Frontiers in Emerging Technology
ISSN *(Online)*: 3104-509X
Volume 2, Issue 6, Nov-Dec 2026, Page 1-14

Research Article Received: 12-09-2026| Accepted: 10-10-2026| Published: 07-11-2026

Human-in-the-Loop Evaluation of LLM-Assisted Requirements Engineering in Regulated Enterprises

Akarambaig Saudagar *[1]

Abstract

The Business Systems Analyst (BSA) leads process and technology improvements, facilitates interaction and collaboration between business and technology teams, balances business and technology needs, and translates business problems and requirements into documented system requirements. Projects and businesses need quality to be successful. Common symptoms include unclear and vague requirements, stakeholder confusion, outdated requirement documents, scope creep, and missing requirements traceability. These problems can put a project on the fast track and cost time and money to rework and refactor, leading to project problems. This shift toward automation and AI within the BSA is reshaping the landscape and requires reaffirming and validating new procedures and processes. This research seeks to build and evaluate a method that, given raw notes from a JAD session, generates a structured Business Requirements Document (BRD) and corresponding user stories using prompt engineering and a large language model (LLM). Ambiguity, traceability, completeness, and the time needed to document the LLMs' work and human evaluation will serve as metrics and criteria for evaluating the quality of the LLMs' outputs against the human-generated baseline. Experimental results have not yet been acquired. The study will compare the time spent by the analyst, the LLM, and the LLM with the analyst to complete documentation, as well as the number of ambiguities, completeness, and traceability observed during the documentation process. This evaluation method is a proposed step-by-step approach for evaluating AI-supported Requirements Engineering in a regulated context. In a future empirical comparison and human review, the reflections on quality, efficiency, and performance improvements will be based on the proposed empirical comparison.

Keywords

Business Systems Analyst, Requirements Engineering, Large Language Models, Prompt Engineering, Human-in-the-Loop, Agile Development, Traceability

1 Senior Business Systems Analyst, MA, USA

INTRODUCTION

Business Systems Analysts (BSAs) play an influential role in today's organizations by translating business needs into system design and development and serving as a liaison between business stakeholders and technical teams. This bridging role becomes even more involved in enterprise Digital Transformation initiatives, which involve many aspects, such as Customer-Facing and Operational Applications [1]. In a regulated industry like investment management, banking, or public-sector health & human services, the success of a project, the readiness for auditing, and regulatory compliance and adherence depend on the quality of the requirements artefacts that are created, including the Business Requirements Document (BRD), Functional Requirements Document (FRD), user stories, and Requirements Traceability Matrix (RTM) [2]. System implementation is key because the accuracy of the requirements the BSA elicits, analyzes, and documents determines whether the desired business objectives are achieved while satisfying regulatory requirements. Generative AI technologies such as Claude, ChatGPT, and Microsoft Copilot have quickly entered the stage, integrated into enterprise processes, and opened new opportunities to improve requirements elicitation, documentation, and management.

While requirements engineering methods have been well developed over the last decades, capturing the results of a Joint Application Development (JAD) session, or a stakeholder workshop, as structured BRDs

***Corresponding Author:** Akarambaig Saudagar
© The Author(s) 2025, This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC-BY-NC)
DOI: 10.65796/fet.vol.2.issue.6.1

and user stories remains largely manual and time-consuming [3]. Session notes may be unstructured, ad hoc notes from stakeholders and conversations, with caveats or phrases, and business context and technical constraints are identified in a fragmented way across stakeholders, sessions, analysts, and projects. Each analyst, depending on their level of experience and the amount of work they have had to do in a given time, will create documentation of differing quality that causes ambiguous requirements, incomplete acceptance criteria, poor traceability between requirements and test cases, scope creep, and expensive rework later in the delivery lifecycle. These risks are also magnified in regulated areas, including benefits eligibility and enrollment, investment operations, and commercial lending, where unclear or missing requirements may lead to compliance gaps and audit findings.

Traditional requirements documentation methods depend heavily on the analyst's judgment to interpret stakeholder information, derive implicit business rules, and define acceptance criteria for development and testing teams [4]. With under-specified requirements, projects can suffer from late delivery, cost overruns, and post-deployment defects, causing loss of stakeholder confidence and of business in the organization. Previous studies on the use of Large Language Models (LLMs) in requirements engineering are mostly conducted in the context of general software development, academic datasets, or individual requirements-related tasks, with limited evidence of use in the end-to-end Business Rules Analysis (BDA) process of transforming stakeholder session notes into structured Business Rules Documents (BRDs) and User Stories in regulated enterprise environments [5; 6]. However, there is also not that much practitioner advice on how to use measurable criteria like specificity, ambiguity, completeness, or likelihood of change to compare between the AI-generated requirements and analyst-written baseline requirements, nor are there many directions on how to govern the use of AI using a human-in-the-loop approach. The dispersion of business analysis expertise among stakeholders such as product owners, project managers, and technical leads presents another challenge for adopting AI-supported tools, because they may not have a formal understanding of business analysis best practices [7].

