LLM-Assisted INVEST Evaluation and Improvement of User Stories: An Industrial Replication Study

Erika Hernández-Agüero[1,2], Christian Quesada-López[1,2], José P. Chaves-Sánchez[2]

[1]Universidad de Costa Rica,
Posgrado en Computación e Informática, Escuela de Ciencias de la Computación e Informática – San José– Costa Rica

[2]Universidad Estatal a Distancia
Dirección de Tecnología de Información y Comunicaciones – San José– Costa Rica

{erika.hernandezaguero,cristian.quesadalopez}@ucr.ac.cr

{ehernandez,cquesadal,jpchaves}@uned.ac.cr

Abstract. *The specification and maintenance of high-quality user stories are critical in agile software development, yet they are often hindered by natural-language ambiguity, evolving business requirements, and the effort required for manual backlog refinement in industrial settings. This paper investigates the use of large language models (LLMs), specifically GPT-5.1, to support the automated evaluation and improvement of user-story quality using the INVEST framework. Building on prior expert-based assessments, we propose a human-in-the-loop procedure that combines LLM automation with requirements-engineering expertise. We conduct an industrial replication study using 49 real user stories from a scholarship-management system, preserving the evaluation–improvement–reevaluation design of prior expert-based work. Results show alignment between GPT-5.1 and expert judgments, particularly after an evaluation–improvement–reevaluation cycle, with strong semantic agreement and convergence in key INVEST dimensions. GPT-5.1 assigns slightly lower scores than experts for Independent, Negotiable, Estimable, and Small, with moderate monotonic correlations ($\rho \approx 0.53–0.65$). After the improvement cycle, expert medians reach 5 across all INVEST criteria, and GPT-5.1 converges strongly on Valuable and Testable while remaining more conservative on Independent and Small; semantic agreement exceeds 85–90% across most dimensions. These findings indicate that GPT-5.1 could reduce manual assessment effort, reinforce structural quality, and support consistent requirements evaluation, while highlighting the complementary role of human oversight in industrial requirements-engineering workflows.*

Keywords: requirements engineering; user stories; large language models; automated quality assessment; INVEST framework; human-in-the-loop; industrial study

1. Introduction

In organizational contexts, seamless integration between business and software development is essential for effective requirements management, as it enables the delivery of products aligned with stakeholder needs and strategic objectives [Bourque and Fairley 2014]. Continuous Software Engineering provides a holistic perspective on the

workflow that connects business strategy, software development, and operations, an approach commonly referred to as BizDevOps [Fitzgerald and Stol 2017; Hernández et al. 2024]. This discipline encompasses key elements such as continuous improvement, innovation, the adoption of agile methodologies in software development, and process automation [Fitzgerald and Stol 2017; Bosch 2014]. These approaches require significant efforts toward automating processes that directly involve the business (Biz) domain [Fitzgerald and Stol 2017]. Within this context, Natural Language Processing for Requirements Engineering (NLP4RE) has emerged as a research area that applies NLP techniques to address challenges in requirements engineering, including requirements generation, elicitation, analysis, validation, and management when expressed in natural language [Marques, Silva, and Bernardino 2024; Belzner, Gabor and Wirsing 2024]. However, it remains necessary to evaluate these approaches in real-world projects to determine their potential to support such activities in industrial settings.

Recent studies investigate the use of Large Language Models (LLMs) to support context-aware requirements generation. In these works, LLM-generated requirements are assessed against expert criteria using frameworks such as INVEST. The findings highlight opportunities for improvement aimed at reducing model hallucinations, establishing best practices for prompt construction [Krishna et al. 2024; Zhang et al. 2024], and strengthening methodological rigor to ensure study replicability [Baltes et al. 2025]. While prior research has examined how LLMs assess user stories under INVEST, there is growing interest in closing the quality loop through evaluation–improvement–reevaluation cycles that simulate interactions between key roles, such as Requirements Engineers (RE) and Product Owners (PO), using prompt-based interventions to guide rewriting, negotiation, and refinement [Zhang et al. 2024; Hernández et al. 2025; Hernández et al. 2026].

This paper presents a dependent exact replication with differentiated context of a prior empirical design, following the replication taxonomy of [Shull et al. 2008], conducted in an industrial agile setting to evaluate and improve user-story quality using LLMs. This study adopts a human-in-the-loop approach [Baltes et al. 2025; Hernández et al. 2026], in which LLM-assisted evaluation and improvement are performed under continuous expert supervision. The researchers actively validate model-generated changes and iteratively calibrate prompts and contextual rules, ensuring domain alignment, preservation of business value, and reliability in the requirements evaluation process. The replication preserves the INVEST instrument, the calibrated prompting protocol, and the evaluation–improvement–reevaluation cycle from prior work, while deliberately varying the application domain, dataset, and LLM configuration. Our goal is to assess the transferability and robustness of the approach under controlled contextual modifications and continuous expert oversight in a real project setting. More specifically, the study pursues three connected aims: to examine baseline alignment between GPT-5.1–assisted INVEST evaluations and expert judgment, to assess changes in user-story quality after the evaluation–improvement–reevaluation cycle, and to identify whether the model applies systematically stricter criteria in particular INVEST dimensions.

Building on this foundation, we replicate a three-stage empirical design (evaluation, improvement, and reevaluation) [Hernández et al. 2026], conducted in an industrial agile context, to assess and improve user-story quality using an LLM. The study involves three expert requirements engineers and a single advanced LLM (GPT-5.1). The design reproduces the evaluation–improvement–reevaluation cycle, modifying only the

project context and the set of user stories while preserving the same calibrated prompting protocol to operationalize quality assessment under the INVEST framework. User stories are first evaluated by both experts and the LLM, then refined through an improvement process that integrates simulated RE–PO roles, typological classification, penalty rules, and domain-specific constraints to preserve scope and business value. This process enforces a standardized user-story format ("As a <role>, I want <capability> so that <benefit>") and measurable acceptance criteria (Given–When–Then), while modeling RE–PO interaction to validate proposed changes. Finally, the improved stories are reevaluated by the same LLM and the three experts under replication conditions, enabling estimation of the intervention's effect on requirements quality and its alignment with expert judgment under INVEST.

