DOI: 10.18372/2310-5461.63.19752
УДК 004.415:004.7

Р. М. Гамрецький
Державний університет «Київський авіаційний інститут»
orcid.org/0000-0001-5514-9783
e-mail: hamretskyi@gmail.com;

В. О. Гнатюк, канд. техн. наук, доцент
Державний університет «Київський авіаційний інститут»
orcid.org/0000-0002-4916-7149
e-mail: viktor.hnatiuk@npp.nau.edu.ua

ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ТА МОДЕЛЕЙ ОЦІНКИ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМАХ

Вступ

Інформаційно-комунікаційні системи (ІКС) сьогодні відіграють ключову роль у забезпеченні ефективної взаємодії між організаціями, клієнтами та технологіями. Висока якість програмного забезпечення (ПЗ), яке є основою цих систем, безпосередньо впливає на надійність, безпеку та продуктивність ІКС. Із зростанням складності ІКС, інтенсифікацією використання хмарних технологій, мікросервісної архітектури та інтеграційних платформ, оцінка якості ПЗ стає надзвичайно важливою для підтримки високих стандартів функціонування систем.

Існуючі методи та моделі оцінки якості ПЗ, такі як ISO/IEC 25010, CMMI, McCall, та інші, дозволяють визначити основні параметри якості, але вони не завжди здатні повною мірою враховувати новітні тенденції та виклики, пов'язані з динамічними змінами у технологічному середовищі. Важливими факторами стають адаптивність, масштабованість, підтримуваність та безпека ПЗ в умовах інтеграції з іншими системами та сервісами, зокрема з використанням технологій штучного інтелекту та кібербезпеки.

З огляду на це, удосконалення існуючих методів та моделей оцінки якості ПЗ є необхідним для підвищення ефективності управління розробкою та експлуатацією ПЗ. Актуальність цієї теми зумовлена необхідністю адаптації методів оцінки до нових викликів, забезпечення більш точних і комплексних інструментів аналізу якості, що відповідають сучасним технологічним стандартам. Це дозволить не тільки підвищити якість ПЗ, але й забезпечити безперервність та надійність роботи ІКС у різних галузях.

Постановка проблеми

ІКС стають дедалі складнішими та динамічнішими, інтегруючи новітні технології, такі як хмарні обчислення, мікросервісна архітектура, інтернет речей (IoT) та штучний інтелект. Ці тенденції створюють нові виклики для забезпечення якості ПЗ, що лежить в основі ІКС. Традиційні методи та моделі оцінки якості ПЗ, як-от ISO/IEC 25010, CMMI, McCall та інші, були розроблені в контексті раніше існуючих технологій та методологій розробки. Вони не завжди враховують специфічні вимоги та ризики, пов'язані з сучасними технологічними трендами.

Проблема полягає в тому, що існуючі методи та моделі оцінки якості ПЗ можуть бути недостатньо ефективними для повноцінного оцінювання якості сучасних ІКС. Зокрема, вони можуть не враховувати такі критично важливі аспекти, як адаптивність до швидких змін, масштабованість у глобальному масштабі, інтеграція з різноманітними сервісами та системами, а також підвищені вимоги до безпеки в умовах зростання кіберзагроз. Відсутність адекватних інструментів оцінки якості може призвести до зниження надійності та безпеки ІКС, що негативно впливає на бізнес-процеси та довіру користувачів.

Таким чином, виникає необхідність у вдосконаленні існуючих методів та моделей оцінки якості ПЗ або розробці нових, які б відповідали сучасним викликам та тенденціям. Це включає інтеграцію нових критеріїв якості, розробку адаптивних методологій та впровадження інструментів, що дозволяють ефективно оцінювати якість ПЗ в умовах динамічного технологічного середовища.

Очікуваним результатом дослідження є виявлення недоліків існуючих методів та моделей оцінки якості програмного забезпечення для ІКС. На основі їх аналізу та порівняння необхідно визначити напрями вдосконалення цих моделей, а також запропонувати нові підходи, які враховують адаптивність, інтеграцію показників безпе-

ки, користувацький досвід та автоматизацію оцінки якості. Це дозволить створити ефективніші інструменти для забезпечення високої надійності, безпеки та продуктивності ІКС.