In the enterprise, documented use of Generative AI tools is spreading rapidly, and organizations must proactively and systematically resist unchecked use to avoid locally inconsistent practices [**8**]. Implementing AI-generated requirements lacks a domain-specific evaluation framework; thus, there is no way to determine whether a requirement is of sufficient quality for requirements engineering and audit teams, regulatory authorities, or quality assurance processes. This paper addresses these shortcomings by introducing a framework for implementing prompt-engineered LLMs in the BSA requirements workflow in regulated enterprise environments. This paper has three important contributions. Second, the study demonstrates a systematic, prompt-driven method for documenting JAD session notes and producing BRDs and user stories with acceptance criteria from the perspectives of investment management, banking, and public-sector personnel in health and human services [9].

The methodology proposes a family of domain-specific prompt templates to guide LLMs in ensuring that the high-level business objective and the system-level functional statements are traceable to each other,

represent the business rules, and include acceptance criteria. Second, the researchers created an evaluation framework that compares requirements provided by LLMs with analyst-provided baselines, accounting for measurable factors such as ambiguity, completeness, traceability, and documentation time savings [10]. This framework will help practitioners ensure the rigor of the original, human-created document carries over to the AI-assisted version, especially in regulated environments. Third, the research suggests a human-in-the-loop governance strategy in the guidelines for when to use AI review and when to use validation and sign-off, so AI-assisted requirements can be safely used in audit-sensitive environments. This minimizes the chances of organizational and compliance issues regarding the use of AI-generated tools in the supervisory context, as it specifies who is responsible for the content generated by the tool.

This study consists of several chapters. Chapter 2 introduces a literature review on requirements engineering, NLP, LLMs, and human-AI governance. Chapter 3 outlines the research problem. Chapter 4 covers the experimental process for the proposed methodology, including dataset selection, experimental conditions, prompt design, manual annotation, and analysis. Chapter 5 describes the evaluation plan and planned measures. Chapter 6 ends with suggested practical recommendations and directions for future research.

2. LITERATURE REVIEW

2.1 Requirements Engineering Foundations and Quality

Requirements Engineering is based on structured Elicitation, Documentation, and quality assurance practices. The BABOK Guide identifies highly effective methods analysts can use to elicit requirements from stakeholders, including interviews, workshops, observation, and collaborative analysis. ISO/IEC/IEEE 29148:2018 documents requirements engineering processes, documentation, validation, and maintenance requirements [11]. The literature also focuses on the form and clarity. To measure the agility of user stories, Lucassen et al. (2016) developed the Quality User Story framework, which relies on syntactic and semantic properties of user stories and can be measured automatically [10].

Femmer et al. (2017) created Requirements Smells, a lightweight Quality Assurance (QA) technique to detect potentially ambiguous or weak wording in requirements specifications [12]. Arora et al. (2015) demonstrated NLP's ability to perform requirements conformance checks to avoid repeated manual checks [13]. Together, these studies provided the basis for the metrics adopted in this study to assess the completeness, clarity, consistency, and structure of the LLMs' generated requirements artifacts.

2.2 Natural Language Processing for Requirements Engineering

Much of the research in the NLP4RE framework before the advent of LLMs focused on analysis rather than complete document creation. Zhao et al. (2021) conducted a systematic mapping study and found studies on quality defect detection, requirements classification, concept extraction, and traceability link recovery, but with limited industrial evaluation and tool availability [14]. Template checking and requirements smell detection are examples of applying linguistic automation tools; however, they do not eliminate domain judgement [15]. Generally, these depend on pre-existing requirements text and use task-specific rules, resources, or models. As such,

they offer a good starting point for quality assurance, but not a direct answer to how generative models can build a complete BRD/user story from an unstructured stakeholder note in regulated enterprise scenarios.

2.3 Large Language Models in Requirements and Software Engineering

In recent years, research on LLMs for software and requirements engineering has increased. Hou et al. (2024) conducted a Systematic Literature Review (SLR) of current applications of LLMs in Software Engineering (SE) and found Requirements Engineering (RE) to be an active area of study, with numerous open questions related to evaluation, reliability, data, and practical application [16]. Arora et al. (2024) discuss how LLMs can assist with elicitation, analysis, specification, and validation, and note the need for cautious human evaluation and piloting [17]. Ronanki et al. (2023) analysed requirements generated by ChatGPT. They compared them with human expert-generated requirements and found that both the positive and negative quality aspects deserve further exploration [18].