2. Related work

Several studies have explored the automation of requirements' engineering tasks, with increasing emphasis on the use of Large Language Models (LLMs) to support different phases of the software development lifecycle. This line of research addresses requirements' document generation and validation, elicitation support, user-story refinement in agile environments, and assistance in activities such as system design, code generation, and testing.

[Krishna et al. 2024] analyzed the use of GPT-4 and CodeLlama for generating and validating Software Requirements Specifications (SRS), comparing model-produced documents with those created by junior software engineers, considering attributes such as clarity, consistency, and completeness. The authors report reductions in authoring time and comparable levels of perceived quality. Complementarily, [Ronanki et al. 2023] studied the use of ChatGPT in requirements elicitation tasks, evaluating dimensions such as abstraction, atomicity, consistency, correctness, clarity, comprehensibility, and feasibility, observing variations in model performance across different attributes.

In agile settings, [Zhang et al. 2024] introduced ALAS, an LLM-based system (GPT-3.5/GPT-4) for collaborative user-story refinement using INVEST, reporting clearer and better-aligned stories but occasionally increased complexity. [Belzner et al. 2024] surveyed LLM applications across software-engineering tasks—requirements, design, testing, and code generation—highlighting productivity gains alongside integration challenges and context dependency. Methodologically, [Wang et al. 2025] evaluated LLMs as automated judges in software-engineering tasks, analyzing alignment with human assessments, while [Liu et al. 2023] proposed G-EVAL, a structured-prompt evaluation framework that improves correlation with human judgments. Both works emphasize the influence of prompt design and risks of model bias, recommending human supervision. From an industrial viewpoint, [Santos et al. 2025] examined practitioner adoption of LLMs in requirements engineering, finding growing use for user-story generation but also concerns regarding context availability, model dependency, and data privacy. [Hernández et al. 2025] presented an empirical study in an industrial project evaluating user-story quality using LLMs under the INVEST framework, comparing model scores with assessments by requirements engineering experts. The study reports high levels of semantic agreement across several INVEST attributes, as well as variations in criteria such as Small, Negotiable, and Estimable, and a dependence on story structure and the presence of acceptance criteria. Taken together, these studies describe diverse uses of LLMs in software and requirements engineering, primarily focused on isolated

evaluation or artifact generation. However, there remains scope for investigating approaches that systematically integrate evaluation–improvement–reevaluation cycles under human supervision, as well as their specific application to improving user stories in industrial contexts using INVEST. This work is situated in this line of research, exploring a human-in-the-loop approach that combines expert evaluation with LLM-assisted support within an integrated cycle. In contrast to work focused on isolated evaluation or generation, we investigate a human-in-the-loop design that closes the loop through evaluation–improvement–reevaluation on industrial user stories, quantifying alignment with experts and the net quality gain under INVEST.

3. Methodology

This section presents a two-phase empirical design grounded in principles of empirical software engineering, dependent exact replication with differentiated context [Shull et al. 2008], and methodological extension through a human-in-the-loop LLM workflow, aimed at assessing and improving user-story quality in an industrial agile context using GPT-5.1.

3.1. Design Overview and Replication Plan

Following replication taxonomy [Shull et al. 2008], our study constitutes a dependent exact replication with differentiated context, in which core elements of the original experimental procedure are preserved while selected contextual factors are deliberately varied to assess robustness and transferability. Specifically, (see Table 1) we hold constant the INVEST evaluation instrument, the calibrated prompting protocol, and the evaluation–improvement–reevaluation workflow, while changing the industrial context, dataset size, and LLM configuration.

Table 1. Comparison between the original study and the replication.

Aspect	Original study [Hernández et al. 2026]	Present study
Replication role	Initial empirical study	Dependent exact replication with differentiated context
Application domain	Academic grade management	Scholarship-process management
Industrial setting	Real-world industrial agile project	Real-world industrial agile project
Dataset size	60 user stories	49 user stories
Evaluation instrument	INVEST	INVEST (unchanged)
Prompting protocol	Calibrated RE evaluation prompt	Same calibrated prompt
Workflow	Evaluation–improvement–reevaluation	Same workflow
LLM	ChatGPT-5 (GPT-5-Thinking)	GPT-5.1 (API-based)
Execution mode	Conversational	Programmatic
Human experts	3 RE experts	3 RE experts (same profile)
Human-in-the-loop	Explicit HITL supervision	Explicit HITL supervision
Baseline variability control	Not reported	Multiple runs per story; median and IQR reported
Primary objective	Baseline human–LLM alignment	Alignment, quality improvement, and robustness

Based on this design, we formulate the following hypotheses: (H1) baseline human–LLM alignment comparable to that reported in the original study; (H2) post-intervention quality gains across all INVEST dimensions; and (H3) a more conservative stance by the LLM on the Independent and Small criteria. The study follows an empirical design adapted from prior work [Hernández et al. 2026], which employed an INVEST-based evaluation prompt with a simulated Requirements Engineer (RE) role and a dataset of 60 real user stories extracted from a product backlog of 112 items. In the original study,

three requirements-engineering experts and a large language model (ChatGPT-5 Plus / GPT-5-Thinking) participated to establish an initial comparison between human and automated evaluations under the INVEST framework, in the domain of academic grade management. The present work preserves the same evaluation instrument, calibrated prompt, and evaluation–improvement–reevaluation cycle, while introducing controlled changes in (i) the industrial project, (ii) the dataset, and (iii) the model employed. Specifically, the original domain is replaced by a student scholarship-process management system, the dataset size is reduced from 60 to 49 user stories, and ChatGPT-5 is replaced by GPT-5.1 accessed programmatically via API. These controlled modifications enable assessment of the approach's transferability to a new industrial context while maintaining methodological continuity.

