

УДК 004.05

РОЗУМІННЯ СПЕЦИФІКАЦІЙ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЯК ФУНДАМЕНТАЛЬНОГО ЕЛЕМЕНТУ МЕТРИК ЯКОСТІ

І. В. Гілета, Р. О. Кіселик

*Національний університет «Львівська політехніка»,
вул. Степана Бандери, 12, Львів, 79013, Україна*

Розуміння та забезпечення якості специфікацій (QS) програмного забезпечення (ПЗ) є критичним превентивним заходом, що визначає надійність та валідність подальших метрик якості продукту. Це дослідження синтезує сучасні дані, підтверджуючи, що якість кінцевого ПЗ є прямою функцією від якості вхідних специфікацій. Аналіз ключових стандартів (ISO 25010, IEEE 1061) та методологій (Goal/Question/Metric, GQM) демонструє консенсус щодо необхідності систематичного вимірювання атрибутів вимог, таких як повнота, узгодженість та однозначність. Виявлено, що GQM забезпечує структуровану основу для емпіричної валідації цих метрик, тоді як коефіцієнт стабільності вимог слугує ключовим практичним індикатором якості специфікації. У роботі також обговорюються суперечності щодо застосування формальних методів та значні обмеження, пов'язані з квантифікацією природної мови (Natural Language Processing, NLP), включаючи проблеми неоднозначності. Проте, інтеграція великих мовних моделей (LLM) відкриває перспективні напрями для автоматизованого аналізу якості та підвищення економічної ефективності розробки (ROI). Зроблені висновки підкреслюють стратегічне зміщення фокусу з метрик коду на метрики вимог як основного фактора ризику.

Ключові слова: специфікація ПЗ, атрибути вимог, метрики якості, стандарти якості ПЗ, економічна ефективність розробки.

Постановка проблеми. Якість вимог (Requirements Engineering, RE) [1, 2] формує фундаментальну основу всього життєвого циклу розробки програмного забезпечення. Відомо, що неповні та змінні вимоги є домінуючими факторами провалу або значних перевитрат проєктів. Вимоги є базою процесу розробки, і проблеми на цій фазі мають глибокий і дороговартісний вплив на кінцеву функціональність системи.

Аналіз останніх досліджень та публікацій. Механізми покращення якості ПЗ вимагають систематичного вимірювання відповідних атрибутів. Це не лише технічна, але й управлінська необхідність.

Специфікація програмного забезпечення – це документ, що описує вимоги; якість специфікації, натомість, є набором внутрішніх атрибутів, які визначають придатність цього документа. Метрика програмного забезпечення – це захід, що дозволяє отримати числове значення певної властивості ПЗ або його специфікацій

[2, 3]. Метрики якості поділяються на ті, що оцінюють властивості самого ПЗ, і ті, що оцінюють умови розробки, включаючи якість специфікації [4].

Мета статті. Організації повинні розробляти метрики, які охоплюють як характеристики кінцевого продукту (надійність, продуктивність), так і якість процесів, які його створюють, починаючи саме з фази специфікації. Актуальність такого завдання полягає в необхідності об'єднання знань про формальні міжнародні стандарти та новітні емпіричні підходи (наприклад, із застосуванням NLP та LLM) для створення цілісного, валідованого фреймворку оцінки якості.

Виклад основного матеріалу дослідження. Центральна наукова гіпотеза полягає в тому, що якість кінцевого продукту є прямою функцією від якості вхідних специфікацій. Якщо ПЗ реалізує неповні або неоднозначні вимоги, воно може вважатися якісним лише за критерієм «відповідності специфікації», але не за критерієм «задоволення потреб користувача». Це зміщує стратегічну увагу на метрики, що вимірюють фундаментальні атрибути специфікації.

Ключові міжнародні стандарти якості одностайно підтверджують необхідність вимірювання та аналізу процесів. До таких стандартів належать CMMI (Capability Maturity Model Integration), ISO 9001, IEEE 1061 (Standard for a Software Quality Metrics Methodology), та ISO 25010 (Systems and Software Quality Requirements and Evaluation, SQuaRE).