Аналіз останніх досліджень і публікацій

Дослідження в галузі оцінки якості ПЗ свідчать про активне прагнення до вдосконалення існуючих моделей та методів.

У працях [1, 2] підкреслюється важливість адаптації методів забезпечення якості ПЗ до специфіки телекомунікаційних мереж та ІКС. Зокрема, в [1] аналізуються аспекти забезпечення якості ПЗ у телекомунікаційних мережах, акцентуючи увагу на необхідності врахування особливостей цих систем для підвищення надійності та продуктивності. У [2] досліджуються ключові фактори якості ПЗ та відповідні метрики, що сприяють покращенню процесів забезпечення якості.

Питання безпеки програмного забезпечення набуває особливої актуальності в умовах зростання кіберзагроз та інтеграції ІКС з IoT та кіберфізичними системами. У роботі [3] проведено ґрунтовний огляд безпеки кіберфізичних систем, підкреслюючи необхідність інтеграції безпекових аспектів у моделі оцінки якості ПЗ. Автори аналізують вразливості та загрози, що виникають у зв'язку з широким впровадженням IoT, та пропонують підходи до підвищення захищеності систем.

Проблеми масштабованості та продуктивності ІКС також є предметом досліджень. У роботах [4, 5] аналізуються питання масштабованості та стабільності топології бездротових мереж та IoT-систем. У [4] звертається увага на важливість масштабованості та стабільності в бездротових mesh-мережах для забезпечення якості ІКС. У [5] розглядаються питання продуктивності, надійності та масштабованості для IoT, підкреслюючи необхідність адаптації існуючих моделей оцінки якості до специфіки IoT-технологій.

Деякі дослідження зосереджуються на прогнозуванні дефектів та підвищенні надійності ПЗ. У систематичному огляді [6] аналізуються метрики прогнозування дефектів у ПЗ, що дозволяє вдосконалити процеси тестування та забезпечення якості. Автори підкреслюють важливість вибору адекватних метрик для ефективного прогнозування та запобігання дефектам.

Використання методів штучного інтелекту та машинного навчання для автоматизації процесів оцінки якості ПЗ стає все більш актуальним. У роботі [7] представлено систему VulDeePecker, яка використовує глибоке навчання для виявлення вразливостей у програмному коді, що підвищує ефективність забезпечення безпеки ПЗ.

Користувацький досвід та зручність використання ПЗ є важливими аспектами якості, на що вказують дослідження [8, 9]. У [8] пропонується рамкова структура для вимірювання якості обслуговування мобільних смарт-пристроїв, що дозволяє визначити кращі практики та покращити користувацький досвід. У [9] автори інтегрують аналітичні статистичні моделі, послідовний аналіз патернів та теорію нечітких множин для оцінки надійності мобільних додатків, враховуючи поведінку користувачів та їх взаємодію з системою.

Дослідження [10] зосереджується на безпеці відкритого ПЗ з елементами штучного інтелекту, розглядаючи вплив володіння кодом на загальний рівень безпеки. Підкреслюється важливість відповідального управління кодом та контролю доступу для забезпечення захищеності систем.

Інтеграція багатокритеріальних підходів до оцінки якості ПЗ є предметом досліджень [11–14]. У [11] пропонується нечіткий багатокритеріальний підхід до інтегрованої оцінки якості ПЗ, що дозволяє враховувати невизначеність та суб'єктивність у процесі оцінки. У [12] використовуються багатокритеріальні методи прийняття рішень для пріоритетного вибору моделей якості ПЗ, допомагаючи організаціям обрати найбільш відповідні моделі для їхніх потреб. У роботі [13] розроблено концептуальну основу для потенційних впроваджень методів багатокритеріального прийняття рішень (MCDM) для оцінки якості дизайну. У [14] аналізуються методи та застосування багатокритеріальних моделей у сфері надійності ПЗ, підкреслюючи їх ефективність для покращення процесів забезпечення якості.