Phase 1 replicates the quality evaluation using the same INVEST-based instrument, involving three expert requirements engineers and a single advanced LLM (GPT-5.1), selected for its suitability for automation and controlled replication. In this phase, a new dataset of 49 user stories from a different industrial project is introduced, allowing analysis of the approach in an alternative domain. Phase 2 introduces an LLM-assisted improvement intervention, maintaining INVEST as the quality reference and using the same 49 user stories as refinement targets. A calibrated prompt models the interaction between Requirements Engineer and Product Owner (RE–PO), enforcing a canonical user-story structure ("As a <role>, I want <capability> so that <benefit>"), measurable acceptance criteria (Given–When–Then), and domain-specific operational rules. This process produces improved versions of the original user stories.

Finally, the refined stories are reevaluated using the same evaluation prompt applied in both the original study and Phase 1, with participation from the three experts and the LLM. This design enables estimation of the LLM-assisted intervention's effect on requirements quality and its alignment with expert judgment under the INVEST framework. The study adopts a human-in-the-loop (HITL) approach to ensure methodological rigor and domain alignment throughout the evaluation and improvement process. The researchers actively supervised the LLM-assisted workflow by reviewing generated refinements, validating proposed changes against project context, and iteratively calibrating prompts and operational rules. This supervision ensured preservation of business value, adherence to domain constraints, and consistency with INVEST criteria, while mitigating model hallucinations and uncontrolled variability.

3.2. Research question

To preserve methodological continuity with the original study while capturing both evaluation alignment and improvement effects, we formulate a single research question:

RQ. How closely do LLM–assisted (GPT-5.1) evaluations align with expert judgment under the INVEST framework, and what changes in user-story quality are observed throughout the evaluation–improvement–reevaluation cycle?

This question integrates the study's three analytical focuses: (i) baseline alignment between expert and GPT-5.1–assisted INVEST evaluations, (ii) changes in user-story quality after guided GPT-5.1 intervention, and (iii) whether GPT-5.1 applies systematically stricter criteria in specific INVEST dimensions, particularly Independent and Small. In this way, the single RQ preserves the study's central methodological logic

while providing sufficient granularity to structure the results on alignment, improvement, and scoring behavior within the same evaluation cycle.

3.3. Project, dataset, participants, and tools

The empirical study was conducted in the context of a real-world software development project aimed at delivering a new version of an application for managing university student scholarship processes. The system supports multiple roles, including administrative staff, academic coordinators, and students. Development followed an agile methodology, with requirements managed as user stories in the product backlog of Azure DevOps Server. The dataset consisted of 49 user stories selected from the project backlog in January 2026. These stories covered functional modules such as user management, staff assignment, information registration, querying, updating and deletion, student request allocation, data maintenance, material assignment and control, and scheduling. The process began with a data preprocessing and normalization phase. Non-standard but functionally relevant stories were retained to reflect the variability typical of industrial documentation practices. All user stories were anonymized to remove sensitive institutional identifiers, system names, and infrastructure references.

Each user story was evaluated using the INVEST framework (Independent, Negotiable, Valuable, Estimable, Small, and Testable) on a five-point Likert scale, with independent assessments conducted by both human experts and GPT-5.1. The dataset included stories with complete acceptance criteria, stories lacking explicit criteria, and functional requirements not following the standard user-story format; nevertheless, most stories adhered to the canonical structure, reflecting ongoing team training and continuous improvement under Scrum. The participating requirements-engineering experts each had over 15 years of professional experience in software development projects, demonstrated expertise in agile methodologies (including formal Scrum adoption), and extensive practice managing INVEST-aligned product backlogs. Two experts were directly familiar with the scholarship application context, which serves a population exceeding 20,000 potentially eligible students. To mitigate evaluator bias, story identifiers and organizational references were anonymized; domain cues unnecessary for INVEST were hidden during evaluation.

This study represents a partial replication of prior work, differing from the original study in the application domain (academic grade management versus scholarship-process management), dataset size (60 versus 49 user stories), and model configuration. While the original study employed ChatGPT-5 (GPT-5-Thinking) in conversational mode, the present work used GPT-5.1 accessed programmatically via API. The same model was applied consistently across all evaluation, improvement, and reevaluation phases. Data preprocessing, including normalization and anonymization, was implemented using Python scripts executed in Visual Studio Code. The evaluation, improvement, and reevaluation phases were carried out programmatically via the OpenAI API using a paid account with active credit. GPT-5.1 was configured with a temperature of 0.2 to reduce stochastic variability and favor consistent outputs; the top-p parameter was kept at its default value (1.0). All other inference parameters were managed by the API and are reported as system defaults. The seed parameter was not exposed. All executions were performed in February 2026 (America/Costa_Rica time zone). To strengthen traceability and minimize procedural variance, fixed and standardized prompts were used throughout all phases, incorporating explicit INVEST protocol rules and control mechanisms to

prevent information invention or automatic completion. LLM outputs were logged together with execution metadata (date, model, temperature) and usage metrics (tokens per call). Given the non-deterministic nature of the model and the lack of explicit seed control, the baseline evaluation was conducted using six independent runs per user story [1](reported in the replication package) to quantify run-to-run variability. For the improvement and reevaluation phases, a predefined base run was applied to ensure comparability across conditions and avoid a combinatorial explosion of variants. For each INVEST criterion and user story, we report the median and interquartile range (IQR) across model runs to characterize nondeterministic variance at baseline.

A human-in-the-loop approach was adopted throughout the process: expert requirements engineers reviewed and validated LLM-generated refinements to ensure domain alignment, preservation of business value, and adherence to INVEST criteria across the entire evaluation–improvement–reevaluation cycle. For reference, the original study executed the full cycle using ChatGPT-5 (Plus plan, GPT-5-Thinking model) in conversational mode. All original-study executions were performed in October 2025 (America/Costa_Rica time zone).