ISO/IEC 25010 надає рамкову структуру для оцінювання якості ПЗ, наголошуючи на важливості визначення характеристик якості (таких як надійність, функціональність), які мають бути точно описані та вимірювані, починаючи з етапу вимог. Модель CMMI розглядає управління вимогами (Requirements Management, RM) як ключову область процесу (Key Process Area, КРА) [5]. Вимірювання, визначені в RM, допомагають розуміти, контролювати та оцінювати процес управління вимогами.

Систематичний аналіз свідчить, що стандарти вимагають не просто наявності метрик, а систематичного вимірювання процесів [6]. Це означає, що зрілість процесу (Process Maturity), підтверджена CMMI, має передувати вимірюванню якості продукту. Без контролю вхідних специфікацій вихідні метрики продукту є ненадійні.

На початкових етапах розвитку інженерії ПЗ використовувалися переважно структурні метрики, такі як метрики Холстеда або обсяг коду (Lines of Code, LOC), які дозволяли оцінювати складність та якість ПЗ за допомогою статичного аналізу коду.

Однак обмеженість таких метрик призвела до зміщення акценту на цільово-орієнтовані підходи. Парадигма Goal/Question/Metric (GQM), заснована Базілі та Вайсом у 1984 році, стала фундаментальною для вимірювання продуктів та процесів. GQM забезпечує структурований підхід, який пов'язує високоцінні бізнес-цілі (які походять безпосередньо зі специфікацій) з конкретними питаннями та метриками [7].

Еволюція ключових підходів вимірювання ілюструється в таблиці 1.

Таблиця 1

**Зв'язок ключових стандартів та методологій зі специфікаціями
та метриками якості**

Стандарт / модель	Фокус вимірювання	Зв'язок зі специфікаціями	Основна ціль
ISO/IEC 25010 (SQuaRE)	Якість продукту та системи	Характеристики якості – функціональність і надійність, визначені на етапі вимог, дають вимірюваність.	Надання рамкової структури для оцінювання якості ПЗ.
IEEE 1061	Методологія метрик якості ПЗ	Надає процес для визначення, збору та аналізу метрик, який застосовується до артефактів RE.	Встановлення процедури вимірювання якості, незалежно від конкретних метрик.
CMMI	Зрілість процесів	RM є ключовою областю. Метрики застосовуються для оцінки зрілості RE-процесу.	Покращення процесу вимірювання, контролю та передбачення.
GQM (Goal/Question/Metric)	Цільово-орієнтоване вимірювання	Зв'язує високорівневі цілі з конкретними питаннями та метриками для емпіричної валідації.	Забезпечення релевантності метрик для цілей організації.

GQM успішно застосовується для визначення та валідації заходів, що стосуються управління вимогами CMMI, демонструючи, як цільове вимірювання може бути емпірично валідоване в реальних проєктах. Перехід від структурних метрик до GQM відображає зміну фокусу від вимірювання як система побудована (структурна якість) до вимірювання чи система задовольняє свої цілі (функціональна/цільова якість). Оскільки ці цілі деталізовані у специфікаціях, якість специфікації є мета-метрикою. Якщо цілі нечіткі або неповно визначені, навіть ідеально валідована GQM-структура вимірюватиме лише ілюзію якості [8].

Ключові атрибути якості специфікацій (QS).

Якість специфікації визначається набором атрибутів, найчастіше серед яких виділяють: однозначність, повноту, узгодженість, правильність, та верифікованість.

Емпіричні дослідження показали, що неоднозначність має найбільшу частку підтипів дефектів серед усіх атрибутів якості вимог. Це підкреслює критичний вплив неоднозначності на інтерпретацію та реалізацію, оскільки слова та речення природної мови часто мають кілька значень, і їхнє правильне тлумачення сильно залежить від контексту. Дефект неоднозначності на етапі вимог діє як «підсилювач», множачи дефекти на етапах реалізації та тестування.

Неповні вимоги є одним із першочергових факторів провалу проєктів, оскільки вони призводять до відсутності необхідної функціональності або помилок, пов'язаних із незадовільним покриттям цілей [7].