Аналіз вказує на необхідність подальших досліджень у напрямку розробки адаптивних та комплексних моделей оцінки якості ПЗ, які б відповідали сучасним технологічним тенденціям та викликам ІКС.

Метою роботи є дослідження існуючих методів та моделей оцінки якості ПЗ в ІКС та визначення методів їх удосконалення.

Виклад основного матеріалу

Провівши аналіз актуальних методів та моделей оцінки якості ПЗ в ІКС можна сформувати їх перелік:

1. Модель ISO/IEC 25010 (Systems and software Quality Requirements and Evaluation).

2. Модель CMMI (Capability Maturity Model Integration).

3. Модель McCall (McCall's Quality Model).
4. Модель Boehm (Boehm's Quality Model).
5. Модель Dromey (Dromey's Quality Model).
6. Модель FURPS/FURPS+.
7. Методологія GQM (Goal-Question-Metric).
8. Модель ISO/IEC 9126.
9. Методологія TQM (Total Quality Management).
10. Методологія Six Sigma.

ISO/IEC 25010 (SQuaRE) – є міжнародним стандартом який описує модель якості продукту та модель якості в процесі виконання програми через характеристики. Модель визначає характеристики та підхарактеристики якості ПЗ як продукту. Вона фокусується на властивостях, які можна оцінити незалежно від контексту використання і які впливають на внутрішню та зовнішню якість продукту [15].

Перелік характеристик які наводяться у стандарті для моделі якості продукту:

1. Функціональна придатність
a. Функціональна повнота – ступінь, до якої всі необхідні функції реалізовані
b. Функціональна коректність – ступінь, до якої функції надають правильні результати
c. Функціональна відповідність – відповідність стандартам, конвенціям або нормативним вимогам
2. Продуктивність ефективності
a. Часова характеристика – швидкість реакції та обробки даних.
b. Використання ресурсів – оптимальне використання апаратних та програмних ресурсів.
c. Ємність – здатність системи обробляти необхідні обсяги даних та користувачів.
3. Сумістність
a. Співіснування – здатність ПЗ працювати разом з іншими системами без негативного впливу.
b. Інтероперабельність – здатність ПЗ взаємодіяти з іншими системами чи компонентами.
4. Зручність використання
a. Результативність – ефективність досягнення користувачем своїх цілей.
b. Навчальність – легкість освоєння ПЗ новими користувачами.
c. Оперативність – інтуїтивність та зрозумілість інтерфейсу.
d. Захист від помилок користувача – запобігання та мінімізація помилок користувача.
e. Доступність – зручність використання для користувачів з обмеженими можливостями.
5. Надійність
a. Зрілість – здатність ПЗ працювати без збоїв.
b. Доступність – готовність системи до роботи в потрібний час.
c. Відмовостійкість – здатність системи продовжувати роботу при виникненні помилок.
d. Відновлюваність – швидкість та легкість відновлення після збоїв.
6. Безпека
a. Конфіденційність – захист даних від несанкціонованого доступу.
b. Цілісність – запобігання несанкціонованим змінам даних.
c. Невідмовність – гарантія того, що дії користувачів не можуть бути заперечені.
d. Підзвітність – відстеження та реєстрація дій користувачів.
e. Аутентифікація – перевірка особи користувача.
7. Супроводжуваність
a. Модульність – розбиття ПЗ на незалежні компоненти.
b. Зручність аналізу – легкість діагностики та аналізу коду.
c. Зручність зміни – простота внесення змін та оновлень.
d. Стабільність – мінімізація негативних наслідків від змін.
e. Тестованість – зручність тестування та виявлення помилок.
8. Переносимість
a. Адаптованість – здатність ПЗ працювати в різних середовищах.
b. Встановлюваність – простота встановлення та налаштування.
c. Сумісність із заміною – можливість заміни іншого ПЗ без втрати функціональності.