3.4. INVEST Evaluation Framework

As in the original study, user-story quality was assessed using the INVEST framework (see Table 2). Each story was rated on a 5-point Likert scale (1 = Does not meet; 2 = Meets less than partially; 3 = Partially meets; 4 = Almost fully meets; 5 = Fully meets). To preserve methodological continuity and enable direct comparison, the same INVEST operationalization and scoring criteria were retained in this replication.

Table 2. User-story characteristics according to INVEST.

ID	Statement / INVEST characteristic
I	Independent: weakly related to other stories, with minimal dependencies that allow it to be developed/tested and deliver value on its own.
N	Negotiable: sufficient detail to discuss and prioritize.
V	Valuable: clear benefit aligned with the project context (role/business or technical).
E	Estimable: clear and bound enough to estimate effort.
S	Small: executable within one sprint (≤ 4 weeks).
T	Testable: verifiable with GWT (Given–When–Then) criteria—happy path, alternatives, and errors.

Note. INVEST = Independent, Negotiable, Valuable, Estimable, Small, Testable.

3.5. Prompt design and Implementation

The prompt design is organized into two interaction stages with the LLM: a preparation stage and an execution stage. During the preparation stage, prompts are constructed using a modular structure that combines role specification, task definition, contextual constraints, and incremental subtasks. Specifically, the prompts follow the formulation:

Initial Prompt$_i$ = Profile$_i$ + Task + Context + Subtask$_i$, ($1 \leq i \leq k$)
Follow-up Prompt$_i$ = Subtask$_i$ + Response$_{i-1}$, ($i > k$)
Where, for the initial prompt:
Profile$_{(i)}$: Describes the role, skills, or specific responsibilities.
Task: A broad description of the goal to be achieved.
Context: Details about the environment, constraints, or specific conditions.

[1] Replication package available at: https://github.com/HAErikam/Replicaci-n49US2026

$Subtask_{(i)}$: A specific step within the overall task.
For the follow-up prompt:
$Subtask_{(i)}$: A specific step within the overall task.
$Response_{(i-1)}$: The response generated after completing the previous subtask, which makes it possible to generate instructions that simulate the interaction among different activities carried out by this evaluator.

For the follow-up prompt, $Subtask_{(i)}$ is combined with $Response_{(i-1)}$, the response generated after completing the previous subtask, (thereby preserving contextual continuity across interactions). This structure supports incremental reasoning and controlled role simulation, allowing the LLM to perform evaluation, improvement, and reevaluation tasks while preserving contextual continuity. Figure 1 summarizes the complete workflow of the study under a human-in-the-loop approach. The process begins with (data preparation), where user stories undergo preprocessing, normalization, and anonymization through automated scripts, ensuring structural consistency and compliance with privacy protocols prior to evaluation. In Phase 1, an initial quality assessment of user stories is conducted using the INVEST framework and a five-point Likert scale. This phase includes independent evaluations by expert requirements engineers and GPT-5.1, establishing a baseline comparison between human and LLM-assisted judgments.

In Phase 2, GPT-5.1 supports the improvement of user stories through a calibrated prompt that simulates interaction between the Requirements Engineer and the Product Owner. User stories are refined according to a canonical structure (As–I want–so that) and verifiable acceptance criteria (Given–When–Then). Human supervision is incorporated at this stage to validate generated changes, ensure domain alignment, preserve business value, and (prevent scope expansion). Modifications are executed in a controlled manner, prioritizing compliance with the INVEST framework and quality attributes such as clarity, completeness, and lack of ambiguity. All improved stories are normalized to the standard format, incorporating measurable criteria when feasible and explicitly recording cases where criteria cannot be defined without introducing non-implied information, as applied in the original study. The process applies classification-specific rules for incomplete stories, incidents, enablers, and epics, ensuring consistent handling of each story type. In the final stage, the improved user stories are reevaluated using the same INVEST instrument, and the resulting data are subjected to quantitative analysis to examine consistency, correlation, and quality changes across the evaluation–improvement–reevaluation cycle. This phase enables estimation of the impact of the LLM-assisted intervention and its degree of alignment with expert judgment.

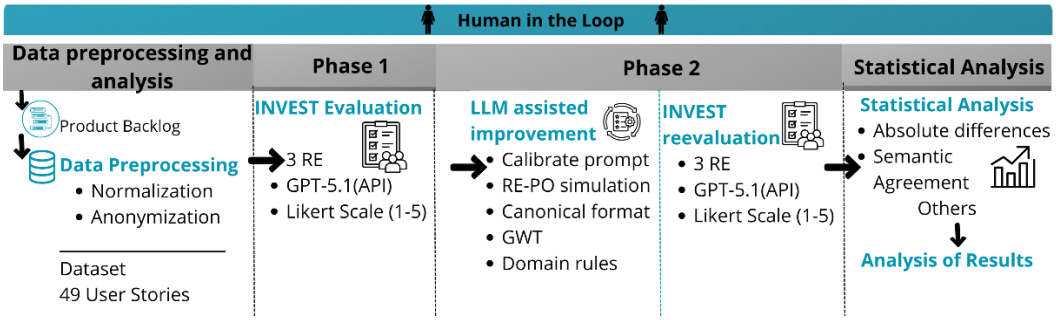

Figure 1. Requirements Quality Evaluation, Improvement, and Reevaluation

Across all stages, the human-in-the-loop approach ensures methodological traceability, variability control, and continuous expert validation. To analyze human–LLM alignment, a complementary set of statistical metrics is employed—absolute differences, Spearman rank correlation, Cronbach's alpha, Kruskal–Wallis, and semantic agreement—combining quantitative and qualitative analyses. This approach supports the identification of convergence patterns, interpretive discrepancies, and opportunities for prompt calibration, while preserving reproducibility and coherence with the original

study's methodology.

3.6. Threats to Validity

External validity. Results stem from a single industrial project (scholarship management) and a specific dataset (49 stories); therefore, external validity remains restricted, and generalization beyond similar contexts should be made with caution. Furthermore, robustness claims are constrained by the use of a single LLM configuration and a single industrial domain; broader validation is required to assess transferability across different teams, organizations, and backlog cultures. We disclose the sampling frame and dataset characteristics to support transferability assessments. **Construct validity.** INVEST ratings can be sensitive to operational definitions and story structure. We retained the original rubrics and prompts to preserve continuity; nonetheless, we include rubrics in the appendix for transparency. Semantic agreement relies on banding thresholds that, while common, remain an abstraction of nuanced judgments. **Internal validity.** Two experts were familiar with the domain, which may bias assessments; we anonymized identifiers and hid nonessential domain cues, but residual familiarity effects are possible. The improvement step could introduce anchoring; we mitigated this with fixed prompts and HITL checks. **Conclusion validity.** We report nonparametric tests with effect sizes and corrected p-values, but multiple metrics increase analysis degrees of freedom. **Model dependence.** Findings depend on GPT-5.1 with temperature=0.2, top-p=1.0, and API defaults; seeds were not exposed. We quantify baseline run-to-run variability and recommend re-runs under alternative models/configurations in future work. **Cost and reproducibility.** We log tokens and execution metadata; exact stories cannot be released due to confidentiality, but we provide prompts, code, and a synthetic dataset to enable method replication.

4. Analysis of Results

This section presents the results of the comparative analysis between evaluations performed by requirements engineers and GPT-5.1, as well as the effect of the evaluation–improvement–reevaluation cycle on user-story quality under the INVEST framework. To improve clarity and align the presentation with the research question, results are organized into three parts: baseline human–LLM alignment, pre/post quality changes after GPT-5.1 intervention, and criterion-specific patterns of stricter GPT-5.1 scoring. To capture complementary dimensions of convergence between human evaluators and the LLM, multiple metrics were employed: run-to-run variability statistics (median IQR, Q1, Q3, mean IQR, SD, minimum, and maximum), absolute differences (magnitude of discrepancies), Spearman rank correlation (relative ordering), semantic agreement (conceptual alignment), and Cronbach's alpha (internal consistency).

4.1. Baseline stability and human–LLM alignment

Baseline run-to-run variability was low across all INVEST criteria. Across the 294 story–criterion pairs (49 user stories × 6 criteria in the initial evaluation), the global median IQR was 0.00 (M = 0.035, SD = 0.196), indicating identical scores across six GPT-5.1 runs in more than half of the cases. At the criterion level, median IQR values were 0.00 for Independent, Negotiable, Valuable, Estimable, Small, and Testable, with only a small number of outliers exhibiting higher dispersion (maximum IQR = 1.75 for Estimable and 1.50 for Small; see Table 3). These findings indicate highly stable baseline

scoring behavior under the calibrated prompting protocol and low-temperature configuration, supporting the robustness of the LLM-assisted evaluation and motivating the use of a predefined base run for the subsequent improvement and reevaluation phases

Table 3. Run-to-run variability at baseline.

Criterion	N (story × criterion)	Median IQR	Q1	Q3	Mean IQR	SD	Min	Max
Independent	49	0.00	0.00	0.00	0.02	0.14	0.00	1.00
Negotiable	49	0.00	0.00	0.00	0.02	0.14	0.00	1.00
Valuable	49	0.00	0.00	0.00	0.06	0.21	0.00	0.75
Estimable	49	0.00	0.00	0.00	0.06	0.29	0.00	1.75
Small	49	0.00	0.00	0.00	0.05	0.26	0.00	1.50
Testable	49	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Note. GPT-5.1; story-level IQR across six runs. IQR values represent run-to-run dispersion across six independent GPT-5.1 baseline evaluations per user story. Ratings use a five-point Likert scale.

Human–LLM (see Table 5) alignment was assessed exclusively in the baseline phase (a), where GPT-5.1 and experts independently evaluated the original user stories. Reevaluation scores (b) were not considered for alignment because the stories had already been modified through the LLM-assisted intervention, which would introduce circularity and potential anchoring effects. Accordingly, post-intervention assessments were used solely to quantify quality improvements.

Alignment between GPT-5.1 and experts was evaluated in phase (a) using Spearman correlation, mean absolute differences (MAD), semantic agreement, and Cronbach's alpha. At the criterion level, Spearman correlations ranged from .497 (Independent) to .725 (Valuable), indicating moderate to high concordance in the relative ranking of stories. MAD values varied between 0.143 (Valuable) and 0.816 (Negotiable), suggesting greater divergence in negotiability and higher consistency in perceived value. Semantic agreement was high across all criteria (93.9%–100%), and Cronbach's alpha ranged from .891 to .971, indicating good to excellent internal consistency between GPT-5.1 and the expert median (see Table 5). Overall, these results show that, although the model tends to score more conservatively at baseline, its evaluations exhibit substantial alignment with human judgment under the INVEST framework, supporting its use as an assistive tool for initial user-story quality assessment.

Table 5. Human–LLM alignment in the initial evaluation phase (a)

Criterion	N	Spearman's ρ	MAD	Semantic agreement (%)	Cronbach's α
Independent	49	.497	0.347	98.0	.896
Negotiable	49	.513	0.816	93.9	.927
Valuable	49	.725	0.143	100.0	.971
Estimable	49	.655	0.592	98.0	.927
Small	49	.702	0.357	98.0	.891
Testable	49	.533	0.245	98.0	.908

Note. ρ = Spearman's rank correlation; MAD = mean absolute difference between GPT-5.1 and the expert median; semantic agreement represents the percentage of stories with differences ≤ 1 Likert point; α = Cronbach's alpha.

The Kruskal–Wallis test applied to the absolute differences (|expert − GPT-5.1|) indicates that disagreement magnitude is not homogeneous across experts for most criteria in evaluation (a). Statistically significant differences were observed for Independent (H = 24.62, p = 4.48e-06), Negotiable (H = 19.31, p = 6.44e-05), Estimable

(H = 26.63, p = 1.66e-06), Small (H = 10.48, p = 0.0053), and Testable (H = 8.89, p = 0.0117), suggesting that at least one expert exhibits a distinct pattern of discrepancy with GPT-5.1. Valuable was the only attribute without evidence of inter-expert differences (H = 0.94, p = 0.624), pointing to relatively stable expert–model disagreement for this dimension.

Figure 2 further illustrates that, in the initial evaluation, most discrepancies between GPT-5.1 and the experts are concentrated at low values (primarily between 0 and 1 point), reflecting a reasonable level of proximity in scoring. Valuable and Estimable exhibit the smallest variability, suggesting relatively consistent interpretations of business value and estimability. In contrast, Small and especially Testable show greater dispersion and several extreme values, revealing more pronounced disagreements in aspects related to story granularity and operational verifiability through GWT criteria. Figure 3 shows moderate correlations between GPT-5.1(a) and the experts ($\rho \approx 0.53$–0.65), with very high agreement between Expert1(a) and Expert3(a) ($\rho = 0.94$), while Expert2(a) exhibits lower correlations with the others. Figure 4 complements these findings by showing high semantic agreement across all criteria, with minor disagreements concentrated particularly in Small and Testable. Semantic agreement refers to grouping Likert scores into Negative (1–2), Neutral (3), and Positive (4–5) bands and assessing whether expert and LLM ratings fall within the same interpretive category.

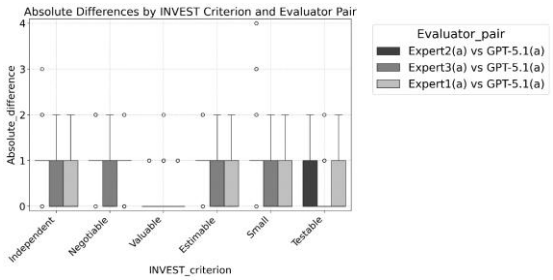

Figure 2. Absolute Differences by Criterion and Evaluator Pair

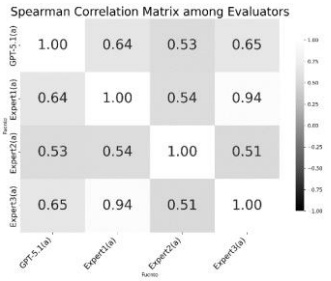

Figure 3. Spearman Correlation Matrix

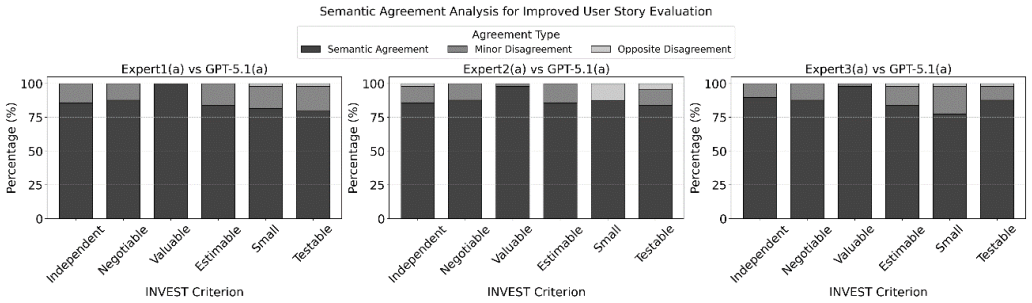

Figure 4. Semantic Agreement Analysis

4.2. Pre/post quality changes after GPT-5.1 intervention

To examine whether user-story quality changed after the LLM-assisted intervention, pre- and post-intervention INVEST scores for the same stories were compared using Wilcoxon signed-rank tests by criterion, with Holm correction and effect sizes reported as r. This paired non-parametric test was selected because the analysis compares repeated ordinal ratings before and after improvement. For expert evaluations (N = 42), significant improvements were observed for Negotiable (Md_pre = 5.0,

Md_post = 5.0, W = 0.0, p_adj = .006, r = .47), Estimable (Md_pre = 4.5, Md_post = 5.0, W = 0.0, p_adj < .001, r = .67), Small (Md_pre = 4.0, Md_post = 5.0, W = 0.0, p_adj < .001, r = .88), and Testable (Md_pre = 4.0, Md_post = 5.0, W = 0.0, p_adj < .001, r = .87), indicating medium to large effects. No statistically significant changes were detected for Independent (p_adj = .618) or Valuable (p_adj = .118). For Negotiable, the significant result despite identical pre/post pre/post medians suggest improvement in the paired score distribution. These patterns are consistent with a ceiling effect, where high baseline scores limit observable median change despite improvements in the paired score distribution.

For GPT-5.1, significant post-intervention gains were found for Independent (Md_pre = 4.0, Md_post = 4.0, W = 0.0, p_adj = .018, r = .45), Negotiable (Md_pre = 4.0, Md_post = 4.0, W = 0.0, p_adj = .037, r = .40), Estimable (Md_pre = 4.0, Md_post = 4.0, W = 0.0, p_adj = .046, r = .38), and Testable (Md_pre = 4.0, Md_post = 4.0, W = 0.0, p_adj < .001, r = .63). Changes in Valuable (p_adj = .077) and Small (p_adj = .077) did not reach statistical significance after Holm adjustment. As in the expert ratings, significant effects with unchanged medians reflect distributional shifts across paired observations not fully captured by the median alone. Taken together, these results indicate that the LLM-assisted intervention led to substantial quality improvements in criteria related to estimability, size, and testability as judged by experts, while GPT-5.1 exhibited significant gains in independence, negotiability, estimability, and testability. Valuable remained comparatively stable across both evaluators.

Descriptive statistics show consistently high scores across all INVEST criteria after the improvement phase, for both human experts and GPT-5.1. In particular, experts reported medians of 5 for Independent, Negotiable, Valuable, Estimable, Small, and Testable, with mean values ranging from 4.03 to 5.00 depending on the criterion. GPT-5.1 produced slightly more conservative assessments, with mean scores of 4.11 (Independent), 4.03 (Negotiable), 4.85 (Valuable), 3.97 (Estimable), 4.00 (Small), and 4.11 (Testable). The largest discrepancies emerged in Negotiable and Estimable, where experts reached mean values close to 5.00 while GPT-5.1 remained around 4.0, suggesting a stricter application of the criteria by the model in the presence of residual ambiguity or incompletely operationalized acceptance criteria. Despite these differences, substantial convergence is observed for Valuable and Testable, indicating strong alignment between human and LLM-assisted evaluations with respect to business value and story verifiability.

4.3. Criterion-specific stricter GPT-5.1 scoring

The non-parametric Kruskal–Wallis analysis applied to absolute differences between GPT-5.1 and human experts revealed statistically significant effects for Independent ($H = 96.70$, $p < 0.001$) and Small ($H = 37.32$, $p < 0.001$), indicating systematic discrepancies in these specific dimensions. In contrast, no significant differences were found for Negotiable ($H = 2.60$, $p = 0.27$), Valuable ($H = 0.03$, $p = 0.98$), Estimable ($H = 4.56$, $p = 0.10$), or Testable ($H = 4.38$, $p = 0.11$), suggesting statistically comparable alignment between LLM and expert evaluations for these criteria. These findings indicate that, following the evaluation–improvement–reevaluation cycle, GPT-5.1 converges with human judgment primarily in dimensions related to business value, negotiability, estimability, and verifiability. The persistent differences observed for Independent and Small reflect a more conservative interpretation by the model regarding functional decoupling and operational granularity. While experts tend to accept implicit

dependencies and slightly larger story sizes within a sprint, GPT-5.1 penalizes these conditions more strictly, favoring stories that are more autonomous and tightly scoped.

Figure 5 illustrates the absolute differences between GPT-5.1 and human experts after the improvement phase, disaggregated by INVEST criteria. Overall, a strong concentration of near-zero values is observed for Valuable and Estimable, indicating substantial convergence between the model and experts in identifying business value and estimating refined stories. Negotiable also exhibits low and stable discrepancies, whereas Testable shows moderate variation, with medians close to 1 and occasional outliers, reflecting localized differences in interpreting acceptance criteria. Independent and Small display greater dispersion and the presence of outliers, confirming that the model applies stricter judgment regarding functional autonomy and story size. Figure 6 shows that correlations between the LLM and the evaluators remain low to moderate ($\rho \approx 0.22–0.31$), indicating limited alignment in the relative prioritization of user stories, even though individual scores tend to converge. Figure 7 complements this pattern by showing a clear predominance of semantic agreement across evaluator–LLM pairs, generally exceeding 85–90% for Independent, Negotiable, Valuable, and Estimable, with slightly higher minor disagreement in Small and Testable. This indicates that, even when differences persist in score magnitude or relative ordering, GPT-5.1 and the experts largely converge in their substantive interpretation of refined story quality, reinforcing the positive effect of the evaluation–improvement–reevaluation cycle and the value of the human-in-the-loop approach for stabilizing INVEST-based judgments.

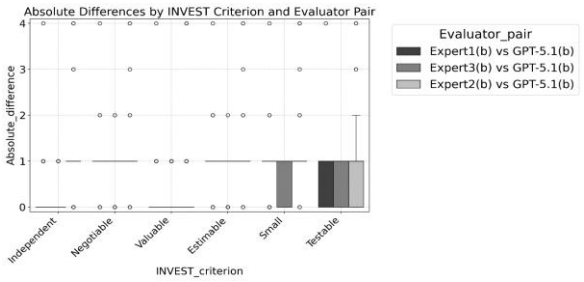

Figure 5. Absolute Differences by Criterion and Evaluator Pair

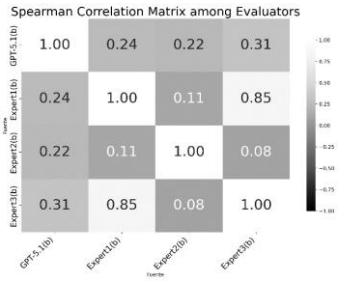

Figure 6. Spearman Correlation Matrix

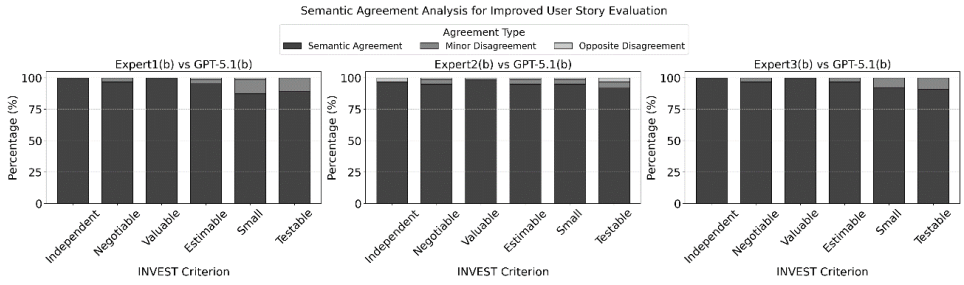

Figure 7. Semantic Agreement Analysis

Overall, GPT-5.1 showed high semantic agreement with expert judgment, although ranking-based correlations remained low, indicating alignment in score magnitude rather than in story prioritization. After the evaluation–improvement–reevaluation cycle, user-story quality improved, and significant differences remained mainly in Independent and Small. Residual non-consensus was concentrated in these criteria, where GPT-5.1 applied stricter judgments on functional autonomy and sprint-

level granularity, particularly when it proposed decomposing five baseline stories into two or more smaller stories, increasing the dataset from 49 to 65 user stories. More limited disagreement also persisted in some Testable assessments, where refinement added acceptance criteria and scenarios to strengthen verifiability and value, although their sufficiency and scope remained open to expert interpretation. Taken together, these findings suggest that GPT-5.1 may provide a useful complementary perspective on the structural aspects of user stories. .