Головним механізмом, який трансформує якість специфікації у вимірювану метрику, є процес верифікації та валідації вимог (ВВ). ВВ включає перевірку документів на однозначність, узгодженість, повноту та точність.

Ключові техніки ВВ, які безпосередньо породжують метрики якості, включають:

1. Трасування: Це процес забезпечення покриття вимог, який перевіряє, що кожен елемент специфікації має відповідне обґрунтування та що кожен реалізований компонент може бути відстежений до конкретної вимоги. Трасування є фундаментальним механізмом для підтримки аргументації щодо якості програмного забезпечення та процесів розробки. Наявність та якість трасувальних зв'язків є мета-метрикою для оцінки верифікованості та повноти специфікацій. Якщо ці зв'язки відсутні, метрика «Покриття тестами» втрачає сенс.

2. Тестування вимог: Вимога має бути тестувальною. Формулювання тест-кейсів, які можуть бути відстежені до вимог, забезпечує верифікованість. Це безпосередньо веде до метрики «Відсоток тестованих вимог».

3. Формальна верифікація: Використовується для перевірки логічної узгодженості та повноти у критичних системах.

Однією з найбільш інформативних практичних метрик, що відображають якість початкової специфікації, є «Коефіцієнт стабільності вимог» (RSC). Він обчислюється як відношення кількості змін у вимогах до загальної кількості реалізованих вимог.

Високий RSC (особливо, якщо він значно перевищує 0.5 або 1.0) є прямим індикатором низької якості початкової специфікації. Такий високий коефіцієнт означає, що команда фокусується на переробці раніше випущеного функціоналу через неповноту або неправильність вихідних вимог, а не на створенні нової бізнес-цінності. Цей показник також дає уявлення про легкість масштабованості системи. Таблиця 2 узагальнює зв'язок між атрибутами специфікації та механізмами їх квантифікації.

Парадигма GQM є найбільш визнаним цільово-орієнтованим підходом для вимірювання в інженерії ПЗ. Вона вимагає, щоб вимірювання починалося з чітко визначеної мети (Goal), яка потім декомпонується на набір питань, які необхідно вирішити, і лише потім визначаються конкретні метрики (Metrics) для збору даних [9].

GQM було успішно застосовано для емпіричної валідації набору заходів управління вимогами, що відповідають специфічним практикам «Ключової області процесу» (КРА) управління вимогами в CMMI. Валідація цих метрик проводилася через опитування практиків та аналіз історичних даних з використанням статистичних методів (наприклад, коефіцієнта Кронбаха Альфа), що підтверджує надійність GQM для вимірювання процесів RE. Таким чином, GQM забезпечує критично важливу структуру для перетворення абстрактних атрибутів якості специфікації в емпірично валідовані заходи.

Таблиця 2

Ключові атрибути якості специфікацій та методи їх валідації

Атрибут якості специфікації	Визначення впливу на метрики якості ПЗ	Ключові техніки ВВ/квантифікації
Повнота	Прямо впливає на покриття вимог та ризик незавершеної функціональності	Трасування до цілей, прототипування, огляд документації.
Однозначність	Найбільший фактор дефектів; спричиняє неправильну реалізацію та високий коефіцієнт стабільності вимог.	NLP-аналіз, контроль структури (шаблони), формальна верифікація (для критичних частин).
Узгодженість	Впливає на логічну коректність та уникнення конфліктів у системі.	Формальна верифікація, моделювання, тестування вимог.
Верифікованість	Ключовий фактор для визначення метрик тестування та валідації. Неверифіковані вимоги унеможливають об'єктивне вимірювання.	Написання тест-кейсів на основі вимог.

Незважаючи на широке використання GQM, існують обґрунтовані критичні зауваження. Виділяють три основні проблеми: 1) підхід не завжди є повторюваним, оскільки різні особи, які починають з однієї мети, можуть прийти до різних питань і метрик; 2) складно визначити оптимальний момент для припинення створення питань і переходу до визначення метрик; 3) деякі питання можуть виявитися неможливими для відповіді в поточній організації.

