empiricamente como a estruturação dos *prompts* pode melhorar a qualidade dos

artefatos gerados por LLMs. Diferentemente de trabalhos existentes, que focam principalmente na aplicação isolada de LLMs, este trabalho propõe uma abordagem integrada baseada em pipelines de *prompts*, contemplando desde a contextualização até a documentação dos requisitos.

Apesar dos resultados promissores, o trabalho apresenta limitações, como o uso de um único modelo de LLM e a avaliação qualitativa realizada em um conjunto restrito de cenários. Nesse sentido, como trabalhos futuros, sugere a ampliação da avaliação para diferentes modelos de LLM, a inclusão de estudos quantitativos com métricas mais objetivas e a validação dos requisitos gerados com stakeholders reais. Além disso, pretende-se explorar outros cenários da Engenharia de Requisitos, como validação de requisitos e modelagem utilizando LLM.

Referências

1. Portugal, R.L.Q., Silva, L., Sousa, H., Leite, J.C.S.P.: Exploring LLMs on Supporting the Elicitation of Knowledge for NFR Catalogs. In: Workshop on Requirements Engineering (WER2025). (2025). Available at: https://werpapers.dimap.ufrn.br/papers/WER2025/wer202510.pdf.
2. Portugal, R.L.Q., Silva, L., Sousa, H., Leite, J.C.S.P.: From elicitation interviews to software requirements: evaluating LLM performance in requirement generation. In: Workshop on Requirements Engineering (WER 2025). (2025). Available at: https://werpapers.dimap.ufrn.br/papers/WER2025/wer202511.pdf (accessed: March 2026).
3. White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., Schmidt, D.: A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT. In: Proceedings of the 30th Conference on Pattern Languages of Programs (PLoP '23). ACM (2023). https://doi.org/10.5555/3721041.3721046 (accessed: March 2026).
4. IEEE: *IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998)*. IEEE, New York (1998). doi: 10.1109/IEEESTD.1998.88286.
5. Pohl, K., Rupp, C.: *Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam – Foundation Level – IREB Compliant*. 2nd edn. Rocky Nook, Santa Barbara, CA (2015).
6. Portugal, R.L.Q., Silva, L., Sousa, H., Leite, J.C.S.P.: Uso de Large Language Models na Engenharia de Requisitos. In: Workshop on Requirements Engineering (WER 2025). (2025). Available at: https://werpapers.dimap.ufrn.br/papers/WER2025/wer202517.pdf (accessed: March 2026).
7. Ronanki, K., Berger, C., Horkoff, J.: Investigating ChatGPT's Potential to Assist in Requirements Elicitation Processes. In: IEEE Conference Publication. IEEE (2023). Available at: https://ieeexplore.ieee.org/document/10371698 (accessed: March 2026).
8. Marques, N., Silva, R.R., Bernardino, J.: *Using ChatGPT in Software Requirements Engineering: A Comprehensive Review*. Future Internet 16(6), 180 (2024). doi: 10.3390/fi16060180. Available at: https://www.mdpi.com/1999-5903/16/6/180 (accessed: March 2026).
9. Mircea, M., Gevrek, E., Schmid, E., Schneider, K.: Supporting Stakeholder Requirements Expression with LLM Revisions: An Empirical Evaluation. arXiv preprint arXiv:2601.16699 (2026). Available at: https://arxiv.org/abs/2601.16699

10. Huang, K., Wang, F., Huang, Y., Arora, C.: Prompt Engineering for Requirements Engineering: A Literature Review and Roadmap. In: 2025 IEEE 33rd International Requirements Engineering Conference Workshops (REW), pp. 548–557. IEEE (2025).https://doi.org/10.1109/REW66121.2025.00081
11. Ronanki, K., Arvidsson, S., Axell, J.: Prompt Engineering Guidelines for Using Large Language Models in Requirements Engineering. arXiv preprint arXiv:2507.03405 (2025). Available at: https://arxiv.org/abs/2507.03405
12. Xiao, T., Zhu, J.: *Foundations of Large Language Models*. arXiv preprint arXiv:2501.09223 (2025). Available at: https://arxiv.org/abs/2501.09223 (accessed: March 2026).
13. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language Models are Few-Shot Learners. In: Advances in Neural Information Processing Systems (NeurIPS 2020), vol. 33, pp. 1877–1901 (2020). Available at: https://dl.acm.org/doi/abs/10.5555/3495724.3495883 (accessed: March 2026).
14. De Bono, E.: *Six Thinking Hats*. Little, Brown and Company, Boston (1986).
15. White, J., Fu, Q., Hays, S., Spencer-Smith, J., Schmidt, D.: ChatGPT Prompt Patterns for Improving Code Quality, Refactoring, Requirements Elicitation, and Software Design. arXiv preprint arXiv:2303.07839 (2023). Available at: https://arxiv.org/abs/2303.07839 (accessed: March 2026).
16. Ren, S., Nakagawa, H., Tsuchiya, T.: Combining Prompts with Examples to Enhance LLM-Based Requirement Elicitation. In: Proceedings of the 2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC 2024), pp. 1376–1381. IEEE (2024). https://doi.org/10.1109/COMPSAC61105.2024.00181
17. Görer, B., Aydemir, F.B.: Generating Requirements Elicitation Interview Scripts with Large Language Models. In: 2023 IEEE 31st International Requirements Engineering Conference Workshops (REW), pp. 44–51. IEEE (2023). https://doi.org/10.1109/REW57809.2023.00015
18. Görer, B., Aydemir, F.B.: GPT-Powered Elicitation Interview Script Generator for Requirements Engineering Training. In: 32nd IEEE International Requirements Engineering Conference (RE 2024), pp. 372–379. IEEE (2024). https://doi.org/10.1109/RE59067.2024.00042
19. Korn, A., Gorsch, S., Vogelsang, A.: LLMREI: Automating Requirements Elicitation Interviews with LLMs. In: IEEE International Requirements Engineering Conference (RE 2025). IEEE (2025). Available at: https://dblp.dagstuhl.de/rec/conf/re/KornGV25.html (accessed: March 2026).
20. Razinskas, M.: Prompt engineering for facilitating requirement elicitation from multi-format inputs. In: Joint Proceedings of the BIR 2025 Workshops and Doctoral Consortium, CEUR Workshop Proceedings, vol. 4034, pp. 236–245. CEUR-WS.org, Aachen (2025). Available at: https://ceur-ws.org/Vol-4034/paper67.pdf (accessed: March 2026).