5. Conclusions

This industrial replication study shows that GPT-5.1 can effectively support the evaluation and improvement of user stories under the INVEST framework, achieving substantial alignment with expert judgment in both magnitude and semantic interpretation. Across the evaluation–improvement–reevaluation cycle, we observed significant improvements in several INVEST criteria (Negotiable, Estimable, Small, Testable), whereas others (Independent, Valuable) remained statistically unchanged, reflecting a mixed improvement profile. Nevertheless, the model maintained stricter criteria for Independent and Small, suggesting that GPT-5.1 provides a complementary structural perspective focused on functional autonomy and operational granularity. In practice, our results suggest deferring to expert judgment for prioritization and negotiation, while leveraging the LLM as a conservative structural checker for independence and scope granularity during backlog refinement. These findings indicate that LLMs can function as quality-control agents that reinforce backlog consistency without replacing expert judgment. In this context, the results highlight the central role of a human-in-the-loop approach for integrating LLMs into industrial requirements-engineering processes. While GPT-5.1 contributes to systematizing evaluation, reducing manual effort, and promoting more rigorous criteria, human oversight remains essential to preserve domain context, validate prioritization decisions, and ensure alignment with business objectives. Together, automation and expert judgment combine consistency and scalability with contextual interpretation and organizational value. In this setting, this work provides a practical and replicable pathway for incorporating LLMs as support tools for requirements evaluation. While promising, these findings should be interpreted within the bounds of the specific industrial context, dataset size, and LLM configuration studied.

6. Acknowledgments

This research was partially supported by Project No. 834-C5-218 ECCI-CITIC-UCR, and the Graduate Program in Computer Science and Informatics at the University of Costa Rica (UCR). During the preparation of this work, AI-based tools were used to support the refinement of academic English, improve readability, and ensure consistency in terminology across sections. After using these tools, the authors carefully reviewed, validated, and edited all generated suggestions to ensure that the manuscript reflects their own scientific contributions, interpretations, and insights. The authors take full responsibility for the integrity, accuracy, and originality of the content presented in this article.

7. References

Baltes, S. et al. (2025) “Guidelines for Empirical Studies in Software Engineering Involving Large Language Models”, arXiv preprint arXiv:2508.15503.

Belzner, L., Gabor, T. and Wirsing, M. (2023) "Large language model assisted software engineering: prospects, challenges, and a case study", In: International Conference on Bridging the Gap between AI and Reality (pp. 355-374). Springer Nature Switzerland.

Bosch, J. (2014) "Continuous software engineering: An introduction", In: Bosch, J. (eds) Continuous Software Engineering (pp. 3-13). Springer, Cham.

Bourque, P., and Fairley, R. (2014). Guide to the Software Engineering Body of Knowledge (Swebok). 335.

Fitzgerald, B. and Stol, K. (2017) "Continuous software engineering: A roadmap and agenda", In Journal of Systems and Software, 123, 176-189.

Hernández–Agüero, E., Quesada–López, C. and Chaves–Sánchez, J. P. (2024) "Integración de Enfoques Ágiles para el Mejoramiento Continuo de Procesos de Software", In: 13th CIMPS, IEEE, pp. 01–14.

Hernández-Agüero, E., Quesada-López, C. and Chaves-Sánchez, J. (2025) "Evaluación de la Calidad de Historias de Usuario Usando Modelos de Lenguaje de Gran Tamaño: Un Estudio en la Industria", In: Anais do XXVIII Congresso Ibero-Americano em Engenharia de Software, pp. 45–59, Porto Alegre, SBC.

Hernández-Agüero, E., Quesada-López, C., and Chaves-Sánchez, J. (2026). "Assessing and Improving the Quality of User Stories Using Large Language Models: An Empirical Study in an Industrial Context". In press.

Krishna, M., Gaur, B., Verma, A. and Jalote, P. (2024). "Using LLMs in software requirements specifications: an empirical evaluation", In: 2024 IEEE 32nd International Requirements Engineering Conference (RE) (pp. 475-483). IEEE.

Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R. and Zhu, C. (2023) "G-EVAL: NLG Evaluation Using GPT-4 with Better Human Alignment", arXiv preprint arXiv:2303.16634.

Marques, N., Silva, R. and Bernardino, J. (2024). Using chatgpt in software requirements engineering: A comprehensive review. Future Internet, 16(6), 180.

Ronanki, K., Berger, C. and Horkoff, J. (2023) "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). IEEE.

Roumeliotis, K., Tselikas, N. and Nasiopoulos, D. (2024). "LLMs in e-commerce: a comparative analysis of GPT and LLaMA models in product review evaluation", In: Natural Language Processing Journal, 6, 100056.

Santos, R., Steinmacher, I., Conte, T., Oran, A. C. and Gadelha, B. (2025) "Adoption of LLMs in Requirements Engineering: What Practitioners Are Worried About?", In: Simpósio Brasileiro de Qualidade de Software (SBQS), pp. 248–258, SBC.

Shull, F. J., Carver, J. C., Vegas, S. and Juristo, N. (2008) "The Role of Replications in Empirical Software Engineering", ESEJ, 13(2), pp. 211–218.

Wang, Y., Zhang, X., Li, Z., Chen, Z. and Wang, X. (2025) "Can LLMs Replace Human Evaluators? An Empirical Study of LLM-as-a-Judge in Software Engineering".

Zhang, Z., Rayhan, M., Herda, T., Goisauf, M., and Abrahamsson, P. (2024). "Llm-based agents for automating the enhancement of user story quality: An early report", In: International Conference on Agile Software Development (pp. 117-126). Springer.