Ці обмеження вимагали розвитку методології. У відповідь на необхідність підвищення надійності та зрілості вимірювань було запропоновано розширений підхід V-GQM (Validation Goal/Question/Metric). V-GQM включає додаткові етапи, такі як валідація самих метрик та аналіз питань, які виходять за рамки початкової парадигми GQM. У контексті GQM також наголошується, що вимірювання стабільних і зрілих сутностей можна проводити об'єктивно, тоді як нестабільні або неформальні сутності (наприклад, рання, неповна специфікація) вимагають суб'єктивної оцінки. Це безпосередньо стосується оцінки якості специфікації на ранніх етапах.

Незважаючи на значний інтерес до якості вимог, систематичні огляди виявляють істотні прогалини в емпіричній обґрунтованості досліджень. Існує необхідність у проведенні більшої кількості досліджень, які не лише визначають метрики якості вимог, але й валідують їх на практиці.

Крім того, більшість існуючих досліджень розглядають «вимоги» у найширшому сенсі. Мало уваги приділяється якості специфічних артефактів, таких як Use Cases або User Stories. Ця обмеженість ускладнює розробку та валідацію метрик, адаптованих до конкретних типів документації чи методологій розробки.

Використання NLP-технік є ключовим напрямком для автоматизованого виведення метрик якості зі специфікацій природною мовою. Однак це пов'язано зі значними викликами:

1. Неоднозначність та контекст: Одним із найзначніших бар'єрів є лексична та контекстуальна неоднозначність мови. NLP-моделі повинні бути здатні розуміти контекст, доменний жаргон та ідіоматичні вирази. Якщо модель неправильно визначає контекст, автоматично отримана метрика (наприклад, оцінка однозначності вимоги) буде хибною.

2. Проблема: Ефективність будь-якої NLP-моделі критично залежить від якості та кількості навчальних даних. Отримання великих, високоякісних, анотованих наборів даних для специфічних доменів є трудомістким і дорогим. Цей бар'єр обмежує масштабованість інструментів, особливо для малих та середніх підприємств.

3. Розрив між дослідженням та практикою: Огляд літератури показує, що лише невелика частка (11%) інструментів на основі NLP, описаних в академічних роботах, реально доступна для практичного використання. Це підтверджує значний розрив між теоретичним потенціалом та практичним впровадженням інструментів для автоматизованого вимірювання якості специфікацій.

Навіть ідеально розроблені та валідовані метрики якості можуть бути неефективними через нетехнічні, організаційні бар'єри. До них належать опір змінам, відсутність чітких цілей, невідповідний вибір метрик, які не узгоджуються з бізнес-цілями, та брак підтримки з боку керівництва. Якщо метрики не інтегровані належним чином у культуру та процеси, їхнє впровадження не матиме практичного сенсу, незалежно від їхньої математичної коректності.

Вдосконалення процесів розуміння та валідації специфікацій має прямі та значні фінансові наслідки. Якість специфікації є вимірюваним фінансовим активом. Дослідження показують, що організації, які впроваджують автоматизовані протоколи валідації на ранніх етапах розробки (тобто на фазі специфікації), можуть зменшити кількість дефектів після релізу до 40%. Це забезпечує значну економію коштів за рахунок зниження витрат на виправлення та підтримку.

Надійність, що походить від чітких та валідованих специфікацій, сприяє підвищенню довіри кінцевих користувачів. Крім того, компанії, які залучають спеціалізовані групи для валідації, повідомляють про прискорення циклу зворотного зв'язку на 25% та скорочення часу виходу на ринок (time-to-market) на понад 15%. Це перетворює валідацію якості специфікацій із технічної необхідності на фактор зростання бізнесу [11, 12].

Коефіцієнт стабільності вимог (RSC) є ключовою прогностичною метрикою, що дозволяє керівництву оцінювати якість специфікації в грошовому еквіваленті. Низький RSC корелює з низькою кількістю дефектів, що, у свою чергу, перекладається на менші операційні витрати та пряме підвищення ROI [11].

Ця метрика використовується для:

1. Прогнозування переробки: Високий RSC свідчить про необхідність непродуктивного переписування вже реалізованих функцій.

2. Оцінки масштабованості: Надмірна кількість змін у вимогах вказує на те, що функціонал системи є крихким і важко масштабується.

Організаційне керівництво вимагає вимірювання в грошовому еквіваленті, і тому метрики якості специфікацій, які можуть бути пов'язані з витратами на дефекти та часом виходу на ринок, є стратегічно важливими.

Застосування технологій NLP, ML та LLM [8] є найбільш перспективним напрямом для автоматизації вимірювання якості специфікацій. Ці технології пропонують механізми для:

- Автоматизованого виявлення дефектів: LLM можуть бути використані для автоматичного виявлення неоднозначності, неузгодженості та неясності у вимогах природною мовою [16].
- Класифікація та верифікація: ML/NLP може класифікувати нефункціональні вимоги (НФВ), які традиційно важко виміряти. LLM також застосовуються для валідації згенерованого коду та специфікацій тест-кейсів, забезпечуючи їх відповідність цілям RE [16].
- Трасування: Розвиток автоматизованого відновлення та підтримки трасувальних зв'язків за допомогою NLP має вирішальне значення для критично важливих систем, підвищуючи достовірність метрик покриття вимог [17].

Незважаючи на потенціал LLM, критичним напрямом майбутніх досліджень є розробка робустих фреймворків для валідації та оптимізації їхніх виходів, адаптованих до нюансів Requirements Engineering. Це необхідно, оскільки інтеграція LLM несе ризик успадкування упереджень (biases) з навчальних даних. Якщо моделі навчаються на корпусі неякісних або упереджених вимог, автоматизовані метрики якості можуть несправедливо класифікувати нові вимоги як дефектні [18].

Крім того, ефективність LLM залишається залежною від якості та кількості анотованих даних. Якщо LLM використовується для вимірювання якості специфікації, але навчалася на неідеально анотованій документації, її метрики, наприклад, «оцінка однозначності», повинні піддаватися суворій перевірці. Області застосування LLM та пов'язані з ними виклики представлені у таблиці 3.

Таблиця 3

Перспективи та обмеження LLM/NLP у квантифікації якості специфікацій

Область застосування	Очікуваний вплив на метрики якості	Ключовий виклик / обмеження	Майбутній напрям дослідження
Виявлення дефектів/неоднозначності	Автоматизоване визначення метрик якості.	Залежність від контексту та неоднозначність мови.	Розробка робустих фреймворків валідації LLM-виходів для RE.
Автоматизоване трасування	Підвищення надійності метрик покриття вимог; ручних зусиль.	Складність підтримки актуальності та точності зв'язків.	Розвиток ефективних алгоритмів Trace Link Recovery&Maintenance.
Формальна верифікація	Автоматизація перевірки коректності згенерованих артефактів.	Ризик «багів у специфікації»; складність формалізації.	Оптимізація LLM для забезпечення відповідності стандартам.

Майбутні дослідження також мають продовжити емпіричну валідацію GQM-подібних підходів (V-GQM) для підвищення їх повторюваності та об'єктивності. Особлива увага потрібна в контексті малих та середніх підприємств, де впровадження складних програм вимірювання, незважаючи на їхню корисність, залишається значно складнішим через обмежені ресурси. Необхідно розробити адаптовані настанови для забезпечення ефективної кількісної оцінки в таких організаціях [18].

Висновки. Систематичний синтез наявних доказів підтверджує, що розуміння та забезпечення якості специфікацій програмного забезпечення є критичним елементом для визначення валідних та надійних метрик якості. Міжнародні стандарти (ISO 25010) та методології (GQM) забезпечують формальні рамки, які вимагають, щоб метрики якості продукту були логічно пов'язані з вимірюванням ключових атрибутів вимог: повноти, узгодженості та однозначності.

Емпіричні дані свідчать про те, що до 60% дефектів можуть походити з помилок у специфікації, що робить метрики якості вимог (наприклад, «Коефіцієнт стабільності вимог») більш стратегічно значущими, ніж метрики коду. Техніки валідації та верифікації, такі як трасування, є не просто етапами процесу, а метаметриками, що дозволяють підтримувати аргументацію щодо загальної якості системи.

Майбутній прогрес значною мірою залежить від подолання викликів квантифікації природної мови, зокрема, лексичної та контекстуальної неоднозначності, що є пріоритетом для досліджень у галузі NLP та LLM. Успішна розробка робустих фреймворків для валідації LLM-виходів та подальший розвиток методологій типу V-GQM відкривають шлях до автоматизованого, об'єктивного та економічно ефективного вимірювання якості специфікацій, що матиме прямий вплив на підвищення ROI та загальної зрілості процесів розробки ПЗ.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. F. N. Colakoglu et al. Software product quality metrics: a systematic mapping study [Електронний ресурс]. URL: <https://scispace.com/pdf/software-product-quality-metrics-a-systematic-mapping-study-2w1ho43zkb.pdf/> (дата звернення: 17.10.2025). – Назва з екрана.
2. Класифікація метрик якості програм [Електронний ресурс]. URL: https://stud.com.ua/28172/informatika/metriki_yakosti_programnogo_zabezpechennya/ (дата звернення: 17.10.2025). – Назва з екрана.
3. The Goal/Question/Metric Paradigm (GQM) [Електронний ресурс]. URL: <https://publica.fraunhofer.de/bitstreams/3c4dda1a-1c71-4a86-8b29-9a164bb31037/download/> (дата звернення: 17.10.2025). – Назва з екрана.
4. Lindstrom. Draft. The GQM approach is not static but might need refinement [Електронний ресурс]. URL: https://fileadmin.cs.lth.se/serg/old-serg-dok/docs-masterthesis/62_Lindstrom_draft.pdf/ (дата звернення: 17.10.2025). – Назва з екрана.
5. Requirements management measures help organizations to understand, control and assess requirements management process [Електронний ресурс]. URL: <https://www.acit2k.org/ACIT2007/Proceeding/204.pdf/> (дата звернення: 17.10.2025). – Назва з екрана.

6. Коефіцієнт стабільності вимог [Електронний ресурс]. URL: <https://qagroup.com.ua/publications/what-is-metrics/> (дата звернення: 17.10.2025). – Назва з екрана.
7. Common Challenges and Pitfalls Implementing quality metrics [Електронний ресурс]. URL: <https://safetyculture.com/topics/quality-metrics/> (дата звернення: 17.10.2025). – Назва з екрана.
8. Role of LLMs and AI in software requirements quality metrics future research [Електронний ресурс]. URL: <https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2025.1519437/full/> (дата звернення: 17.10.2025). – Назва з екрана.
9. Challenges in quantifying natural language requirements understanding and its impact on metrics [Електронний ресурс]. URL: <https://shelf.io/blog/challenges-and-considerations-in-nlp/> (дата звернення: 17.10.2025). – Назва з екрана.
10. Major challenges of Natural Language Processing [Електронний ресурс]. URL: <https://www.geeksforgeeks.org/nlp/major-challenges-of-natural-language-processing/> (дата звернення: 17.10.2025). – Назва з екрана.
11. Case Study Compilation - How Software Testing Companies Maximize Client ROI [Електронний ресурс]. URL: <https://moldstud.com/articles/p-case-study-compilation-how-software-testing-companies-maximize-client-roi/> (дата звернення: 17.10.2025). – Назва з екрана.
12. Formal specification in software engineering [Електронний ресурс]. URL: <https://afzalbadshah.medium.com/formal-specification-in-software-engineering-b66852d96fc6/> (дата звернення: 17.10.2025). – Назва з екрана.
13. Empirical research on requirements quality: a systematic mapping study [Електронний ресурс]. URL: https://www.researchgate.net/publication/358630030_Empirical_research_on_requirements_quality_a_systematic_mapping_study/ (дата звернення: 17.10.2025). – Назва з екрана.
14. An Empirical Study of Software Requirements Verification and Validation Techniques along their Mitigation Strategies [Електронний ресурс]. URL: https://www.researchgate.net/publication/281645652_An_Empirical_Study_of_Software_Requirements_Verification_and_Validation_Techniques_along_their_Mitigation_Strategies/ (дата звернення: 17.10.2025). – Назва з екрана.
15. Requirements impact as recorded in the issue reports [Електронний ресурс]. URL: <https://arxiv.org/html/2402.00594v2/> (дата звернення: 17.10.2025). – Назва з екрана.
16. NLP-assisted software testing SLR [Електронний ресурс]. URL: https://pureadmin.qub.ac.uk/ws/files/219213040/NLP_assisted_software_testing_SLR_Mar_01.pdf. / (дата звернення: 17.10.2025). – Назва з екрана.
17. Research on applying ML and NLP techniques to the classification of non-functional requirements and performance metrics [Електронний ресурс]. URL: https://research.manchester.ac.uk/files/183513868/2020_12_16_NLP4RE_Accepted.pdf. / (дата звернення: 17.10.2025). – Назва з екрана.
18. Traceability, the ability to trace relevant software artifacts to support reasoning about the quality of the software [Електронний ресурс]. URL: <https://arxiv.org/html/2405.10845v1/> (дата звернення: 17.10.2025). – Назва з екрана.
19. Гілета І. В., Кіселик Р. О., Карнаухов В. А., Коминар Т. Н. Огляд сучасних підходів до оцінювання якості програмного забезпечення Наукові записки № 1 (70) / 2025. С. 72-82, doi: 10.32403/1998-6912-2025-1-70-72-82.

