# Story AI Generator: A ChatGPT-Based Tool for User Story Generation

Reine Santos
Institute of Computing
Federal University of Amazonas
Manaus, Amazonas, Brazil
rms@icomp.ufam.edu.br

Hamiê Tomás
Federal Institute of Education, Science and Technology of Amazonas
Manaus, Amazonas, Brazil
hamie.tomas@ifam.edu.br

Bruno Gadelha
Institute of Computing
Federal University of Amazonas
Manaus, Amazonas, Brazil
bruno@icomp.ufam.edu.br

Tayana Conte
Institute of Computing
Federal University of Amazonas
Manaus, Amazonas, Brazil
tayana@icomp.ufam.edu.br

Ana Carolina Oran
Institute of Computing
Federal University of Amazonas
Manaus, Amazonas, Brazil
ana.oran@icomp.ufam.edu.br

## Abstract

The use of Large Language Models (LLMs) has fostered the development of tools to support Requirements Engineering activities, particularly the automated generation of user stories from textual descriptions. However, further evidence is still needed regarding the practical acceptance and applicability of such tools in Software Engineering contexts.This paper presents Story AI Generator, a web-based tool built upon the US-Prompt approach to support the generation of structured user stories during early requirements elicitation and specification activities. Story AI Generator is based on a customized use of ChatGPT and facilitates the generation of user stories from product vision descriptions using predefined prompting strategies, while also supporting the organization and guidance of relevant input information during story creation. The tool aims to simplify the user story generation process and align generated stories with high-quality user story standards based on the QUS framework through a simple and responsive web interface. To evaluate the tool, we conducted an exploratory study with 11 Software Engineering professionals using TAM-based questionnaires and qualitative feedback analysis. The results indicated a positive acceptance of the tool, with at least 81.8% of participants strongly agreeing regarding the clarity of interaction, ease of use, low cognitive effort, and simplicity of the story generation process. Participants also perceived the tool as useful for improving productivity, supporting feature identification, reducing manual effort, and facilitating stakeholder communication during requirements activities. In addition, the generated user stories were considered clear, organized, and aligned with the provided context. These findings suggest that Story AI Generator is a promising tool for supporting early Requirements Engineering activities and provide empirical evidence regarding the acceptance of LLM-based approaches for user story generation.

**Demo Video: https://doi.org/10.5281/zenodo.20373672**

## Keywords

ChatGPT, LLM, Technology Acceptance, Requirements Engineering, User Stories, Prompt Engineering, Empirical Study

## 1 Introduction

Requirements Engineering (RE) is responsible for discovering, analyzing, documenting, and verifying the services and constraints of a system, representing a critical stage in software development to ensure that the final product meets users' needs [22]. In this context, requirements elicitation is widely recognized as one of the most challenging activities and one of the main causes of project failure, since it requires transforming often vague expectations into clear and measurable requirements through interaction with stakeholders [6, 23].

The increasing adoption of Large Language Models (LLMs) in software development tasks has significantly transformed the Requirements Engineering process, particularly in the creation of user stories [2, 26]. User stories are essential artifacts for capturing end-user needs and translating them into clear and objective functionalities for the development team [14–16]. However, the quality of automatically generated user stories and the acceptance of automated generation techniques by Software Engineering professionals still represent important challenges [11].

To support requirements engineers, product owners, and other professionals involved in requirements elicitation activities by providing the automated generation of higher-quality user stories, we present the Story AI Generator tool. Story AI Generator is an LLM-based tool resulting from a research process aimed at investigating ways to improve the effectiveness of specialized prompts for user story generation. This process followed the Design Science Research (DSR) method, and its evolution can be observed through two previously published studies [20, 21].

The tool aims to provide a practical and simplified experience for specifying user stories, accelerating the requirements specification phase. It was designed to generate user stories aligned with seven quality criteria from the Quality User Story (QUS) framework [11]: Well-Formed, Atomic, Minimal, Conceptually Sound, Unambiguous, Full Sentence, and Estimatable. To investigate users' perceptions and acceptance, an exploratory study was conducted with Software Engineering professionals experienced in user stories, whose expertise supported the assessment of generated stories and tool applicability. Participants used the Story AI Generator and assessed it using the TAM3 model and qualitative feedback on perceived benefits, limitations, and improvement opportunities.

# 2 Background and Related Work

## 2.1 User Stories

User stories are widely used artifacts for specifying requirements in agile software development [10]. They describe functionalities from the user's perspective, typically following the format proposed by Connextra [4]: "As a⟨user type⟩, I want⟨goal⟩, so that⟨some reason⟩." Since user stories communicate user needs and guide development activities [13], their quality is essential in RE.

Frameworks such as INVEST (Independent–Negotiable–Valuable–Estimatable–Scalable–Testable) [24] and the Quality User Story (QUS) framework [11] provide criteria for assessing user story quality. QUS defines 13 criteria related to syntactic, pragmatic, and semantic quality aspects. This study adopts seven criteria for analyzing individual user stories, excluding those requiring multiple stories, as presented in the Supplementary Material.[1]

## 2.2 Prompt Engineering Techniques

Several prompt engineering techniques have been proposed, including zero-shot, few-shot, and Chain-of-Thought prompting. Among them, Meta-Prompting stands out by using carefully structured prompts composed of explicit instructions, examples, and criteria that guide the model during task execution. This technique creates a structured reasoning flow, encouraging the model to follow predefined steps and generate outputs that are more refined, consistent, and aligned with the expected requirements [3, 27].

Another relevant approach is Few-Shot Prompting, which enables LLMs to generalize from a small set of examples, allowing the generation of appropriate responses without additional training [3, 27]. Both techniques were adopted in the tool and have shown potential for improving the quality and consistency of AI-generated artifacts, particularly in agile and RE contexts.

## 2.3 Related Work

Recent studies have explored the use of LLMs and intelligent approaches to support Requirements Engineering activities, including requirements elicitation, organization, refinement, and user story generation. Some works propose tools and frameworks such as RITA [12], which integrates LLM-based tasks for classifying user feedback, identifying non-functional requirements, and generating requirements specifications and user stories from online reviews; MARE [9], a collaborative multi-agent framework for elicitation, modeling, verification, and specification activities; GeneUS [15], a GPT-4-based tool that applies Chain-of-Thought and "Refine and Thought" strategies to generate and refine user stories; and Tool4IoTReq [19], a tool for classifying and grouping semantically similar requirements in IoT environments. Other studies focus specifically on user stories, such as the multi-agent system proposed by Sami et al. [18] for automated generation and prioritization, and ALAS [28], an autonomous LLM-based agent system for improving agile user story quality. Additionally, studies by Sakib et al. [17] and Hernández-Agüero et al. [8] demonstrated the potential of LLMs to transform app reviews into backlog-ready user stories and automatically evaluate and improve story quality using frameworks such as RUST and INVEST. Ayach and Fontao [1] also explored prompt engineering to support proto-persona creation during Lean Inception activities.

Although previous studies have proposed tools and intelligent approaches to support Requirements Engineering activities, the Story AI Generator differs by focusing on the practical generation of structured user stories through an accessible web-based environment using prompt engineering strategies. While GeneUS and multi-agent approaches emphasize automated pipelines and agent collaboration, the proposed tool combines prompt engineering techniques with a user-oriented interface to support early requirements elicitation and reduce manual effort during story creation. Unlike Tool4IoTReq and ALAS, which focus on requirements organization and story refinement, respectively, the Story AI Generator is centered on generating structured user stories from initial product descriptions. The proposed tool provides a practical and interactive environment for real software engineering activities. Unlike the direct use of general-purpose LLM interfaces such as ChatGPT, the Story AI Generator offers a structured and domain-oriented environment that applies specialized prompt engineering strategies to support the generation of quality user stories in RE contexts.

# 3 Tool Story AI Generator: Crafting User Stories

To support the requirements specification process, we developed the Story AI Generator, a tool designed to operationalize and support, in a guided and simplified manner, the automated generation of user stories without requiring advanced knowledge of prompt engineering. The Story AI Generator is an LLM-based tool distributed under the MIT license. This section presents its overall structure, objectives, target audience, architecture, and usage workflow in the context of automated user story generation.

## 3.1 Tool Description

The Story AI Generator: Crafting User Stories with LLMs is based on the User Story Prompt (US-Prompt) approach, introduced in previous studies [20, 21]. It constitutes the core mechanism for user story generation within the tool. It was designed to enable contextualized, consistent, and quality-driven generation of user stories, guided by the QUS framework. The tool encapsulates the US-Prompt so that end users do not need to understand its technical details, requiring only basic input information to obtain well-structured user stories.

The tool provides a practical and easy experience, accelerating the requirements specification phase. This enables teams to produce user stories in a more consistent, structured, and time-efficient manner, benefiting different roles within the development team.

*3.1.1 Tool Features.* The Story AI Generator provides a set of functionalities designed to support and simplify the creation of user stories. The main functionalities include:

- Automated generation of user stories based on predefined quality criteria, ensuring consistency and improving the specification process;
- Guided support for user story creation through an interface that reduces process complexity and minimizes the effort required from users;
- Structured data collection mechanisms that organize and assist the input process, helping users provide relevant and complete information;

[1]Supplementary Material available at: Zenodo Repository.

- Customizable input configuration, allowing users to define project-specific elements such as stakeholders, functionalities, and contextual information;
- Standardized formatting of generated user stories according to QUS criteria, promoting quality and uniformity in the resulting specifications.

*3.1.2 Tool Users.* The Story AI Generator was designed to serve different profiles of professionals involved in Requirements Engineering activities and in the process of defining software features throughout software development. Its main target audience includes requirements engineers, product owners, systems analysts, requirements analysts, product managers, and other professionals responsible for requirements elicitation and specification.

In addition to the professional context, the tool can also support researchers, students, and trainees who wish to understand or apply practices related to user story writing and requirements specification, serving as an educational and experimental support tool. It aims to support both experienced and novice users, promoting greater consistency and quality in user story creation.

## 3.2 Tool Architecture

The tool was developed in Python, using the Django framework for both backend and frontend. The backend is responsible for processing requests, managing data, and integrating with external APIs, such as the OpenAI API, through secure and authenticated HTTP calls.

When the user submits data through the input form, the backend processes the information and automatically constructs the prompt used for user story generation. The prompt is then sent to the ChatGPT API, enabling the model to generate content automatically without requiring users to manually design prompts or have advanced knowledge of prompt engineering (Figure 1) .

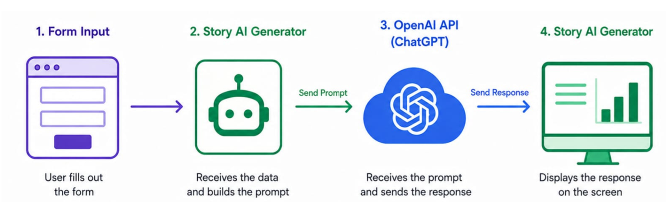


**Figure 1: Story AI Generator Operating Diagram.**

The frontend was built using Django's template system, providing a simple and responsive form for data input. It is complemented by HTML5, CSS3, and JavaScript to improve usability and deliver a more intuitive user experience.

Regarding deployment, the application was hosted on Amazon Web Services (AWS), making it accessible via the web and device through the URL www.storyaigenerator.com.br. For version control and development management, Git and GitHub were used, ensuring traceability of changes and efficient collaboration.

## 3.3 Tool Usage

The tool interface was designed to be simple, intuitive, and easy to navigate, allowing users with different profiles to use it without prior training.

The following images present the main frontend screens, illustrating the workflow from the initial data input to the display of the generated user stories. These images aim to demonstrate the visual organization, layout of elements, and clarity of input fields, contributing to a smooth and objective user experience.

Figure 2 shows the tool's main input form. The interface also provides a usage guide intended to support users during the user story generation process. It presents the main steps for filling out the form as well as simple examples of system descriptions. The web form is divided into the following sections:

- **Product Description**: brief description of the system.
- **User Information**:
  - **User (role or function)**: the role of the user (e.g., *Administrator*, *Regular user*).
  - **User Description**: brief description of the user profile.
  - **User Actions**: main actions the user intends to perform.
- **Additional Information**: system-specific details and supplementary information required for generation.

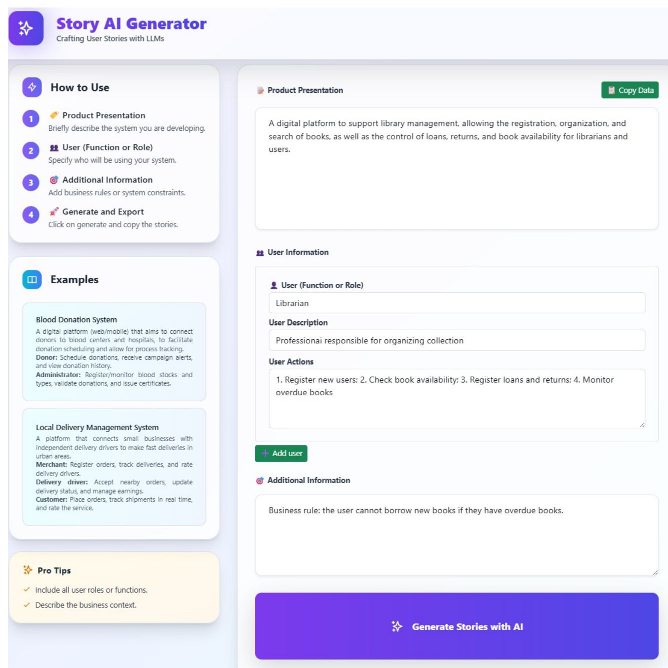


**Figure 2: Input form of the Story AI Generator.**

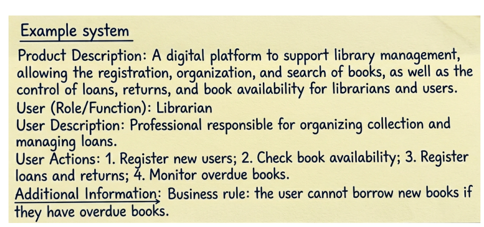
Example system
Product Description: A digital platform to support library management, allowing the registration, organization, and search of books, as well as the control of loans, returns, and book availability for librarians and users.
User (Role/Function): Librarian
User Description: Professional responsible for organizing collection and managing loans.
User Actions: 1. Register new users; 2. Check book availability; 3. Register loans and returns; 4. Monitor overdue books.
Additional Information: Business rule: the user cannot borrow new books if they have overdue books.

**Figure 3: Example of a Library Management System.**

In this example, we use a Library Management system as an example, as shown in Figure 3. After completion, the user clicks the *"Generate Stories with AI"* button, triggering the backend to process the input data and send a structured prompt to the OpenAI API.

*3.3.1 User Story Generation with ChatGPT.* Upon receiving the form data, the backend automatically consolidates the prompt and sends it to the ChatGPT API. The integration with the OpenAI API is performed through authenticated HTTP requests. The model returns set of user stories according to the expected criteria. This output is then directly rendered in the web interface, allowing users to review or export the generated user stories, shown in Figure 4.

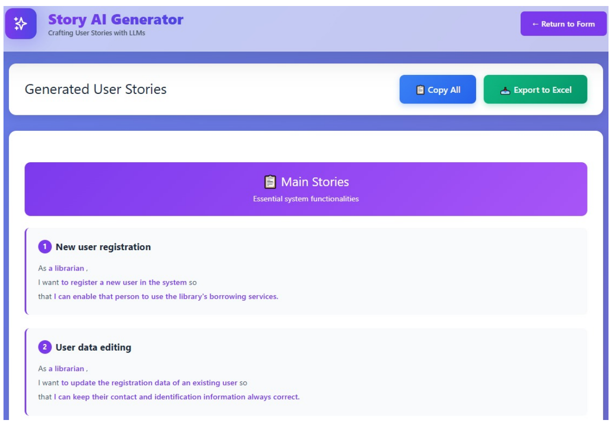


**Figure 4: Output interface with generated user stories.**

In this way, the Story AI Generator operationalizes the method proposed in this research. By encapsulating prompt engineering best practices combined with the capabilities of LLMs, the tool aims not only to facilitate the adoption of such approaches by practitioners, but also to support the generation of higher-quality requirements in a faster and less cognitively demanding manner.

# 4 Exploratory Study with SE Professionals

## 4.1 Study Methodology

The study aimed to answer the following research question: **RQ: "What perceptions, limitations, and opportunities for improvement are identified by Software Engineering professionals through the use of the Story AI Generator tool?"**. To address this question, an exploratory study was conducted following the methodological guidelines proposed by [25], involving Software Engineering professionals, in which participants used the tool and provided feedback regarding their user experience.

*4.1.1 Study Design.* The study was conducted following a structured protocol, according to the Software Engineering experimentation guidelines proposed by [25] in order to ensure the systematization of the activities and the reliability of the collected data, considering the following aspects:

- `Participants:` Participants were recruited in person during the Brazilian Symposium on Software Quality (SBQS), a national event that brings together Software Engineering professionals from academia and industry. The diversity of participants and the relevance of the event provided an appropriate environment for conducting the study and collecting qualified perceptions regarding the use of the Story AI Generator tool. Participants were selected based on prior experience with user stories in academic or professional contexts. In total, 11 participants voluntarily agreed to take part in the study and signed an informed consent form.
- `Instrumentation:` The study used four instruments:
  (i) An Informed Consent Form (ICF);
  (ii) The Story AI Generator tool, used for the automated generation of user stories [2];
  (iii) A document containing example scenarios, provided to participants during their interaction with the tool;
  (iv) A data collection questionnaire based on the TAM3 model, adapted to the context of the study;

  The questionnaire evaluated constructs related to technology acceptance using a five-point Likert scale and included open-ended questions to capture participants' perceptions and suggestions regarding the use of the tool. All the instrumentation was made available as supplementary material.

*4.1.2 Study Execution.* The study was conducted in person and followed the same standardized execution sequence for all participants in order to ensure procedural uniformity and reduce possible biases during data collection.

A pilot study was initially conducted with two participants experienced in user stories to evaluate the clarity of the instruments and the use of the Story AI Generator tool. This stage resulted in improvements to the tool interface and functionality, while no changes were required in the data collection instruments.

The experimental study was conducted individually with all participants following a standardized procedure. After receiving instructions and signing the informed consent form, participants used the Story AI Generator tool to create user stories based on predefined or self-defined scenarios. Subsequently, they completed the characterization questionnaire and the TAM3-based evaluation instrument, followed by a semi-structured interview conducted by the researcher. The entire procedure was carried out with each participant, with an average duration of 28 minutes and 49 seconds.

## 4.2 Study Results

The results were analyzed through a mixed-methods approach combining quantitative and qualitative analyses. Quantitative data from the TAM3-based questionnaire were analyzed using descriptive statistics to evaluate technology acceptance constructs, including perceived usefulness, perceived ease of use, and behavioral intention to use. Qualitative data from open-ended questions and semi-structured interviews were analyzed through thematic analysis to identify recurring perceptions, limitations, and improvement opportunities related to the use of the Story AI Generator tool. The qualitative analysis was conducted by the author and reviewed by two PhD researchers. In addition, the dataset supporting this study was made available as supplementary material, including generated user stories, questionnaire responses, and interview records.

**Section 1 - Participants' demographic profile**

[2]Available at: https://www.storyaigenerator.com.br/. Story AI Generator.

Participants presented a geographically diverse distribution, as shown in Table 1, suggesting that the collected perceptions were not restricted to a specific regional context. In addition, most participants (81.8%) reported using user stories in both academic and industrial environments.

**Table 1: Characterization of the participants.**

| ID | Location | Context | Frequency | Experience |
|---|---|---|---|---|
| P1 | São Borja, RS | Both | Monthly | More than 10 years |
| P2 | Porto Alegre, RS | Both | Weekly | More than 5 years |
| P3 | Campo Mourão, PR | Both | Sporadically | Less than 5 years |
| P4 | Rio de Janeiro, RJ | Both | Daily | Less than 5 years |
| P5 | Brasília, DF | Both | Weekly | More than 10 years |
| P6 | Morada Nova, CE | Academia | Sporadically | Less than 5 years |
| P7 | Fortaleza, CE | Both | Daily | More than 5 years |
| P8 | Manaus, AM | Industry | Sporadically | More than 20 years |
| P9 | Manaus, AM | Both | Weekly | Less than 5 years |
| P10 | Manaus, AM | Both | Monthly | More than 5 years |
| P11 | Manaus, AM | Both | Monthly | Less than 5 years |

*4.2.1 Questionnaire Results: Acceptance Evaluation.* The evaluation questionnaire collected data related to closed-ended questions based on the TAM3 model constructs and open-ended questions aimed at capturing perceptions of user experience and tool acceptance.

**Perceived Ease of Use:** The results related to Perceived Ease of Use indicate a predominantly positive evaluation of the tool. For all statements of this construct, at least 81.8% of participants strongly agreed regarding the clarity of interaction, low cognitive effort, overall ease of use, and simplicity of generating user stories. These results suggest that participants perceived the tool as intuitive and easy to use, which was reinforced by qualitative responses highlighting the organized interface and the rapid generation process.

**Perceived Usefulness:** The results related to Perceived Usefulness indicate that most participants agreed or strongly agreed that the tool positively contributes to their performance, productivity, effectiveness in generating user stories, and usefulness in their work activities. Qualitative responses reinforced this perception, highlighting the tool as useful for accelerating the creative process and supporting the specification of user stories.

**Behavioral Intention to Use:** Regarding Behavioral Intention to Use, 63.6% of participants strongly agreed with the statements related to future use of the tool and expressed interest in adopting it in educational, research, and software development contexts, especially to support the creation and communication of user stories.

**Perceived Relevance:** The results indicate a generally positive evaluation, although with a greater presence of neutral responses, particularly regarding the importance of the tool in participants' current work activities. Only 36.4% of participants strongly agreed that the tool is important in their work, suggesting that, although it is perceived as useful and easy to use, it is not yet considered essential in all professional contexts.

**Perceived Output Quality:** The results related to Perceived Output Quality indicate a positive evaluation of the user stories generated by the tool. Most participants agreed that the generated stories presented satisfactory quality and were aligned with the provided context, although some limitations requiring refinement were identified, particularly regarding non-functional requirements and story detailing. The only disagreement responses were observed for O2 and O3, related to the perceived quality of the generated outputs, reported by participant P6, who acknowledged the alignment of the generated stories with the provided context but suggested improvements in story refinement, particularly regarding similar stories and epics. The remaining responses were neutral, indicating satisfactory perceptions with opportunities for improvement.

*4.2.2 Interview Results: Participants' Perceptions.* The analysis of participants' perceptions was conducted based on the open-ended questionnaire responses and the semi-structured interviews performed at the end of the experiment. The data analysis followed procedures inspired by Grounded Theory (GT), a qualitative method based on systematic data collection and analysis procedures for understanding social phenomena [7].

The analysis process was conducted through two stages of coding: open coding and axial coding. Open coding involved the identification, comparison, and categorization of concepts emerging from the data, while axial coding was employed to examine relationships between categories and identify connections among the identified perceptions [5]. The main findings regarding participants' perceptions of the tool and their implications are presented below.

**Observed Benefits of Using the Tool** – Participants highlighted that the tool requires low cognitive effort, presents an intuitive interface, and enables the rapid generation of user stories. According to P1, *"generated an interesting number of user stories quickly and effortlessly"*. In addition, the tool contributes to improving performance by fostering the identification of new functionalities and accelerating manual and repetitive activities. Participants considered it an effective support for user story generation, according to P7: *"...fostering new functionalities and accelerating manual and repetitive activities"*. The generated stories were considered aligned with the provided context, as reported by P3: *"Without the tool, I would perform my tasks the same way, but with it, based on the sample generated, it was excellent to compare the results with what I would have produced myself..."*.

Regarding the interface structure, composed of fields oriented toward contextual description, P3 stated: *"...The structuring of the tool into fields (description, actors, and additional information) facilitates the process of describing the project context..."*. This approach was perceived as more intuitive than general-purpose AI tools, which are usually based on a single generic input field. Participants indicated that this structure, combined with the provided examples, supports context completion and reduces the need for prior training.

**Perceived Use of the Tool as Support** – Participants mainly perceived the tool as a starting point for the requirements specification activity. The tool was associated with initial support for both requirements analysts and product owners, assisting in problem understanding and in the initial structuring of user stories. P6 highlighted: *"...the tool is excellent as a starting point for the generation of user stories; therefore, both in academia and in industry, this initial kickstart is valuable"*. The use in the academic context was associated with testing and training activities. Participants indicated that the tool can be used by professors in practical exercises and in disciplines related to Requirements Engineering. P2, for example, stated that: *"I would love to use it with my students in the Requirements* *course and even to test my own knowledge"*. In the industrial context, participants emphasized the use of the tool as support for validation with stakeholders. P5 indicated that: *"...using the tool will support me in writing the Stories so that I can present them to the stakeholder early and obtain validation in order to proceed to the requirements specification with clearer rules"*, contributing to the definition of rules that are better understood before the formal specification.

**Additional Information, Acceptance Criteria, and NFRs** – This phenomenon highlights limitations perceived by participants, especially regarding the generation of acceptance criteria and the treatment of non-functional requirements (NFRs). Participants reported the absence of acceptance criteria in the generated user stories. This perception was pointed out by P6, who highlighted: *"...a non-functional requirement was not considered, nor were the acceptance criteria"*, leading him to associate this absence with the following conclusion: *"I noticed that the tool did not consider the entire context I described..."*. Similarly, P7 commented on the quality of the stories: *"...they could have been specified a little more; I provided some information that the user stories eventually did not comprehend"*, basing this conclusion on the fact that the tool abstracted unsupported information.

Regarding non-functional requirements, participants observed that the tool did not consider them as expected. P2 identified that: *"Two generated user stories represent non-functional requirements, so this is where it is necessary to think about how to handle requirements of this category"*. This was interpreted by the participant as an inadequacy. In general, although the tool is perceived as useful for the initial generation of user stories, there are limitations related to the incorporation of additional information, the representation of acceptance criteria, and the treatment of non-functional requirements, indicating opportunities for the evolution of the tool.

**Expectations for Tool Evolution** – Participants suggested improvements related to the interface, the prompt-based generation process, and the expansion of the generated elements, indicating expectations for greater control and completeness of the generated user stories. Participants suggested interface improvements, as P10: *"...is that the tool should make the description provided in the previous step available to users so that they can evaluate the generated stories in comparison with the provided text..."*. They also suggested the possibility of editing and refining stories, as mentioned by P2: *"But if I start describing the scenario that I know, then it could keep talking to me and generating more, right? Because everything came at once... I don't know if conceptually there is a difference, okay? But maybe it would give me the feeling that I am talking to someone and having ideas and debating, right?"*.

The generation of new elements also emerged as mentioned by P6: *"I am looking forward to seeing the version of the tool considering acceptance criteria and also non-functional requirements"*. The generation of documentation was also suggested by P5, who stated: *"...with the evolution of the tool, we may add more instructions about the requirements and the documentation could be generated automatically"*. At the same time, the need for improvements related to data protection and privacy was highlighted as a condition for the use of the tool in industry contexts, as suggested by P7: *"...perhaps using a private cloud...so that the data does not go to another instance and remains there within the industry..."*.

## 5 Study Limitations

Every research study has limitations regarding its results [25]. The current version of the Story AI Generator presents some limitations. Tool focuses on generating functional user stories from initial textual product descriptions and does not explicitly support acceptance criteria, non-functional requirements, or interface-related requirements. Since the generated outputs rely on LLM responses, the stories may still contain ambiguities, inconsistencies, or incomplete information, requiring human review and refinement. Additionally, the tool currently provides limited customization, user control, and iterative rewriting support during the generation process.

## 6 Conclusions and Future Work

This paper presented the Story AI Generator, a web-based tool for generating structured user stories through prompt engineering strategies to support Requirements Engineering activities. An exploratory study with Software Engineering practitioners evaluated the tool in early requirements elicitation activities. The results indicated positive acceptance regarding usability, interface clarity, and organization of generated user stories, suggesting potential to support initial elicitation and requirements structuring tasks.

The evaluation also revealed limitations and improvement opportunities, including the lack of explicit support for acceptance criteria and non-functional requirements, limited customization and user control, and limited support for iterative rewriting. A trade-off was observed between the creative capabilities enabled by LLMs and the level of control expected by users.

Overall, the findings reinforce the potential of LLM-based tools to support practical Requirements Engineering activities. Future work includes refining the product vision, validating user stories, generating acceptance criteria, deriving test cases from acceptance scenarios, generating and validating non-functional requirements, and supporting backlog generation and prioritization.

## Artifact Availability

All artifacts used and produced during the experiment, including generated user stories, TAM questionnaire responses, interview transcripts, and the source code of the Story AI Generator tool, are publicly available in the following repository: **Zenodo Repository**. And the Story AI Generator tool is available at: https://www.storyaigenerator.com.br.

## Acknowledgements

We thank all the participants in the empirical study and the Research Group members for their support. We would like to thank the financial support granted by CNPq - 313137/2025-0; CNPq 406506/2025-6; CNPq 445029/2024-2; CNPq - 314797/2023-8; CNPq - 443934/2023-1; CAPES- Financing Code 001; Amazonas State Research Support Foundation - FAPEAM - through POSGRAD 2026-2027. The authors acknowledge the use of OpenAI's ChatGPT (GPT-5 Thinking) for translation and linguistic refinement of the manuscript. The authors critically reviewed all AI-assisted revisions and take full responsibility for the final manuscript, in accordance with CNPq's Policy on Integrity in Scientific Activity (Ordinance No. 2.664/2026).

# References


[1] Fernando Ribeiro Ayach and Awdren Fontao. 2024. Towards the acceptance of a Prompt Engineering-based Approach to Build Proto-personas during Product Discovery. In *Congresso Brasileiro de Software: Teoria e Prática (CBSoft)*. SBC, 49–58.

[2] Lenz Belzner, Thomas Gabor, and Martin Wirsing. 2023. Large language model assisted software engineering: prospects, challenges, and a case study. In *International Conference on Bridging the Gap between AI and Reality*. Springer Nature Switzerland, Cham, 355–374. doi:doi/abs/10.1007/978-3-031-46002-9_23

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020). https://arxiv.org/pdf/2005.14165

[4] Mike Cohn. 2004. *User stories applied: For agile software development*. Addison-Wesley Professional.

[5] Juliet Corbin and Anselm Strauss. 2014. *Basics of qualitative research: Techniques and procedures for developing grounded theory*. Sage publications.

[6] Hugo Ferreira Martins, Antônio Carvalho de Oliveira Junior, Edna Dias Canedo, Ricardo Ajax Dias Kosloski, Roberto Ávila Paldês, and Edgard Costa Oliveira. 2019. Design Thinking: Challenges for Software Requirements Elicitation. *Information* 10, 12 (2019). doi:10.3390/info10120371

[7] Barney G Glaser, Anselm L Strauss, and Elizabeth Strutzel. 1968. The discovery of grounded theory; strategies for qualitative research. *Nursing research* 17, 4 (1968), 364.

[8] Erika Hernández-Agüero, Christian Quesada-López, and José P Chaves-Sánchez. 2026. LLM-Assisted INVEST Evaluation and Improvement of User Stories: An Industrial Replication Study. In *Congresso Ibero-Americano em Engenharia de Software (CIbSE)*. SBC, 61–75.

[9] Dongming Jin, Zhi Jin, Xiaohong Chen, and Chunhui Wang. 2024. Mare: Multi-agents collaboration framework for requirements engineering. *arXiv preprint arXiv:2405.03256* (2024).

[10] Garm Lucassen, Fabiano Dalpiaz, Jan Martijn EM Van Der Werf, and Sjaak Brinkkemper. 2015. Forging high-quality user stories: towards a discipline for agile requirements. In *2015 IEEE 23rd international requirements engineering conference (RE)*. IEEE, 126–135.

[11] Garm Lucassen, Fabiano Dalpiaz, Jan Martijn EM Van Der Werf, and Sjaak Brinkkemper. 2016. The Use and Effectiveness of User Stories in Practice. In *Requirements Engineering: Foundation for Software Quality*, M. Daneva and O. Pastor (Eds.). Lecture Notes in Computer Science, Vol. 9619. Springer, Cham, 187–202. doi:10.1007/978-3-319-30282-9_14

[12] Manjeshwar Aniruddh Mallya, Alessio Ferrari, Mohammad Amin Zadenoori, and Jacek Dąbrowski. 2026. RITA: A Tool for Automated Requirements Classification and Specification from Online User Feedback. *arXiv preprint arXiv:2601.11362* (2026).

[13] Ana Carolina Oran, Gleison Santos, Bruno Gadelha, and Tayana Conte. 2021. A framework for evaluating and improving requirements specifications based on the developers and testers perspective. *Requirements Engineering* 26, 4 (2021), 481–508.

[14] Jay U Oswal, Harshil T Kanakia, and Devvrat Suktel. 2024. Transforming Software Requirements into User Stories with GPT-3.5-: An AI-Powered Approach. In *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*. IEEE, 913–920.

[15] Tajmilur Rahman and Yuecai Zhu. 2024. Automated user story generation with test case specification using large language model. arXiv. Retrieved from https://arxiv.org/abs/2404.01558.

[16] Krishna Ronanki, Beatriz Cabrero-Daniel, and Christian Berger. 2024. ChatGPT as a Tool for User Story Quality Evaluation: Trustworthy Out of the Box?. In *Agile Processes in Software Engineering and Extreme Programming – Workshops (Lecture Notes in Business Information Processing, Vol. 489)*, P. Kruchten and P. Gregory (Eds.). Springer, Cham. doi:10.1007/978-3-031-48550-3_17

[17] Shadman Sakib, Oishy Fatema Akhand, Tasnia Tasneem, and Shohel Ahmed. 2026. From Reviews to Requirements: Can LLMs Generate Human-Like User Stories? *arXiv preprint arXiv:2603.28163* (2026).

[18] Malik Abdul Sami, Zheying Zhang, Muhammad Waseem, Kai-Kristian Kemell, Zeeshan Rasheed, Tomas Herda, Md Toufique Hasan, Jussi Rasku, and Pekka Abrahamsson. 2025. A multi-agent llm system for automated requirements analysis: a study on user story generation and prioritization. In *Euromicro Conference on Software Engineering and Advanced Applications*. Springer, 178–187.

[19] José Ruan Rodrigues Sampaio, Bruno Carvalho da Silva, Rodrigo Pereira dos Santos, and Davi Viana. 2025. Uma ferramenta para apoiar a análise de requisitos de software de IoT utilizando técnicas de agrupamento. In *Simpósio Brasileiro de Engenharia de Software (SBES)*. SBC, 935–941.

[20] Reine Santos, Gabriel Freitas, Igor Steinmacher, Tayana Conte, Ana Oran, and Bruno Gadelha. 2025. User Stories: Does ChatGPT Do It Better?. In *Proceedings of the 27th International Conference on Enterprise Information Systems - Volume 2: ICEIS*. INSTICC, SciTePress, 47–58. doi:10.5220/0013365500003929

[21] Reine Santos, Igor Steinmacher, Tayana Conte, Ana Oran, and Bruno Gadelha. 2025. AI-Generated User Stories: Are They Good Enough?. In *Anais do XXXIX Simpósio Brasileiro de Engenharia de Software* (Recife/PE). SBC, Porto Alegre, RS, Brasil, 741–747. doi:10.5753/sbes.2025.11321

[22] Ian Sommerville. 2011. *Software Engineering* (9th ed.). Pearson, Boston. https://engineering.futureuniversity.com/BOOKS%20FOR%20IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf?utm_source=chatgpt.com

[23] María José Torres-Igartua, Ángel J. Sánchez-García, Jorge Octavio Ocharán-Hernándezss, and Sodel Vázquez-Reyest. 2024. Analysis of Natural Language Processing Techniques and Tools for Requirements Elicitation: A Systematic Literature Review. In *2024 12th International Conference in Software Engineering Research and Innovation (CONISOFT)*. 48–57. doi:10.1109/CONISOFT63288.2024.00017

[24] Bill Wake. 2003. *Invest in good stories, and Smart Tasks*. https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/ Accessed: 28 April 2023.

[25] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, Anders Wesslén, et al. 2012. *Experimentation in software engineering*. Vol. 236. Springer.

[26] Ravi Teja Yarlagadda. 2021. Software engineering automation in IT. *International Journal of Innovations in Engineering Research and Technology* (2021). Retrieved from https://ssrn.com/abstract=3797346.

[27] Yifan Zhang, Yang Yuan, and Andrew Chi-Chih Yao. 2023. Meta prompting for ai systems. *arXiv preprint arXiv:2311.11482* (2023).

[28] Zhang Zheying, Rayhan Maruf, Herda Tomas, Goisauf Manuel, and Abrahamsson Pekka. 2024. LLM-based agents for automating the enhancement of user story quality: An early report. In *International Conference on Agile Software Development* (Cham). Springer Nature Switzerland, 117–126.