Наприклад, функціональна повнота може бути розрахована за наступною формулою:

$$P_{completness} = \frac{N}{T} \cdot 100\,\%,$$

де N – кількість реалізованих функцій, T – кількість запланованих функцій.

Якщо реалізовано 18 із 20 необхідних функцій, то

$$P_{completness} = \frac{18}{20} \cdot 100\,\% = 90\,\%.$$

Результат вказує на те, що заплановані функції були реалізовані на 90 %.

Модель якості в процесі виконання – це концептуальна структура або набір критеріїв, які використовуються для оцінки та забезпечення якості ПЗ під час його фактичного виконання або експлуатації. Ця модель фокусується на динамічних аспектах роботи ПЗ, тобто на тому, як система поводиться в реальних умовах використання.

Перелік характеристик які наводяться у стандарті для моделі якості продукту:

1. Ефективність – точність та повнота, з якою користувачі досягають своїх цілей.

2. Продуктивність – ресурси та зусилля, витрачені користувачем для досягнення цілей.

3. Задоволеність – ступінь задоволення користувача від використання ПЗ.

4. Безпека – захист користувачів від ризиків, пов'язаних з використанням ПЗ.

5. Покриття контексту – здатність ПЗ бути ефективним у різних умовах використання.

Наприклад, ефективність може бути розрахована за наступною формулою:

$$E_{effectiveness} = \frac{N}{T} \cdot 100\,\%,$$

де N – кількість досягнених цілей, T – загальна кількість спроб.

Якщо користувачі точно досягнуть 90 із 100 цілей, то

$$E_{effectiveness} = \frac{90}{100} \cdot 100\,\% = 90\,\%.$$

Ця формула показує точність і повноту досягнення цілей, що свідчить про ефективність користувачів при виконанні завдань.

Загальна якість ПЗ (Q) у наведених моделях може бути розрахована як середньозважене значення оцінок характеристик:

$$Q = \frac{\sum_{i=1}^{n} C_i \cdot w_i}{\sum_{i=1}^{n} w_i},$$

де C_i – оцінка i-ї характеристики (наприклад, функціональної придатності, продуктивності тощо); w_i – вага i-ї характеристики, яка визначається залежно від пріоритетів системи.

Важливо враховувати не тільки оцінку характеристики, а й її вагу, оскільки кожна характеристика може бути менш чи більш важливою для конкретного ПЗ.

Модель CMMI – фокусується на вдосконаленні процесів розробки ПЗ та включає кілька рівнів зрілості організації, що впливають на якість програмних продуктів [16].

CMMI складається з декількох компонентів, які допомагають організаціям покращувати свої процеси:

1. Області процесів (Process Areas): визначені ключові області, що охоплюють різні аспекти діяльності організації.

2. Рівні зрілості (Maturity Levels): п'ять рівнів, які відображають ступінь зрілості процесів в організації.

3. Практики (Practices): конкретні рекомендації та дії, які слід виконувати для досягнення певного рівня зрілості.

П'ять рівнів зрілості CMMI:

1. Рівень 1: Початковий (Initial)

a. Процеси є непередбачуваними, погано контрольованими та реактивними.

b. Успіх залежить від окремих осіб, а не від організаційних процесів.

2. Рівень 2: Керований (Managed)

a. Основні проєкти керуються та контролюються.

b. Процеси плануються та виконуються відповідно до політик; наявні навчені люди та ресурси.

3. Рівень 3: Визначений (Defined)

a. Процеси добре розуміються, описуються в стандартах, процедурах та інструментах.

b. Організація використовує стандартизований набір процесів, які адаптуються для окремих проєктів.

4. Рівень 4: Кількісно керований (Quantitatively Managed)

a. Процеси контролюються за допомогою збору та аналізу даних.

b. Організація встановлює кількісні цілі щодо якості та продуктивності процесів.

5. Рівень 5: Оптимізуючий (Optimizing)

a. Фокус на постійному покращенні процесів через інновації та оптимізацію.

b. Організація активно ідентифікує слабкі місця та впроваджує покращення.