REFERENCES

1. F. N. Colakoglu et al. (2021) Software Product Quality Metrics: A Systematic Mapping Study. Retrieved October 17, 2025, from <https://scispace.com/pdf/software-product-quality-metrics-a-systematic-mapping-study-2w1ho43zkb.pdf>.
2. Classification of software quality metrics. (n.d.). Stud.com.ua. Retrieved October 17, 2025, from https://stud.com.ua/128170/informatika/metriki_yakosti_programnogo_zabezpechennya/.
3. The Goal/Question/Metric Paradigm (GQM). (n.d.). Retrieved October 17, 2025, from [https://publica.fraunhofer.de/bitstreams/3c4dda1a-1c71-4a86-8b29-9a164bb31037/](https://publica.fraunhofer.de/bitstreams/3c4dda1a-1c71-4a86-8b29-9a164bb31037/download) download.
4. Lindstrom. (n.d.). The GQM approach is not static but might need refinement. Retrieved October 17, 2025, from https://fileadmin.cs.lth.se/serg/old-serg-dok/docs-masterthesis/62_Lindstrom_draft.pdf.
5. Requirements management measures help organizations to understand, control and assess requirements management process. (2007). ACIT. Retrieved October 17, 2025, from <https://www.acit2k.org/ACIT2007/Proceeding/204.pdf>.
6. Stability ratio of requirements. (н.д.). QAGroup. Retrieved October 17, 2025, from <https://qagroup.com.ua/publications/what-is-metrics/>.
7. Common challenges and pitfalls implementing quality metrics. (n.d.). SafetyCulture. Retrieved October 17, 2025, from <https://safetyculture.com/topics/quality-metrics/>.
8. Role of LLMs and AI in software requirements quality metrics future research. (2025). Frontiers in Computer Science. <https://doi.org/10.3389/fcomp.2025.1519437>.
9. Challenges in quantifying natural language requirements understanding and its impact on metrics. (n.d.). Shelf.io. Retrieved October 17, 2025, from <https://shelf.io/blog/challenges-and-considerations-in-nlp/>.
10. Major challenges of Natural Language Processing. (n.d.). GeeksforGeeks. Retrieved October 17, 2025, from <https://www.geeksforgeeks.org/nlp/major-challenges-of-natural-language-processing/>.
11. Case study compilation - How software testing companies maximize client ROI. (n.d.). MoldStud. Retrieved October 17, 2025, from <https://moldstud.com/articles/p-case-study-compilation-how-software-testing-companies-maximize-client-roi>.
12. Formal specification in software engineering. (n.d.). Medium. Retrieved October 17, 2025, from <https://afzalbadshah.medium.com/formal-specification-in-software-engineering-b66852d96fc6>.
13. Empirical research on requirements quality: a systematic mapping study. (2022). ResearchGate. <https://doi.org/10.1145/1235>.
14. An empirical study of software requirements verification and validation techniques along their mitigation strategies. (2015). ResearchGate. Retrieved October 17, 2025, from https://www.researchgate.net/publication/281645652_An_Empirical_Study_of_Software_Requirements_Verification_and_Validation_Techniques_along_their_Mitigation_Strategies.
15. Requirements impact as recorded in the issue reports. (2024). arXiv. <https://doi.org/10.48550/arXiv.2402.00594>.
16. NLP-assisted software testing_SLR. (n.d.). Queen's University Belfast. Retrieved October 17, 2025, from https://pureadmin.qub.ac.uk/ws/files/219213040/NLP_assisted_software_testing_SLR_Mar_01.pdf.