White et al. (2023) and Yamani et al. (2025) compared the results of three LLMs using the Quality User Story framework, highlighting the value of comparing models and using structured quality scoring when generating user stories [19; 20]. These studies suggest that AI could be used for requirements work. However, they do not show that LLM enterprise artifact generation is more effective or efficient than manual generation by analysts.

2.4 Human-AI Collaboration and AI Governance

Studies into collaboration with AI show that clear, corrective, and accountable feedback is important in AI-supported processes. Amershi et al. (2019) propose guidelines for human interaction that focus on helping users understand the system's capabilities and limitations, as well as providing tools to fix inconsistencies that may occur in the AI's answers [21]. In regulated or audit-sensitive processes, artifacts such as the NIST AI Risk Management Framework provide a framework for managing AI risks, encompassing governance, measurement, and risk management activities, and are applicable across sectors [22].

The requirements engineering process is also affected by human factors, including communication, domain knowledge, and the analyst's experience. In an agile environment, the business analyst breaks down, clarifies, and adds detail to user stories for development and testing teams [23]. This evidence suggests a human-in-the-loop scenario, with the LLM providing a draft that analysts review for coherence and alignment with legal, regulatory, and business constraints and objectives, and then approve.

2.5 Research Gap

The literature is disorganized among these people. Requirements engineering research has established the foundation for what good requirements should look and feel like, but not yet in the age of generative AI. NLP4RE work automates tasks, not necessarily full documents, such as classification or ambiguity detection. Recent LLM studies are promising for elicitation/specification. However, they primarily use academic datasets, student participants, and software projects in general, and seldom validate their output against analyst-developed baseline documentation using commonly known criteria. Human-AI governance frameworks, in contrast, are broad and are not specific to requirements work. What remains to be done is a practitioner-informed, full-service way that uses real and actual JAD session notes,

converts notes into structured BRDs and user stories, calibrates them for level of ambiguity, completeness, and their ability to have traceability, and also involves human-in-the-loop to review the user stories specifically for regulated industries, including financial services and public-sector health and human services.

3. PROBLEM STATEMENT

The pivotal role in enterprise information systems development is occupied by the Business Systems Analyst (BSA), who elicits the stakeholders' needs, analyzes the enterprise processes, and converts the needs into requirements that can guide the technical implementation [24]. Requirements engineering frameworks are well-established, and structured elicitation methods such as the Joint Application Development (JAD) session are in use. However, translating the unstructured artefacts created in the elicitation session into a formal requirements document remains largely a manual, analyst-driven process [25]. Quality issues with manual conversion are systematic and affect all project results, touching many stakeholder groups and organizational areas.

Business Systems Analysts consistently gather stakeholder feedback through stakeholder workshops, interviews, and co-creation sessions, then sort through unstructured or semi-structured notes to identify Business Requirements Documents (BRDs), Functional Requirements Documents (FRDs), user stories, and Requirements Traceability Matrices (RTMs) [26]. The transformation process is analytically complex, and interpretation, experience, and workload will vary among data analysts. In natural-language requirements, ambiguities, incomplete business-rule specifications, and missing artifact traces may cause defects in downstream development, scope creep, or rework during development and testing [27]. In regulated domains such as investment management, banking, and public-sector health and human services, insufficient traceability or opaque requirements may be interpreted as audit review results and are a risk in both cases; when gaps in requirements travel to production systems, end users may face system mistakes and service disruptions in the process of applying for benefits or interacting with sponsors or retailers through financial services.

Several stakeholder groups are affected by the problem. Business Systems Analysts spend considerable time producing manual documentation, and the work product depends on the analyst's experience level and any concurrent workload requirements [28; 29]. Business representatives and subject-matter experts may misunderstand requirements at the outset, or multiple interactions may be needed to clarify them, further extending the project timeframe. Development and quality assurance resources build and evaluate products based on poorly defined specifications, leading to late-detected defects when they are most expensive to remediate [30]. Some requirement-related issues are not found until late in development projects and require design changes and rework, causing scope creep, delays, and budget overruns for the project management agencies and project managers. Regulated industries have gaps in compliance requirements, creating traceability issues for compliance functions, audit trails, and regulatory reporting, as well as end-user deficiencies from system errors and failures caused by missing requirements.