CMMI визначає 22 області процесів, розділені між рівнями зрілості від 2 до 5. Деякі з ключових областей включають:

1. Управління вимогами (Requirements Management).

2. Планування проєкту (Project Planning).

3. Моніторинг та контроль проєкту (Project Monitoring and Control).

4. Управління ризиками (Risk Management).

5. Забезпечення якості процесу та продукту (Process and Product Quality Assurance).

6. Розробка вимірювань та аналіз (Measurement and Analysis).

7. Інтегрована команда проєкту (Integrated Project Management).

8. Технічне рішення (Technical Solution).

9. Верифікація та валідація (Verification and Validation).

Модель CMMI сама по собі не визначає конкретних формул для оцінки рівнів зрілості процесів. Вона є моделлю для вдосконалення процесів, яка базується на принципах управління якістю, процесного підходу та забезпечення ефективності. Однак, для оцінки рівнів зрілості та ефективності процесів організації можна використовувати адаптовані метрики та формули.

В CMMI можна оцінити сам процес: процесна зрілість, дотримання стандартів процесів, ефективність процесу, відхилення процесу, оцінка

ризиків у процесах, покриття метрик процесу, зрілість організації, покриття навчання, дотримання графіку впровадження.

Наприклад, для оцінки зрілості процесів, за CMMI:

$$M_{process} = \frac{N}{T} \cdot 100\,\%,$$

де N – кількість налаштованих процесів, T – загальна кількість необхідних процесів.

Наприклад, для досягнення рівня 3 реалізовано 7 із 10 необхідних процесів, то

$$M_{process} = \frac{7}{10} \cdot 100\,\% = 70\,\%.$$

Модель McCall визначає якість ПЗ через три основні групи факторів: операційні, можливості модифікації та можливості перехідності [17]. Ці фактори розділені на 11 атрибутів якості, які детально описують вимоги до ПЗ:

1. Ревізійна характеристика (Product Revision): здатність до змін
   a. Супроводжуваність: легкість виправлення дефектів або внесення покращень.
   b. Гнучкість: здатність адаптувати ПЗ до змін вимог або середовища.
   c. Тестованість: легкість перевірки ПЗ на відповідність вимогам.
2. Перехідна характеристика (Product Transition): адаптивність до нових середовищ
   a. Переносимість: здатність ПЗ працювати на різних платформах.
   b. Повторне використання: використання компонентів ПЗ в інших системах.
   c. Інтероперабельність: здатність ПЗ взаємодіяти з іншими системами.
3. Операційна характеристика (Product Operations): операційні характеристики
   a. Коректність: відповідність ПЗ специфікаціям та вимогам.
   b. Надійність: здатність ПЗ працювати без збоїв протягом визначеного часу.
   c. Ефективність: оптимальне використання ресурсів при виконанні функцій.
   d. Цілісність: захист ПЗ від несанкціонованого доступу або змін.
   e. Зручність використання: легкість навчання та використання ПЗ.

Для моделі McCall'а може бути застосований принцип розрахунку характеристик такий же як і для стандарту ISO/IEC 25010.

Модель Boehm орієнтована на розуміння очікувань користувачів та їх задоволення через атрибути якості, що охоплюють коректність, надійність, ефективність, зручність використання, підтримуваність та переносність [18].

Модель Боема організована в три рівні ієрархії:

1. Високорівневі основні характеристики якості:
   a. Зручність використання (As-is utility): наскільки ПЗ відповідає потребам користувача в його поточному стані.
   b. Супроводжуваність (Maintainability): легкість внесення змін, виправлення помилок та оновлення ПЗ.
   c. Переносимість (Portability): здатність ПЗ працювати в різних середовищах або на різних платформах.
2. Проміжні характеристики якості, пов'язані з основними:
   a. Переносимість (Portability): здатність ПЗ працювати в різних середовищах або на різних платформах.
   b. Надійність (Reliability): здатність ПЗ виконувати необхідні функції протягом визначеного часу без збоїв.
   c. Ефективність (Efficiency): оптимальне використання ресурсів, таких як час процесора, пам'ять тощо.
   d. Зручність використання (Usability/Human Engineering): легкість, з якою користувачі можуть навчитися та використовувати ПЗ.
   e. Тестованість (Testability): зручність перевірки ПЗ на відповідність вимогам та виявлення дефектів.
   f. Зрозумілість (Understandability): легкість розуміння структури та логіки ПЗ розробниками та іншими зацікавленими сторонами.
   g. Модульність (Modularity): структурована організація ПЗ на незалежні компоненти.
3. Низькорівневі метрики якості:
   a. Незалежність від пристрою (Device Independence).
   b. Самодостатність (Self Containedness).
   c. Точність (Accuracy).
   d. Повнота (Completeness).
   e. Міцність/цілістність (Robustness/Integrity).
   f. Послідовність (Consistency).
   g. Відповідальність (Accountability).
   h. Ефективність пристрою (Device Efficiency).
   i. Доступність (Accessibility).
   j. Комунікабельність (Communicativeness).
   k. Самоописовість (Self-descriptiveness).
   l. Структурованість (Structuredness).
   m. Лаконічність (Conciseness).
   n. Розбірливість (Legibility).
   o. Збільшуваність (Augmentability).

Для моделі Боема може бути застосований принцип розрахунку характеристик такий же як і для стандарту ISO/IEC 25010.

Модель Dromey фокусується на тому, як окремі властивості компонентів ПЗ впливають на загальну якість [19].

Dromey класифікує компоненти ПЗ на три типи:

1. Вхідні компоненти (Input Components) – дані та умови, що входять до системи.

2. Механізми (Mechanisms) – алгоритми та логіка, що обробляють вхідні дані.

3. Вихідні компоненти (Output Components) – результати роботи системи.

Для кожного типу компонентів визначаються властивості, які можуть впливати на якість:

1. Коректність (Correctness):

a. Дійсність вхідних даних.

b. Правильність алгоритмів.

c. Точність вихідних даних.

2. Ефективність:

a. Складність алгоритмів.

b. Використання ресурсів.

3. Зручність використання (Internal):

a. Модульність.

b. Зрозумілість коду.

c. Можливість повторного використання.

4. Зручність використання (Contextual):

a. Відповідність вимогам користувача.

b. Адаптивність до змін у середовищі.

Для моделі Dromey може бути застосований принцип розрахунку характеристик такий же як і для стандарту ISO/IEC 25010.

Модель FURPS/FURPS+ – була розроблена компанією Hewlett-Packard (HP) як спосіб класифікації вимог до ПЗ [20]. Вона допомагає структуровано підходити до визначення та аналізу вимог, розділяючи їх на функціональні та нефункціональні категорії.

Категорії моделі FURPS:

1. Функціональність (Functionality): включає функції, можливості та загальні можливості системи.

a. Функціональні можливості.

b. Безпека.

c. Інтероперабельність.

2. Зручність використання (Usability): оцінює легкість вивчення та використання системи.

a. Зручність інтерфейсу користувача.

b. Консистентність інтерфейсу.

c. Документація.

d. Можливість навчання.

3. Надійність (Reliability): відображає стійкість системи до збоїв та її здатність працювати без помилок.

a. Частота відмов.

b. Відновлюваність.

c. Точність.

d. Прогнозованість.

4. Продуктивність (Performance): оцінює швидкість та ефективність роботи системи.

a. Час відгуку.

b. Пропускна здатність.

c. Ефективність використання ресурсів.

d. Масштабованість.

5. Супроводжуваність (Supportability): відображає легкість підтримки та модифікації системи.

a. Тестованість.

b. Модульність.

c. Гнучкість.

d. Розширюваність.

e. Можливість локалізації.

FURPS+ додає до моделі додаткові категорії нефункціональних вимог:

1. Обмеження дизайну (Design Constraints): обмеження на дизайн системи, такі як стандарти, правила, технології.