17. Research on applying ML and NLP techniques to the classification of non-functional requirements and performance metrics. (2020). University of Manchester. Retrieved October 17, 2025, from https://research.manchester.ac.uk/files/183513868/2020_12_16_NLP4RE_Accepted.pdf.
18. Traceability, the ability to trace relevant software artifacts to support reasoning about the quality of the software. (2024). arXiv. <https://doi.org/10.48550/arXiv.2405.10845>.
19. Hileta I. V. Kiselyk R.O., Karnauchov V.A, Komynar T.N. Overview of modern approaches to software quality assessment Scientific Notes No. 1 (70) / 2025. P. 72--82, doi: 10.32403/1998-6912-2025-1-70-72-82.

doi: 10.32403/1998-6912-2025-2-71-82-93

UNDERSTANDING SOFTWARE SPECIFICATIONS AS A FUNDAMENTAL ELEMENT OF QUALITY METRICS

I. V. Hileta, R. O. Kiselyk

*Lviv Polytechnic National University,
12 Stepan Bandera St., Lviv, 79013, Ukraine
e-mail: ivan.v.hileta@lpnu.ua*

Understanding and ensuring the quality of software (SW) specifications (QS) is a critical preventive measure that determines the reliability and validity of subsequent product quality metrics. This study synthesizes current data, confirming that the quality of the final SW is a direct function of the quality of the input specifications. Historical data shows that incomplete and changing requirements are the dominant causes of project failure, shifting the strategic focus from code metrics to requirements metrics as the primary risk factor.

An analysis of key standards such as ISO 25010 (SQuaRE) and IEEE 1061 demonstrates industry and academic consensus on the need for systematic measurement of requirements attributes such as completeness, consistency, and unambiguity. Among these attributes, ambiguity has the largest share of defect subtypes, highlighting its critical impact on interpretation and implementation.

It has been found that the Goal/Question/Metric (GQM) methodology applied to requirements management (RM) practices in the CMMI model provides a structured framework for empirical validation of these metrics. The central practical indicator of specification quality is the Requirements Stability Ratio, whose high value directly correlates with unproductive rework and low functionality scalability.

The report also discusses the contradictions in the application of formal methods, which, although they eliminate ambiguity, can increase complexity and lead to “bugs in the specification itself,” which can account for up to 60% of all input defects. Significant limitations are associated with natural language quantification (NLP), including

problems of lexical and contextual ambiguity, as well as dependence on the quality of training data.

However, the integration of large language models (LLMs) opens up promising avenues for automated defect detection and tracing, reducing post-release defects by up to 40% through early automated validation. The conclusions drawn emphasize a strategic shift in focus to requirements metrics as a key lever for increasing Return on Investment (ROI) in software development.

Keywords: *software specification, requirements attributes, quality metrics, requirements metrics, software quality standards, development cost-effectiveness.*

Стаття надійшла до редакції 06.10.2025.

Received 06.10.2025.