Manually induced requirements transformation (MRT) persists due to intertwined challenges. Requirements work largely relies on human judgment to express

stakeholder intent, business rules, and regulatory and domain constraints. Previous generations of rule-based and natural language processing tools could handle only limited subtasks such as ambiguity detection and requirements classification, but never generate a contextually relevant complete requirements document [31]. Empirical studies have not properly assessed generative AI systems that can produce complete requirements artifacts, which have been widely available only since 2023 and have already been used informally in practice [32]. While most research uses academic or synthetic datasets, Real-world JAD notes and BRDs for financial or public-sector projects are confidential and proprietary and therefore cannot replicate the complexity and domain-oriented nature required for enterprise artifacts. Another challenge is the lack of a systematic method for comparing AI-generated requirements with the analytic baseline written by requirements analysts, one that is valid for assessing criteria such as ambiguity, completeness, and traceability [33]. In regulated sectors, organizations are rightly concerned about using AI-powered requirements practices without integrating governance and controls into the human-in-the-loop, and existing practices are informal, unvalidated, and not part of quality assurance.

Success would be a repeatable, validated methodology to obtain requirements via a Large Language Model (LLM) using text input, reviewed by a structured analyst, and producing requirements of equal or greater quality than manually created requirements, with a measurable gain in production time. AI-assisted BRDs and user stories would also deliver lower ambiguity-indicator scores and higher completeness than baseline created by analysts alone, with scores measured against existing enablers for Quality Requirements [34]. All requirements would be bi-directionally traceable to test cases to support audit and regulatory requirements.

Stalling time on documentation turnaround would be reduced, allowing the Business Systems Analyst to work more efficiently on higher-value documentation-related tasks, including stakeholder facilitation, business process analysis, and solution validation. A human-in-the-loop (HITL) governance model would clearly define the exception cases in which analysts must read and sign off, allowing safe co-development in environments where regulations and audits must be complied with [35]. The methodology would be transferable to other providers, such as financial service providers and the public sector, including health and human service providers, and robust to changes in the business context, regulation, and stakeholder structure. To achieve this outcome, it would be necessary to validate it through an empirical study that employs existing artefacts or quality measures from the enterprise and compares them with an existing manual method to assess its effectiveness and usability within the enterprise.

4. METHODOLOGY

4.1 Study Design and Dataset Selection

This study will use a controlled repeated-measures design, with three conditions using the same source material. The dataset will comprise at least thirty anonymized recordings of JAD sessions, with roughly equal sample sizes drawn from investment management, banks, and public sector health human services organizations. An acceptable record should include enough stakeholder context to explain business aspects such as business requirements, business rules, user stories, nonfunctional constraints, and acceptance criteria. Records with incomplete information for purposes of blinded

completeness recording and records that cannot be reliably anonymized will be excluded. To estimate within-case variance, the study will complete a small pilot and use an a priori power analysis to determine the final sample size. A data set register will document domains, note length, session length (if available), and whether the analyst wrote baseline artifacts.

4.2 Experimental Conditions and Prompt Protocol

Each JAD will be analyzed separately in an analyst-only baseline condition, an LLM-only condition, and an LLM-plus-human-review condition. Full participation will be achieved with at least three experienced BSAs across analyst conditions, with case assignments rotated to minimize individual analyst and learning effects. The study will evaluate the effectiveness of at least three existing LLMs from different suppliers or families. The study will provide each model with the same anonymized notes, BRD, output user story schema, and version-controlled prompt template, based on the explicit role instruction, the grounding rules in the notes, and the structured output constraints [36]. They will document the model name, prompt version, date of use, model version, decoding settings, and run identifier. LLM-only outputs will not be edited before scoring. In the human review condition, a BSA who did not author the analyst-only baseline for that case will review the LLM draft, correct unsupported content, resolve ambiguity, complete missing information that is supported by the source notes, and approve the final artifact.

4.3 Blinded Review and Scoring

Three reviewers with requirements engineering experience will score the artifacts using a predefined rubric. Review packets will conceal the generation condition, model identity, and producing analyst as far as practical. Reviewers will have access to the source notes and the artifact being scored so that completeness and traceability can be judged against the same evidence. The rubric will draw on ISO/IEC/IEEE 29148, the Quality User Story framework, Requirements Smells, and automated template checking [37]. Disagreements will be resolved only after independent scoring is complete, and inter-rater agreement will be reported.

4.4 Measures and Statistical Analysis

Documentation time will be measured in minutes from release of the source notes to submission of the final artifact. Ambiguity will be reported as confirmed ambiguity or requirements smell flags per requirement. Completeness will be the proportion of required rubric elements supported by the source notes. Traceability will be the proportion of generated requirements with valid links to source JAD statements and, where test cases are available or derived for the study, valid downstream test links [38]. The primary comparisons will evaluate whether, and by how much, using an LLM to facilitate documentation reduces documentation time without materially affecting completeness or traceability or increasing ambiguity, relative to the analyst-only baseline. The study will compare conditions using mixed-effects models in which source session and analyst will both be random factors. If model assumptions are violated, the study will use non-parametric alternatives. The study will report effect sizes and 95% confidence intervals alongside significance testing.

4.5 Data Governance and Reproducibility

Access to the model will be granted after anonymising all source artefacts. The study will de-identify or alter protectively

identifiable information, secrets, account information, and other information with consistent placeholders. The study will have access only to the tools it is allowed to use to process the data classified in the study and will not use external model providers where the terms of use prohibit training on the enterprise data provided for the study. The study will version prompts and metadata for all scoring rubrics; analyst edits and traceability links will ensure auditability and replication. This study does not represent requirements commonly developed by artificial intelligence without verification by a competent human verifier. These controls ensure the assessment aligns with human-AI interaction and risk-management requirements.

5. EVALUATION PLAN

5.1 Evaluation Metrics and Analysis Plan

As measurable data on the system's performance are unavailable, this section describes its impact and assessment. Tables 1 and 2 show the study conditions and define and summarize the criteria required to substantiate the validity of the study results presented; these tables do not present numerical results. Three regulated scenarios will use the same input (JADs) and will be evaluated using the following outputs: LLM-only, analyst-only, and LLM-and-human. The primary research hypothesis is that the documentation phase will be completed at a higher rate with LLM-humans than with the analyst-only baseline, without harming requirements quality. The study will examine performance variations between domains, analysts, and LLMs. The study will use these data to evaluate these hypotheses during the next blinded scoring stage.

Table 1 shows three evaluation configurations: LLM-only, analyst-only, and LLM plus human checks. The same anonymised JAD notes are used for each condition. The table compares the generation methods, degree of human involvement, planned measures, and study status. Outcomes measure and report documentation time, ambiguity level, completeness, traceability, review time, and edit rate for comparison.

Table 1: Planned comparison across evaluation conditions.

Approach	Input	Generation	Human Role	Planned Measures	Status
Analyst-Only Baseline	Same anonymized JAD notes	Manual BRD and user story authoring	Full authorship	Time, ambiguity, completeness, RTM coverage	To be measured
LLM-Only	Same anonymized JAD notes	Same prompt and schema across selected LLMs	None before scoring	Same core measures	To be measured
LLM + Human Review	Same anonymized JAD notes	LLM draft followed by analyst revision	Required review and approval	Core measures plus review time and edit rate	To be measured

Table 2 provides the metrics and validation procedures for the evaluations. Documentation time is measured in minutes from when source notes are released to when source notes are submitted. Ambiguity is the confirmed ambiguity or requirement smell flags. Completeness measures how many required rubric elements the notes support, while traceability measures links between notes and a valid source, as well as test links and a downstream test, using a blinded review process.

Table 2: Planned metric definitions and validation procedures.

Metric	Operational Definition	Planned Validation
Documentation Time	Minutes from release of source notes to submission of the final artifact	System timestamps and analyst logs
Ambiguity	Confirmed ambiguity or requirements smell flags per requirement	Automated screening plus blinded human confirmation
Completeness	Proportion of required rubric elements supported by the source notes	Blinded scoring against the source notes and artifact template
Traceability	Proportion of requirements with valid source links and downstream test links when available	RTM audit by blinded reviewers

Outcome measures focus on documentation time, ambiguity, completeness, and traceability. Time study logs will document time during the study. Ambiguity will be assessed using automated screening, with predefined language defects confirmed through anonymous human review. Completeness will be assessed based on the source-grounded elements listed in the rubric, and RTM coverage will be checked from source notes to generated requirements to test cases if test cases are available and/or derived for study [39]. The study will report final experimental results as interrater agreement, effect sizes, confidence intervals, and statistical comparisons. Until such experiments are carried out, none of the numbers in those tables can be taken as experimental results.

5.2 Planned Evaluation Workflow

The planned evaluation system should be sequential to support accurate, objective comparisons across all study situations. Preparation and anonymization will protect JAD inputs from privacy violations and loss of requirement content. The study will then use the same inputs to produce analyst-generated and LLM assistant outputs, which humans will check. It will also then anonymise and mark these requirement products to reduce reviewer bias. The study will then assess the time and quality of the documentation for each condition, including completeness, ambiguity, and traceability. The research will compare and present the results of each technique and use statistical tests to determine whether differences between techniques are statistically significant in quality and efficiency. Figure 1 shows the evaluation process, which starts with preparing the JAD inputs, calculating the

metrics, generating the artifacts, plotting the data, conducting statistical tests, and reporting the results.

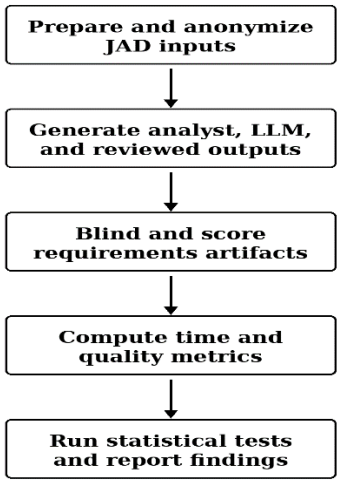


Figure 1: Anticipated evaluation from dataset creation to statistical comparison and reporting.

6. CONCLUSION & FUTURE WORK

That study suggests a systematic approach to formalizing JAD session notes as a Business Requirements Document or user story using purpose-built LLM prompts. It will be useful for the Business Systems Analyst. A long-standing challenge in requirements engineering is capturing and transferring evidence collected through collaborative requirements engineering methods into the requirements document in a way that satisfies organisational and regulatory requirements, which a human-in-the-loop approach can achieve. Generative AI tools are now at analysts' fingertips, and the paper provides a starting point for systematically incorporating them not to supplant what they currently do, but to use them as a tool for improvement.

The methodology is not only an approximation of potential efficiency gains. For regulated sectors like financial services, public sector health systems, and more, it can help compliance specialists determine which compliance components require human intervention and ingenuity before standardising and speeding up the remainder with AI capabilities, without jeopardising audit readiness. A trajectory is sketched for the adoption process to discuss within the group how AI support will be integrated into their thinking and how the analyst will stay at the heart of all value, validation, and stakeholder involvement.

The approach also reflects ongoing discussions about the evolving industry and professional practice expectations for BSAs, and how BSAs' character and role are shifting from documentation to AI-assisted performance, governance, curation, and interpretation. Three research directions can be considered priorities for this work. The proposed experiment should also use de-identified JAD notes, multiple professional scorers, multiple LLMs, double-blinded scorers (humans), and stipulated statistical comparisons. Further studies could then examine downstream impacts on defects, change request/rework cycles, and delivery time.

Additionally, assessing LLMs designed for particular regulatory or industry use cases or

organizations, such as fine-tuned versions based on industry-specific or organization-specific templates/business rules and the possible use of additional context, such as regulatory bodies, will assist in solving data privacy concerns and enhancing the precision of such models in specific use cases. Finally, integrating the methodology with more end-to-end automation by automatically generating a Requirements Traceability Matrix (RTM), automatically generating test cases, and automatically assessing change request impacts will provide a comprehensive lifecycle automation framework with ongoing connectivity to tools such as JIRA and Azure DevOps, complementing the third role, the systems engineer analyst, as the orchestrator of AI-supported systems engineering.

REFERENCES

1. T. Ritter and C. L. Pedersen, "Digitization capability and the digitalization of business models in business-to-business firms: Past, present, and future," Industrial Marketing Management, vol. 86, pp. 180–190, 2020. DOI: 10.1016/j.indmarman.2019.11.019.
2. P. A. Laplante and M. Kassab, Requirements Engineering for Software and Systems, 4th ed., CRC Press, 2022. DOI: 10.1201/9781003129509.
3. International Institute of Business Analysis, A Guide to the Business Analysis Body of Knowledge (BABOK Guide), Version 3, IIBA, 2015. ISBN 9781927584026.
4. P. Sinha and H. K. Jain, "Reusing business components and objects for modeling business systems: The influence of decomposition characteristics and analyst experience," Journal of Systems and Software, vol. 131, pp. 550–569, 2017. DOI: 10.1016/j.jss.2016.07.036.
5. H. Cheng, J. H. Husen, Y. Lu, T. Racharak, N. Yoshioka, N. Ubayashi, and H. Washizaki, "Generative AI for Requirements Engineering: A Systematic Literature Review," Software: Practice and Experience, vol. 56, no. 2, pp. 141–170, 2026. https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.70029
6. N. Marques, R. R. Silva, and J. Bernardino, "Using ChatGPT in Software Requirements Engineering: A Comprehensive Review," Future Internet, vol. 16, no. 6, Art. no. 180, 2024. https://www.mdpi.com/1999-5903/16/6/180
7. Harris and C. Clark, "Understanding the Diffusion of Business Analysis Responsibilities," Journal of Computer Information Systems, vol. 64, no. 2, pp. 278–288, 2024. https://www.tandfonline.com/doi/pdf/10.1080/08874417.2023.2193964?casa_token=i_WyeNJeCeUAAAAA:v3JLLAwJpH9AY8FlK3KUgJ8yPj08pYjSEWWYnj7L5F4vb2MliQ-QkfE1PXWciquaeKYxpkKRB4s93G4
8. M. Silic, D. Silic, and K. Kind Trüller, "From Shadow IT to Shadow AI: Threats, risks and opportunities for organizations," Strategic Change, pp. 1–16, 2025. https://onlinelibrary.wiley.com/doi/pdf/10.1002/jsc.2682?casa_token=TKb9ncwsJ6sAAAAA:7Q0MCEczHLaE6tMQTcpLcpVDvfY-gMvteq6ynp-_Q32TP6dPb_gKTgrj-_5MlALvKTP9j-hi7NroWe0
9. D. Hidellaarachchi, J. Grundy, R. Hoda, and K. Madampe, "The effects of human aspects on the Requirements Engineering process: a systematic literature review," IEEE Transactions on Software Engineering, vol. 48, no. 6, pp. 2105–2127, 2022. DOI: 10.1109/TSE.2021.3051898.

10. G. Lucassen, F. Dalpiaz, J. M. E. M. van der Werf, and S. Brinkkemper, "Improving agile requirements: the Quality User Story framework and tool," Requirements Engineering, vol. 21, pp. 383–403, 2016. DOI: 10.1007/s00766-016-0250-x.
11. ISO/IEC/IEEE 29148:2018, Systems and software engineering — Life cycle processes — Requirements engineering, ISO, 2018.
12. H. Femmer, D. Méndez Fernández, S. Wagner, and S. Eder, "Rapid quality assurance with Requirements Smells," Journal of Systems and Software, vol. 123, pp. 190–213, 2017. DOI: 10.1016/j.jss.2016.02.047.
13. C. Arora, M. Sabetzadeh, L. Briand, and F. Zimmer, "Automated Checking of Conformance to Requirements Templates Using Natural Language Processing," IEEE Transactions on Software Engineering, vol. 41, no. 10, pp. 944–968, 2015. DOI: 10.1109/TSE.2015.2428709.
14. L. Zhao, W. Alhoshan, A. Ferrari, K. J. Letsholo, M. A. Ajagbe, E.-V. Chioasca, and R. T. Batista-Navarro, "Natural Language Processing for Requirements Engineering: A Systematic Mapping Study," ACM Computing Surveys, vol. 54, no. 3, Art. 55, 2021. DOI: 10.1145/3444689.
15. K. Großer, A. S. Ahmadian, M. Rukavitsyna, Q. Ramadan, and J. Jürjens, "Benchmarking requirement template systems: Comparing appropriateness, usability, and expressiveness," Requirements Engineering, vol. 29, pp. 481–522, 2024. https://link.springer.com/content/pdf/10.1007/s00766-024-00427-0.pdf
16. X. Hou et al., "Large Language Models for Software Engineering: A Systematic Literature Review," ACM Transactions on Software Engineering and Methodology, vol. 33, no. 8, Art. 220, 2024. DOI: 10.1145/3695988.
17. C. Arora, J. Grundy, and M. Abdelrazek, "Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs," in Generative AI for Effective Software Development, Springer, 2024, pp. 129–148. DOI: 10.1007/978-3-031-55642-5_6.
18. K. Ronanki, C. Berger, and J. Horkoff, "Investigating ChatGPT's Potential to Assist in Requirements Elicitation Processes," in 2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pp. 354–361, 2023. DOI: 10.1109/SEAA60479.2023.00061.
19. J. White et al., "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," in Proceedings of the 30th Conference on Pattern Languages of Programs, 2023. DOI: 10.64346/PLoP2023p05.
20. Yamani, M. Baslyman, and M. Ahmed, "Leveraging LLMs for User Stories in AI Systems: UStAI Dataset," in Companion Proceedings of the 33rd ACM Symposium on the Foundations of Software Engineering, 2025. DOI: 10.1145/3727582.3728689.
21. S. Amershi et al., "Guidelines for Human-AI Interaction," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Paper 3, pp. 1–13, 2019. DOI: 10.1145/3290605.3300233.
22. E. Tabassi, Artificial Intelligence Risk Management Framework (AI RMF 1.0), NIST AI 100-1, National Institute of Standards and Technology, 2023. DOI: 10.6028/NIST.AI.100-1.
23. D. D. Gregorio, "How the Business Analyst supports and encourages collaboration on agile projects," in 2012 IEEE International Systems Conference, 2012. DOI: 10.1109/SysCon.2012.6189437.

24. N. Evans and A. Fernando, "The Enterprise 4.0 Business Analyst: Digital Business Transformation Requires a New Mindset," Knowledge and Process Management, vol. 32, no. 2, pp. 83–96, 2025. https://onlinelibrary.wiley.com/doi/pdf/10.1002/kpm.1799
25. S. Eltahier, S. M. Al-Ghuribi, M. A. Mohammed, and I. Saeed, "Automated Software Requirements Elicitation: A Systematic Mapping Study," Information, vol. 17, no. 8, Art. no. 777, 2026. https://www.mdpi.com/2078-2489/17/8/777
26. X. Franch, C. Palomares, C. Quer, P. Chatzipetrou, and T. Gorschek, "The state-of-practice in requirements specification: an extended interview study at 12 companies," Requirements Engineering, vol. 28, pp. 377–409, 2023. https://link.springer.com/content/pdf/10.1007/s00766-023-00399-7.pdf
27. G. Bishop, S. Chakraborty, H. Zhou, J. Dehlinger, L. Deng, J. Lin, and B. Kist, "On-demand generation of high-quality software engineering datasets using large language models and ontologies," Automated Software Engineering, vol. 33, Art. no. 73, 2026. https://link.springer.com/content/pdf/10.1007/s10515-026-00617-w.pdf
28. M. A. Umar and K. Lano, "Advances in automated support for requirements engineering: A systematic literature review," Requirements Engineering, vol. 29, pp. 177–207, 2024. https://link.springer.com/content/pdf/10.1007/s00766-023-00411-0.pdf
29. J. Frattini, D. Fucci, D. Méndez, R. Spínola, V. Mandić, N. Taušan, M. O. Ahmad, and J. Gonzalez-Huerta, "An initial theory to understand and manage requirements engineering debt in practice," Information and Software Technology, vol. 159, Art. no. 107201, 2023. https://www.sciencedirect.com/science/article/pii/S0950584923000551
30. J. Perera, E. Tempero, Y. C. Tu, and K. Blincoe, "Modelling the quantification of requirements technical debt," Requirements Engineering, vol. 29, no. 4, pp. 421–458, 2024. https://link.springer.com/content/pdf/10.1007/s00766-024-00424-3.pdf
31. G. Paiva, E. D. Canedo, and G. P. Rocha Filho, "From issue titles to requirements: An empirical study of large language models and prompt engineering strategies," Requirements Engineering, vol. 31, Art. no. 7, 2026. https://link.springer.com/content/pdf/10.1007/s00766-026-00462-z.pdf
32. K. Stein, A. Mirzai, J. Axmann, and T. Vietor, "Integrating the capabilities offered by large language models into the requirements engineering process," Digital Engineering, vol. 10, Art. no. 100098, 2026. https://www.sciencedirect.com/science/article/pii/S2950550X26000117
33. Hemmat, M. Sharbaf, S. Kolahdouz-Rahimi, K. Lano, and S. Y. Tehrani, "Research directions for using LLM in software requirement engineering: A systematic review," Frontiers in Computer Science, vol. 7, Art. no. 1519437, 2025. https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2025.1519437/pdf

34. R. Santos, G. Freitas, I. Steinmacher, T. Conte, A. C. Oran, and B. Gadelha, "User Stories: Does ChatGPT Do It Better?," in Proc. 27th Int. Conf. Enterprise Information Systems (ICEIS), vol. 2, Porto, Portugal, 2025. https://www.scitepress.org/Papers/2025/133655/133655.pdf
35. L. Zhu, Q. Lu, M. Ding, S. U. Lee, and C. Wang, "Designing meaningful human oversight in AI," AI and Ethics, vol. 6, Art. no. 286, 2026. https://link.springer.com/content/pdf/10.1007/s43681-026-01147-7.pdf
36. M. Krishna, B. Gaur, A. Verma, and P. Jalote, "Using LLMs in Software Requirements Specifications: An Empirical Evaluation," in Proc. 2024 IEEE 32nd Int. Requirements Engineering Conf. (RE), Reykjavik, Iceland, 2024, pp. 475–483. https://arxiv.org/pdf/2404.17842
37. E. Wolf, A. Trendowicz, and J. Siebert, "Quality assessment of software requirements using artificial intelligence methods: A systematic literature review," Information and Software Technology, vol. 191, Art. no. 107979, Mar. 2026. https://publica-rest.fraunhofer.de/server/api/core/bitstreams/1cb96a79-9ef1-4b55-a777-d05dc1951972/content
38. J. Mucha, A. Kaufmann, and D. Riehle, "A systematic literature review of pre-requirements specification traceability," Requirements Engineering, vol. 29, pp. 119–141, 2024. https://link.springer.com/content/pdf/10.1007/s00766-023-00412-z.pdf
39. H. Li and L. Zheng, "Enhancing Requirements via Structured Formalization and Process-State Consistency Validation: An LLM-Assisted Test-Driven Framework," IET Software, vol. 2025. https://ietresearch.onlinelibrary.wiley.com/doi/pdfdirect/10.1049/sfw2/6714956

Conflict of Interest: No Conflict of Interest
Source of Funding: Author(s) Funded the Research
How to Cite: Saudagar, A. (2026). Human-in-the-Loop Evaluation of LLM-Assisted Requirements Engineering in Regulated Enterprises. *Frontiers in Emerging Technology, 2*(6), 1-14.