2. Вимоги до реалізації (Implementation Requirements): специфічні вимоги до реалізації, що впливають на розробку.

3. Вимоги до інтерфейсу (Interface Requirements): вимоги щодо взаємодії системи з іншими системами чи компонентами.

4. Фізичні вимоги (Physical Requirements): вимоги, пов’язані з апаратними або фізичними аспектами.

Модель FURPS/FURPS+ є ефективним інструментом для всебічного визначення та управління вимогами до ПЗ. Вона допомагає забезпечити високу якість продукту, задовольняючи як функціональні потреби, так і нефункціональні аспекти, що впливають на загальний користувацький досвід і ефективність системи.

Методологія GQM допомагає оцінювати якість ПЗ шляхом визначення цілей (Goals), формулювання питань (Questions) для аналізу якості та вимірювання метрик (Metrics), що відповідають на ці питання [21].

GQM базується на трирівневому підході:

1. Рівень цілі (Goal Level)

a. Об’єкт вимірювання: Що саме ми оцінюємо? (продукт, процес або ресурс)

b. Мета вимірювання: Навіщо ми це оцінюємо? (для покращення, контролю, розуміння тощо)

c. Точка зору: З якої перспективи ми проводимо оцінку? (користувач, розробник, менеджер)

d. Контекст: За яких умов проводиться оцінка? (проект, організація, технологія)

Приклад:

– Об’єкт: Процес тестування ПЗ;

– Мета: Покращити ефективність процесу тестування;

– Точка зору: Менеджер з якості;

– Контекст: У межах великого проєкту з розробки ПЗ для фінансового сектору.

2. Рівень питань (Question Level).

a. Питання формулюються таким чином, щоб допомогти зрозуміти, чи досягаються встановлені цілі.

b. Питання повинні бути конкретними та спрямованими на різні аспекти цілі.

Приклад:

– який відсоток виявлених дефектів на етапі тестування?

– який середній час, необхідний для виявлення та виправлення дефекту?

– яка частота повторних дефектів (дефектів, які виникають знову після виправлення)?

3. Рівень метрик (Metric Level)

a. Метрики надають кількісні відповіді на питання.

b. Можуть бути об'єктивними (наприклад, кількість дефектів) або суб'єктивними (наприклад, задоволеність команди).

Приклад:

– метрика для першого питання: Кількість дефектів, виявлених під час тестування, поділена на загальну кількість дефектів;

– метрика для другого питання: Середній час між виявленням дефекту та його виправленням;

– метрика для третього питання: Кількість повторних дефектів, поділена на загальну кількість виправлених дефектів.

В спрощеному варіанті застосування методології може мати наступний вигляд:

– ціль – підвищити продуктивність системи;

– питання – як швидко система обробляє запити?

– метрика – середній час відповіді.

$$T_{response} = \frac{T}{N},$$

де T – загальний час відповіді, N – кількість запитів.

Наприклад, якщо система обробила 100 запитів за 2000 мс, то

$$T_{response} = \frac{2000}{100} = 20ms.$$

Методологія GQM адаптивна, і формули метрик створюються індивідуально залежно від цілей. Головний акцент у моделі GQM – це визначення правильних питань і вибір метрик, що відповідають бізнес-цілям.

ISO/IEC 9126 – попередник стандарту ISO/IEC 25010, який визначає менший набір характеристик але також описує дві моделі [22].

Модель якості продукту:

1. Функціональність (Functionality)

a. Придатність (Suitability): відповідність функцій ПЗ встановленим завданням та цілям.

b. Точність (Accuracy): здатність ПЗ надавати правильні або погоджені результати чи ефекти.

c. Взаємодія (Interoperability): здатність ПЗ взаємодіяти з однією або більше зазначеними системами.

d. Відповідність (Compliance): відповідність ПЗ стандартам, конвенціям або нормативним вимогам, пов'язаним з функціональністю.

e. Безпека (Security): здатність ПЗ захищати інформацію та дані, щоб забезпечити, що доступ до них мають лише авторизовані особи.

2. