# Transforming user story definition: From deterministic to AI-powered automation

**Andra-Paula AVASILOAIE[1], Augustin SEMENESCU[1,2], Eduard-Cristian POPOVICI[1], Ionuț-Cosmin CHIVA[1]**

[1] National University of Science and Technology Politehnica Bucharest, Romania

[2] Academy of Romanian Scientists, Bucharest

andra.avasiloaie@stud.etti.upb.ro, augustin.semenescu@upb.ro, eduard.popovici@upb.ro, ionut_cosmin.chiva@upb.ro

**Abstract:** Agile software development necessitates the definition of user stories; however, the conventional manual process frequently leads to inefficiencies and inconsistencies. This paper investigates the shift from the deterministic user story creation to the AI-powered automation, addressing the limitations of the manual methods and the benefits of the AI integration. The study highlights the development of a deterministic JIRA extension for e-commerce applications. Despite its efficiency, the deterministic tool lacked adaptability across different domains. To overcome this, an AI-powered enhancement utilizing OpenAI's API was introduced, enabling scalable and accurate user story generation through natural language processing and machine learning. AI-driven tools automate user story creation, improving accuracy, reducing misinterpretation risks, and streamlining workflows. The research traces the evolution from a deterministic automation app using templates to an AI-driven app, emphasizing the role of prompt engineering in refining AI-generated outputs. The results demonstrate that AI integration not only enhances efficiency but also extends user story definition to diverse application domains, contributing to more scalable and adaptable software development practices.

**Keywords:** AI-powered automation, agile software development, user story generation, prompt engineering, natural language processing.

# Transformarea modului de definire a user story-urilor: de la un proces determinist la automatizarea asistată de inteligența artificială

**Rezumat:** Dezvoltarea software agilă presupune definirea clară a user story-urilor; totuși, procesul convențional, realizat manual, generează frecvent ineficiențe și inconsistențe. Acest articol investighează trecerea de la crearea deterministă a user story-urilor la automatizarea bazată pe inteligența artificială, abordând limitările metodelor manuale și evidențiind beneficiile integrării AI. Studiul evidențiază dezvoltarea unei extensii deterministe pentru JIRA, destinată aplicațiilor de comerț electronic. Deși eficient, acest instrument nu oferea adaptabilitate în diverse domenii. Pentru a depăși această limitare, a fost implementată o îmbunătățire bazată pe inteligență artificială, care utilizează API-ul OpenAI și permite generarea scalabilă și precisă a user story-urilor, prin procesarea limbajului natural și învățarea automată. Instrumentele bazate pe inteligența artificială automatizează crearea user story-urilor, îmbunătățind acuratețea, reducând riscurile de interpretare greșită și eficientizând fluxurile de lucru. Cercetarea urmărește evoluția de la o aplicație de automatizare deterministă care folosește șabloane la o aplicație bazată pe inteligența artificială, subliniind importanța prompt engineering-ului în rafinarea rezultatelor generate de AI. Rezultatele demonstrează că integrarea AI nu doar crește eficiența, ci extinde și procesul de definire a user story-urilor către diverse domenii de aplicație, contribuind la practici de dezvoltare software mai scalabile și adaptabile.

**Cuvinte-cheie:** automatizare asistată de AI, dezvoltare software agilă, generare de user story-uri, ingineria prompturilor, procesare a limbajului natural.

## 1. Introduction

Artificial Intelligence and its potential to automate traditionally laborious tasks have been of paramount interest in the ever-changing field of software engineering (Terragni, Roopb & Blincoe, 2024). This study investigates the shift from the deterministic user story creation to the AI-powered automation, addressing the limitations of the manual methods and the benefits of the AI integration.

https://doi.org/10.33436/v35i2y202505

The increasing complexity of software projects demands more efficient and adaptable approaches to requirements engineering, prompting the exploration of the AI-driven solutions (Sami et al., 2024). This research aims to contribute to the growing body of knowledge on AI in the software development by demonstrating the practical application of OpenAI's GPT-3 API for user story generation (Buruk, 2023).

Agile software development has long relied on user stories as a fundamental building block, guiding the iterative delivery of software functionality (Stephen, 2024). However, the traditional manual process of crafting user stories often results in inefficiencies and inconsistencies, hindering the overall development workflow (Sami et al., 2024). These challenges include communication complexities, early-stage uncertainties, and accurate resource estimation, further emphasizing the need for automated solutions (Sami et al., 2024). Moreover, maintaining consistency and clarity in user stories across large development teams can be difficult, leading to misinterpretations and ultimately impacting the quality of the final product (Bakhsh, Joy & Alam, 2024).

The initial approach involved developing a deterministic Jira extension tailored for e-commerce applications (Chinthapatla, 2024). Jira is a widely adopted project management software developed by Atlassian, designed specifically for the agile software development. It facilitates task management, sprint planning, issue tracking, and effective collaboration within development teams. The key features relevant to this article include structured templates for user story creation, backlog management, sprint tracking, and integration capabilities with external AI modules. This rule-based tool demonstrated promising results, achieving up to 60% time savings in the user story creation. Similar deterministic approaches have been explored in other domains, showcasing the potential for automation in specific contexts (Altawaiha & Al-Hgaish, 2024). However, its inherent reliance on pre-defined rules limited its adaptability and scalability across different project domains.

Recognizing the limitations of the deterministic approaches, the potential of the AI-driven automation using OpenAI's GPT-3 API (OpenAI, 2020; OpenAI, 2023) was explored. By utilizing the capabilities of the large language models, we aim to address the challenges of scalability, adaptability, and consistency in user story generation, ultimately contributing to more streamlined and effective software development practices (Haile, 2024a; Rimon, 2024). The paper details the development, evaluation, and implications of this AI-driven approach, focusing on the benefits, challenges, and best practices for integrating the AI into the user story generation process.

While similar research has explored the application of the AI in software testing and quality assurance (Bakhsh, Joy & Alam, 2024; Haile, 2024b), this study specifically focuses on the transformative potential of the AI-driven automation within the business analysis domain. By leveraging the capabilities of the large language models like GPT-3, it aims to address the unique challenges faced by business analysts in defining and refining user stories.

While (Pîrcălabu, Țăpuș & Damian, 2024) focus on providing programming suggestions and code generation within an integrated development environment, this research leverages similar principles of natural language processing and machine learning to automate and enhance the creation of user stories. By applying these techniques to the requirements engineering phase, we aim to achieve comparable improvements in efficiency, accuracy, and consistency, ultimately streamlining the software development lifecycle.

While reviewing recent advances in the AI-driven user story generation, it was observed that many existing frameworks – though innovative – present either high complexity or lack sufficient implementation details, making direct replication in a production context challenging. Consequently, this research focuses on developing a pragmatic, easily integrated solution tailored to Jira’s ecosystem, prioritizing simplicity, transparency, and operational ease.

The subsequent sections will begin with a review of the related work on comparable applications. Then there will be provided an overview of the deterministic app and the AI-driven app, including its transformation. Additionally, there will be outlined the prompt engineering work and the results obtained. Finally, the conclusions will be presented, along with directions for the future research.

## 2. Related work

Related work AI assistance in writing user stories is gaining traction to enhance the development workflows and the communication between teams. Tools like Atlassian's Jira and UserStoryGen automate story creation and refinement, reducing manual effort. Generative AI models such as ChatGPT and Claude by Anthropic help draft and iterate user stories, ensuring clarity and efficiency.

### 2.1. User story generation

Agile methodology is an iterative and incremental approach to the software development, emphasizing flexibility, collaboration, and continuous improvement. A core practice within Agile is the use of user stories – concise, clearly written requirements expressed from the user's perspective. These are typically formatted as: "As a [role], I want [functionality], so that [benefit]." User stories help ensure that the development efforts remain focused on user needs and business value. In the traditional deterministic user story generation, predefined templates and explicit rules are used to standardize story creation without involving AI-based dynamic adjustments.

Jira by Atlassian has incorporated the AI-driven features that assist in refining user stories and managing product backlogs. Leveraging machine learning, Jira can suggest story structures, auto-complete acceptance criteria, and identify inconsistencies in requirements (Mansour, 2023). Another approach proposed by (Zhang et al., 2024) is AgileGen, which uses testable requirements written in Gherkin to maintain the semantic consistency between the requirements and the generated code. While AgileGen focuses on ensuring code aligns with user needs through testable requirements, this research focuses on the earlier stage of requirements engineering, specifically automating and enhancing the generation of user stories themselves. Similarly, there will be leverage the Gherkin standard in defining acceptance criteria for the user stories generated by the AI-powered approach.

### 2.2. Natural language AI assistants for user story generation

OpenAI is a widely used tool that assists product managers in drafting user stories. It can generate stories from high-level feature descriptions, suggest edge cases, and provide validation criteria, accelerating backlog grooming. Product managers can refine stories by engaging with ChatGPT, allowing for dynamic adjustments and creative exploration. (API Platform | OpenAI, 2024).

Anthropic's Claude operates in a similar fashion, offering safety-focused AI assistance in generating and refining user stories. Designed to provide thoughtful and context-aware responses, Claude can help ensure that the generated user stories align with ethical considerations and user-centric principles. This is particularly beneficial for the industries with stringent compliance requirements, such as healthcare and finance (Anthropic, 2023).

The present work builds upon this foundation by introducing a specialized user story assistant integrated with Jira through a custom extension. By leveraging OpenAI's API, the assistant not only automates the user story generation but also enables the iterative refinement through conversational interactions. This approach seeks to decrease the time needed for backlog refinement while preserving user stories of high quality and standardization that align with the project objectives and possess a well-structured format, validated through the collaboration with the development teams. These tools and the AI assistants form the backbone of the current automation strategies. However, recent research studies have advanced this field further, offering systematic reviews and new frameworks that complement and extend the existing solutions.

### 2.3. Recent research advancements

Recent research has emphasized the importance of automating user story generation through various AI-based techniques. The study (Dos Santos, Bouchard & Petrillo, 2024a) presented a comprehensive systematic literature review, identifying a variety of Natural Language Processing

(NLP) and Machine Learning (ML) approaches for generating user stories. Their study highlighted a significant limitation: the scarcity of publicly available datasets, which poses challenges to training highly effective models and evaluating the quality of the generated stories objectively.

Furthermore, in a separate study, (Dos Santos, Bouchard & Napoleão, 2024b) demonstrated an AI-driven framework specifically leveraging GPT models and advanced prompt engineering techniques to automate user story creation. Their research underlines the importance of a clear communication between the stakeholders and the development teams, emphasizing the iterative refinement of the stories. Additionally, (Zadeh et al., 2024) explored the broader integration of the AI within the agile project management, identifying key innovations, challenges, and substantial benefits, such as the improved backlog management, optimized team collaboration, and effective risk mitigation strategies.

# 3. From templates to the AI-powered generation

## 3.1. Deterministic user story definition: the JIRA extension

The present initial approach to user story automation involved the development of a deterministic Jira extension tailored for e-commerce applications.

A well-crafted user story should provide a clear and succinct overview of the users’ goal, the functionality they require, and the reason or motivation behind their need. This information helps to align the development team's efforts with the actual user requirements, ensuring the software being built is relevant and valuable to the end users.

A validated structure for complex user stories was established, which consists of three key parts: a short introduction that includes the user persona, the user's goal, and the user's reason or objective; a detailed description that provides an overview of the functionality and its implementation, including the relevant elements and behaviors; and the user’s acceptance criteria following the Gerkins standard and a structured format. According to the previous research, formatting a user story alone can account for up to 20% of the time spent writing it.

To address the challenges associated with the manual user story creation, a deterministic JIRA extension was developed, targeting the specific needs of e-commerce applications (Sami et al., 2024). The extension incorporated a predefined template for user stories, ensuring consistency and reducing ambiguity. It served as a base for the user story definition process, guiding users through a structured workflow. By leveraging the JIRA integration and the predefined template, the deterministic extension achieved a time savings of up to 60% compared to the traditional manual approach to user story creation.

The steps in creating user stories involve choosing the epics that will serve as the high-level objectives or themes for any software application and capturing the broad functionality and capabilities to be delivered to users. Next, choose the user stories that are part of each epic, such as allowing new users to create an account using their email address, enabling returning users to log in with their username and password, providing a way for customers to reset their password if forgotten, and ensuring users can stay logged in after navigating away from the website. Personalize the templates with the specific details and requirements for the e-commerce application, customizing user personas, adjusting acceptance criteria, and aligning the examples and language with the context of the e-commerce project. Tailoring the user story templates to the unique business needs will help ensure the defined user stories are relevant, comprehensive, and accurately capture the expectations of the end users. Finally, create the user stories.

The deterministic solution achieved significant improvements in efficiency and consistency, saving up to 60% of the time required for user story writing in the e-commerce use case. This demonstrated the potential for streamlining user story creation within a specific application domain.

## 3.2. Limitations of the deterministic approach

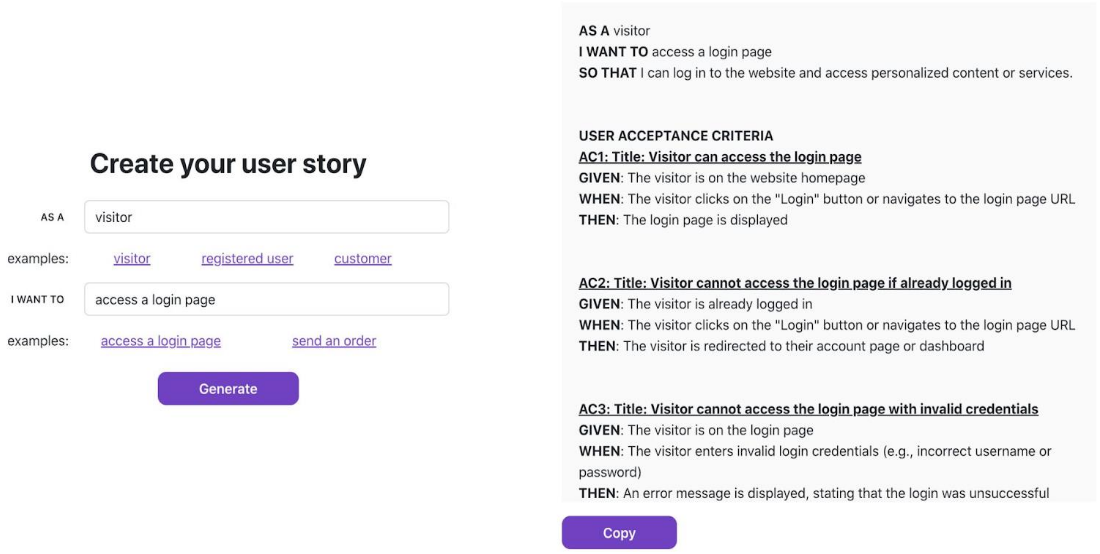
While the deterministic JIRA extension proved effective in the e-commerce domain, its reliance on rigid, predefined templates limited the adaptability across different application areas. Each industry and project context introduce unique requirements, user personas, and terminology that must be accurately reflected in user stories.

Extending this deterministic approach required manually developing and maintaining domain-specific templates, a process that was both time-consuming and error-prone, demanding significant domain expertise. These constraints reduced scalability and hindered responsiveness in the agile environments where requirements evolve frequently.

Recent studies (Dos Santos, Bouchard & Napoleão, 2024b; Zadeh, Khoulenjani & Safaei, 2024) have also highlighted the inherent difficulties the deterministic systems face in adapting to nuanced project needs and dynamic stakeholder expectations. To overcome these challenges, transitioning to an AI-enhanced solution that leverages natural language processing to interpret diverse project contexts and autonomously generate user stories aligned with best practices. The detailed workflow supporting this AI-powered approach will be presented in the following section.

## 3.3. AI-powered user story generation

To address the limitations of the deterministic approach, OpenAI's API has been integrated into the user story definition tool, enabling the AI-driven automation and scalability. The AI-powered solution utilizes the natural language processing and machine learning techniques to analyze project requirements, extract relevant information, and generate high-quality user stories, following the improved structure and formatting presented above.



AS A visitor

I WANT TO access a login page

**AS A** visitor
**I WANT TO** access a login page
**SO THAT** I can log in to the website and access personalized content or services.

**USER ACCEPTANCE CRITERIA**

**AC1: Title: Visitor can access the login page**
**GIVEN**: The visitor is on the website homepage
**WHEN**: The visitor clicks on the "Login" button or navigates to the login page URL
**THEN**: The login page is displayed

**AC2: Title: Visitor cannot access the login page if already logged in**
**GIVEN**: The visitor is already logged in
**WHEN**: The visitor clicks on the "Login" button or navigates to the login page URL
**THEN**: The visitor is redirected to their account page or dashboard

**AC3: Title: Visitor cannot access the login page with invalid credentials**
**GIVEN**: The visitor is on the login page
**WHEN**: The visitor enters invalid login credentials (e.g., incorrect username or password)
**THEN**: An error message is displayed, stating that the login was unsuccessful

**Figure 1.** BA-Assistant client interface

The interface presented in Figure 1 is a proprietary research contribution developed within this study, allowing Business Analysts and Product Owners to easily generate user stories by simply specifying the user type and the action they need to perform. This streamlined process simplifies the creation of user stories, as it automatically generates the required structure based on just two key inputs: the role of the user and the desired action. By reducing the need for manual entry and ensuring consistency, this feature enhances productivity and helps ensure that the user stories are aligned with the project's goals and requirements.

### 3.3.1. First prototype's architecture

The high-level architecture depicted in Figure 2 reflects a tailored AI-powered system developed as part of this research, designed to process and respond to user queries by leveraging OpenAI's technologies. Each component within the architecture is detailed below to illustrate its role in enabling efficient and context-aware user story generation.

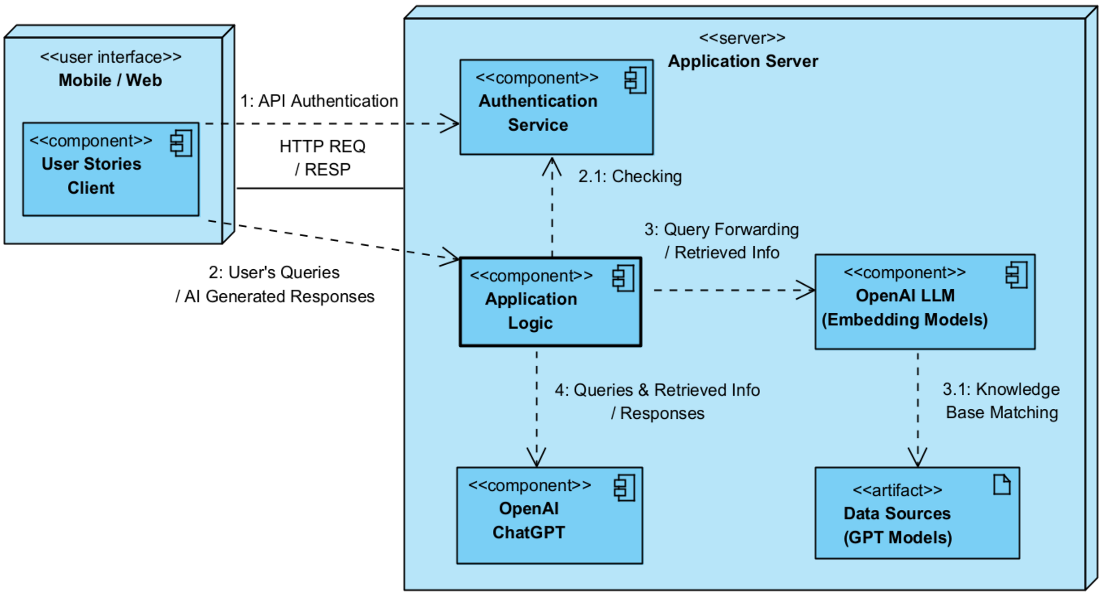


**Figure 2.** High-level architecture of the first AI-driven application prototype

The system operates through a combination of interconnected components, ensuring seamless and scalable performance from the client interface to the backend models. This integrated process facilitates an efficient query handling and response generation, creating a streamlined user experience.

The process begins when the client interface gathers the user's query and transmits it to the server of the application. The server of the application then forwards the query to the OpenAI Embedding Models, which convert the input into vector representations. These vectors are matched against stored data within a vector database or other data sources to identify the most relevant content. The retrieved information, alongside the original user query, is sent to the OpenAI GPT model to generate an appropriate response. The final response is delivered to the client interface for display to the user, completing the interaction loop. The client interface serves as the primary point of user interaction. Whether through a web or mobile platform, the interface is crafted to be intuitive and responsive, enabling users to easily submit queries or requests. Upon receiving user input, the client interface efficiently transmits the data to the server of the application and later displays AI-generated responses. This interface simplifies the communication between users and the system, fostering an engaging experience.

The server of the application functions as the central hub, orchestrating the flow of information between the client interface, OpenAI Embedding Models, and GPT models. It processes incoming queries, prepares them for embedding and search operations, and manages the communication with the OpenAI models. Additionally, the server of the application handles the session management, ensuring the continuity for longer interactions such as chatbot conversations or virtual assistant engagements. By coordinating each stage of the query processing, the server ensures timely and accurate generation of user responses. Data sources play a crucial role in housing the knowledge necessary for generating responses. These sources encompass local databases, files, and structured or unstructured data repositories. Using OpenAI Embedding Models, this data is indexed into vectors, allowing for efficient semantic search that prioritizes meaning over exact keyword matches. This enhances the accuracy of the content retrieval and expands the system's capacity to handle diverse queries across various domains.

The OpenAI Embedding Models are responsible for transforming both user queries and stored data into vector embeddings. These numeric representations capture the semantic essence of the text, facilitating precise and meaningful searches. By enabling similarity-based searches, the embedding models help locate relevant content within the vector database, enhancing the quality and relevance of the retrieved information. Upon retrieval of relevant content, the OpenAI GPT model generates the final response. With advanced language capabilities, GPT processes user’s input along with retrieved knowledge to create intelligent, context-aware replies. The model excels in natural language understanding, producing coherent, accurate, and human-like text for a wide range of user queries.

The generated response is routed back to the server of the application, which subsequently transmits it to the client interface for user display. This marks the conclusion of the query cycle, with the user receiving a well-formulated, informative answer. The system may also present additional resources, such as links or visual aids, enhancing the overall user experience. The feedback collected during the interactions can be used to refine and optimize the future performance, ensuring a continuous improvement in the system’s capabilities.

### 3.3.2. Data and prompt engineering

The AI-powered user story generation relied on a dataset of existing user stories, used as templates in the JIRA extension application. This data was used to fine-tune and prompt the GPT-3 model, enabling it to generate user stories that adhered to best practices and maintained consistency with industry standards.

Prompt engineering played a crucial role in refining the AI-generated outputs and ensuring the relevance and quality of the produced user stories. As part of this research, prompt engineering focused on improving the structure, content, and format of prompts to generate user stories that align with expected standards. This process ensures that the AI consistently produces stories that adhere to the desired structure and formatting conventions, as illustrated in Figure 3.

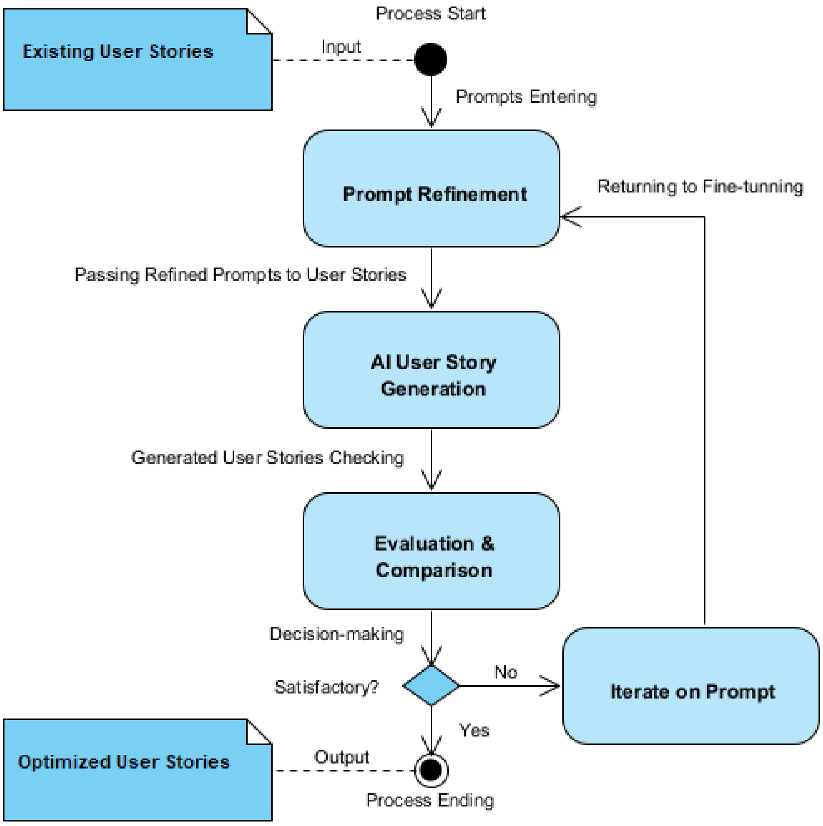


**Figure 3.** Prompt engineering workflow

When incorporating the AI into the user story definition, the prompt must be designed to guide the model in accurately understanding the context, requirements, and project-specific conventions. Crafting effective prompts may involve supplying sample user stories, outlining essential components such as user personas, acceptance criteria, and business value, and specifying the tone and style. By clearly defining these elements, the AI can generate more relevant and structured user stories.

Through iterative refinement and continuous testing, the AI tool can be adjusted to produce user stories that are comprehensive and closely aligned with the project's needs. This ongoing prompt engineering is vital to maintaining the quality and consistency of the AI-generated user stories, allowing them to integrate seamlessly into the broader software development process. Iterating on prompts plays a significant role in enhancing the AI-driven user story creation. One approach involves experimenting with different structures, such as initiating prompts with a high-level description versus a detailed template, to determine which method yields the most relevant outputs. Including sample user stories or project-specific information further helps guide the language model, ensuring the generated outputs meet established standards and reflect the project's objectives.

Various prompt styles can also be tested, presenting requirements as either bullet points or in narrative form, to identify which format delivers clearer and more actionable user stories. By systematically testing these prompts, the AI's outputs can be evaluated for gaps or inconsistencies, allowing for adjustments to improve accuracy and completeness.

The continuous iteration based on user feedback and domain-specific requirements ensures that the prompts evolve alongside the project's needs. This iterative process refines the quality and relevance of the AI-generated user stories, reinforcing their value in streamlining development workflows. Ultimately, the success of the AI-powered user story definition relies on the ability to craft, test, and refine prompts to achieve optimal performance and alignment with project goals.

To further assess the robustness and flexibility of the proposed solution, comparative tests were conducted using two different AI models: ChatGPT and Gemini. These tests aimed to evaluate whether different models would yield consistent and high-quality user stories when given similar prompts. For ChatGPT, we utilized prompts such as: "*Given a user role [Customer], generate a user story describing the checkout process emphasizing usability and security*" and "*Generate a detailed user story for adding and managing product returns in an e-commerce application*." Corresponding prompts for Gemini were phrased to maintain the conceptual equivalence while ensuring the clarity of the instruction: "*Write a concise user story using the format 'As a..., I want..., so that...', focusing specifically on improving user experience during the online checkout process*" and "*Create a clear and structured user story about efficiently handling product returns within an online retail platform*.". This comparative approach highlighted not only the capabilities of each model in adhering to standardized user story formats but also the differences in the level of detail, phrasing, and emphasis provided. While both models produced coherent and structurally sound user stories, ChatGPT demonstrated a higher degree of contextual understanding and completeness, particularly in the scenarios requiring nuanced acceptance criteria. Gemini, by contrast, excelled in generating concise and well-structured outputs but occasionally lacked depth in elaborating on the business value or acceptance tests. These findings reinforce the relevance of selecting the AI tools based on project-specific needs and suggest that, at this stage, ChatGPT may offer a slight advantage in more complex or detailed requirements engineering contexts.

In the current approach, fine-tuning was not performed on any specific dataset, opting instead to leverage GPT-3's inherent capabilities enhanced by advanced prompt engineering. This choice significantly simplified the implementation while maintaining an effective story generation. Nevertheless, for future research or industry-scale implementations, datasets such as the publicly available "*User Story NeoDataset*" (over 20,000 user stories) from Mendeley could be effectively utilized for fine-tuning, potentially further improving generated user story accuracy, domain relevance, and context specificity.

To illustrate the practical application of the AI-powered assistant, Table 1 presents a set of representative prompts, the corresponding AI-generated user stories, and brief evaluative comments. For instance, when prompted to "Generate a user story for a customer completing a secure online checkout", the system responded with: "As a customer, I want to securely complete my payment during checkout, so that I can safely finalize my purchases without concerns about data breaches." This response demonstrates structural clarity and relevance to the user goal, as well as attention to security – a critical aspect in e-commerce contexts.

**Table 1.** Examples of AI-generated user stories based on specific prompts in e-commerce context

| Prompt | AI Response | Comments |
|---|---|---|
| Generate a user story for a customer completing a secure online checkout. | As a customer, I want to securely complete my payment during checkout, so that I can safely finalize my purchases without concerns about data breaches. | The story is well-structured and addresses both the user goal and the security aspect.<br>Clear and actionable. |
| Create a user story describing how a customer can return a product through the online portal. | As a customer, I want to initiate a return for my purchased items through the online portal, so that I can easily send back products that do not meet my expectations. | Concise and clear.<br>Successfully captures the key functionality and user benefit. |
| Write a user story for a logged-in user viewing their order history. | As a logged-in user, I want to view my complete order history, so that I can track my past purchases and manage my orders efficiently. | Accurate and aligned with standard user story format.<br>Useful for backlog refinement. |
| Generate a user story about receiving notifications for shipped orders. | As a customer, I want to receive notifications when my order is shipped, so that I am informed about the delivery status of my purchases. | Covers both functionality and user value.<br>Precise and relevant for enhancing user experience. |

These examples confirm that the AI-generated user stories maintain structural consistency and practical relevance, aligning well with Agile standards and ensuring clarity for the development teams. Moreover, the quality and precision of these outputs directly reflect the prompt engineering workflow illustrated in Figure 3, where iterative design and refinement steps help ensure that the AI produces responses adhering to expected formats and project requirements. This alignment demonstrates that a well-structured prompt engineering process not only enhances output clarity but also strengthens reproducibility across different use cases.

## 4. Conclusions and future work

The research findings highlight a clear evolution in the process of user story definition, moving from the manual, deterministic methods to the AI-powered solutions. The deterministic JIRA extension explored in this study illustrates the potential to streamline user story creation within specific domains, such as the e-commerce. However, the integration of the AI, as demonstrated using OpenAI’s API, has significantly broadened the capabilities of user story definition tools. The AI-powered approach introduces a greater adaptability, enhanced efficiency, and improved accuracy compared to the deterministic methods.

Unlike the deterministic solution, which was confined to the e-commerce applications, the AI-driven approach proves versatile across a wide range of domains. This adaptability allows for configuration to support application types ranging from healthcare to finance by fine-tuning the language model and adjusting training data. The automated generation of user stories reduces the time and effort required, allowing the development teams to focus on strategic objectives. Additionally, the AI model’s capacity to interpret context and apply domain-specific knowledge results in more consistent and accurate user stories, mitigating the risk of misinterpretation.

The scalability of the AI-powered tools further enhances their applicability to enterprise-level projects with complex requirements. These tools can manage high volumes of user stories while maintaining quality and coherence. A key advantage lies in the continuous improvement capabilities of the AI models, which learn from users' feedback and past data to refine outputs and enhance relevance over time. Through the AI integration, user story definition has transformed into a more scalable, efficient, and accurate process, equipping the development teams with tools to drive greater success in their software initiatives.

Although comparable methodologies from literature offer valuable frameworks, the research deliberately avoided direct implementation of these models. The main reasons include the absence of any detailed technical guidance—such as specific prompt engineering strategies or dataset configurations—and the high complexity of certain proposed solutions, which are often domain-specific. Instead, the approach emphasizes the operational simplicity and seamless Jira integration, ensuring a practical applicability across diverse projects while maintaining a scalability and high-quality output.

Despite these advancements, several challenges remain. Ensuring data privacy and security is paramount, as the AI models often rely on sensitive project data. Robust measures must be implemented to safeguard user's information. Transparency and explainability are equally important for building trust with users, necessitating the development of features that clarify the reasoning behind the AI-generated user stories. Furthermore, maintaining a balance between automation and human oversight is essential to preserve the quality and contextual accuracy of the outputs.

The future research and development efforts should aim to address these challenges while expanding the scope of the AI-powered solutions in the software development. One avenue for the future work is the development of a new version of the tool that leverages the latest ChatGPT API, enhancing the AI's capabilities further. This upgraded version will integrate with JIRA, providing a seamless extension for user story creation across diverse project environments. Such advancements hold the potential to redefine the landscape of the agile development by offering more intelligent, automated, and adaptive tools that align with the evolving software demands.

## REFERENCES

Altawaiha, I. & Al-Hgaish, A. (2024) ClassDiagGen Tool: Fine-Tuning the GPT-3 Model for Auto- mated Class Diagram Generation from Textual Descriptions. *Research Square* [Preprint]. doi:10.21203/rs.3.rs-4350615/v1.

Anthropic (2023) *Introducing Claude.* https://www.anthropic.com/news/introducing-claude [Accessed 6 January 2025].

Bakhsh, M.M., Joy, M.S.A. & Alam, G.T. (2024) Revolutionizing BA-QA Team Dynamics: AI-Driven Collaboration Platforms for Accelerated Software Quality in the US Market. *Journal of Artificial Intelligence General Science.* 7(1), 63–76. doi:10.60087/jaigs.v7i01.296.

Buruk, O. O. (2023) *Academic Writing with GPT-3.5: Reflections on Practices, Efficacy and Transparency.* arXiv [Preprint]. doi:10.48550/arXiv.2304.11079.

Chinthapatla, S. (2024) Data Engineering Excellence in the Cloud: An In-Depth Exploration. *International Journal of Science Technology Engineering and Mathematics*. 13(3), 11-18.

Dos Santos, C.A., Bouchard, K. & Napoleão, B.M. (2024b) Automatic User Story Generation: A Comprehensive Systematic Literature Review. *International Journal of Data Science and Analytics.* 20(1), 1-24. Doi: 10.1007/s41060-024-00567-0.

Dos Santos, C.A., Bouchard, K. & Petrillo, F. (2024a) AI-Driven User Story Generation. *Proceedings of the 2024 International Conference on Advances in Computer and Data Sciences*

*Applications (ACDSA), 1-2 February 2024, Victoria, Seychelles*. pp. 1-6. IEEE. doi:10.1109/ACDSA59508.2024.10467677.

Haile, A. (2024a) AI-Driven Software Testing Automation: Machine Learning Strategies for Performance Optimization in Distributed Networks. *ResearchGate.* doi:10.13140/RG.2.2.21010.16320.

Haile, A. (2024b) Innovating Software Testing with AI and Machine Learning: Automation for Efficient Distributed Network Management. *ResearchGate*. doi:10.13140/RG.2.2.17654.72004.

Mansour, S. (2023) *Atlassian welcomes AI to the team.* https://www.atlassian.com/blog/announcements/atlassian-intelligence-ga [Accessed 6 January 2025].

OpenAI (2020) *OpenAI API*. https://openai.com/api/ [Accessed 6 January 2025].

OpenAI (2023) *GPT-3 model*. https://platform.openai.com/docs/models/gpt-3 [Accessed 6 January 2025].

OpenAI (2024) *API Platform*. https://openai.com/api/ [Accessed 6 January 2025].

Pîrcălabu, T., Țăpuș, N. & Damian, A.I. (2024) Programming assistance based on Naeural AI OS Platform*. Romanian Journal of Information Technology and Automatic Control [Revista Română de Informatică și Automatică]*. 34(4), 43-54. doi:10.33436/v34i4y202404.

Rimon, S.M.T.H. (2024) Leveraging Artificial Intelligence in Business Analytics For Informed Strategic Decision- Making: Enhancing Operational Efficiency, Market Insights, And Competitive Advantage. *Journal of Artificial Intelligence General Science (JAIGS).* 6(1), 600-625. doi:10.60087/jaigs.v6i1.278.

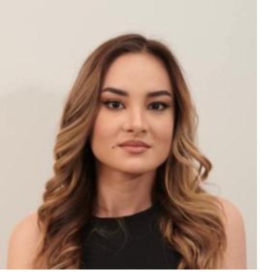
Sami, M. et al. (2024) *AI based Multiagent Approach for Requirements Elicitation and Analysis*. arXiv [Preprint]. doi:10.48550/arxiv.2409.00038.

Stephen, M. (2024) The Intersection of Technology and Writing Support: How ChatGPT is Changing Writing Center Dynamics. *ResearchGate*: https://www.researchgate.net/publication/383263568_The_Intersection_of_Technology_and_Writing_Support_How_ChatGPT_is_Changing_Writing_Center_Dynamics [Accessed 6 January 2025].

Terragni, V., Roop, P. & Blincoe, K. (2024) *The Future of Software Engineering in an AI-Driven World*. arXiv [Preprint]. doi:10.48550/arXiv.2406.07737.

Zadeh, E. K., Khoulenjani, A. B. & Safaei, M. (2024*)* Integrating AI for Agile Project Management: Innovations, Challenges, and Benefits. *International Journal of Industrial Engineering and Construction Management.* 1(1), 1-10.

Zhang, S., Xing, Z., Guo, R., Xu, F., Chen, L., Zhang, Z., Zhang, X., Feng, Z. & Zhuang, Z. (2024) Empowering Agile-Based Generative Software Development through Human-AI Teamwork. *arXiv* [Preprint]. doi:10.48550/arxiv.2407.15568.



**Andra-Paula AVASILOAIE** is a doctoral candidate in Engineering and Management at the National University of Science and Technology Politehnica Bucharest, Romania, where she previously earned a Master's degree in Computer Science. She is the co-founder of a software development product that provides cloud infrastructure and an advanced interface, enabling developers to deploy full-stack applications with maximum speed and efficiency. Her professional background encompasses business analysis, software development management, and product marketing.

**Andra-Paula AVASILOAIE** este doctorand în Inginerie și Management la Universitatea Națională de Știință și Tehnologie Politehnica din București, unde a obținut anterior un master în Informatică. Este cofondatoarea unui produs software care oferă infrastructură cloud și o interfață avansată, permițând dezvoltatorilor să lanseze aplicații full-stack cu viteză și eficiență maxime. Experiența sa profesională include analiză de business, managementul dezvoltării aplicațiilor software și marketing de produs.

**Augustin SEMENESCU** is a Habilitated Professor at the National University of Science and Technology Politehnica Bucharest, Romania, with over 30 years of experience in engineering, mathematics, and economics. His research interests include modelling and control processes, metallurgical equipment, applied mathematics, and materials science. Professor Augustin Semenescu has authored over 50 patents in engineering and medical fields, more than 30 innovation patents, and over 50 books and articles related to engineering, management and marketing. He has been involved in over 100 national and international projects, serving as project director or responsible person, and has been a team member in over 150 projects since 1984. His contributions have been recognized with numerous awards, including over 300 gold medals and 20 silver medals at international exhibitions. Professor Semenescu is an Corresponding Member of the Academy of Romanian Scientists and a Full Member of the American Romanian Academy of Arts and Sciences, reflecting his esteemed standing in the academic community.

**Augustin SEMENESCU** este profesor abilitat la Universitatea Națională de Știință și Tehnologie Politehnica din București. Are o experiență de peste 30 de ani în inginerie, matematică și economie. Domeniile sale de cercetare includ modelarea și controlul proceselor, echipamentele metalurgice, matematica aplicată și știința materialelor. Profesorul Augustin Semenescu este autor a peste 50 de brevete în inginerie și medicină, a mai mult de 30 de brevete de inovație, precum și a peste 50 de lucrări științifice și articole în domeniile ingineriei, managementului și marketingului. Din 1984 până în present, a fost implicat în peste 250 de proiecte naționale și internaționale, ăn calitete de director de proiect, responsabil sau membru al echipei. Contribuțiile sale au fost recunoscute la nivel internațional prin numeroase premii, incluzând peste 300 de medalii de aur și 20 de medalii de argint obținute la saloane internaționale de inventică. Profesorul Augustin Semenescu este Membru Corespondent al Academiei Oamenilor de Știință din România și membru titular al Academiei Româno-Americane de Arte și Științe, ceea ce reflectă recunoașterea de care se bucură în comunitatea academică.

**Eduard-Cristian POPOVICI** is an Associate Professor at the National University of Science and Technology Politehnica Bucharest, Romania, specializing in telecommunications, software architecture, AI-driven systems, machine learning, cybersecurity, and modern communication networks, including 4G/5G and IoT. He has led numerous research projects, driving advancements in autonomous cybersecurity systems, Blockchain, quantum computing, and smart environments. Professor Eduard-Cristian Popovici has authored over 50 scientific papers and contributed to more than 40 research projects, including over 20 international initiatives.

**Eduard-Cristian POPOVICI** este conferențiar universitar la Universitatea Națională de Știință și Tehnologie Politehnica București, România, specializat în telecomunicații, arhitecturi software, sisteme bazate pe inteligență artificială, învățare automată, securitate cibernetică și rețele moderne de comunicații, inclusiv 4G/5G și Internet of Things (IoT). A coordonat numeroase proiecte de cercetare, contribuind semnificativ la progrese în domeniul sistemelor autonome de securitate cibernetică, Blockchain, calcul cuantic și medii inteligente. Profesorul Eduard Popovici a publicat peste 50 de lucrări științifice și a contribuit la peste 40 de proiecte de cercetare, dintre care peste 20 au fost inițiative internaționale.

**Ionuț-Cosmin CHIVA** is an Assistant Lecturer at the National University of Science and Technology Politehnica Bucharest, Romania, specializing in telematics systems, intelligent localization, and traffic management using Wi-Fi and Bluetooth monitoring. He has co-authored over 40 scientific papers and is pursuing a Ph.D. in Industrial Engineering and Robotics, focusing on augmented localization technologies.

**Ionuț-Cosmin CHIVA** este asistent universitar la Universitatea Națională de Știință și Tehnologie Politehnica București, specializat în sisteme telematice, localizare inteligentă și managementul traficului utilizând monitorizare Wi-Fi și Bluetooth. A fost coautor a peste 40 de lucrări științifice și urmează studii doctorale în Inginerie Industrială și Robotică, concentrându-se pe tehnologii de localizare augumentată.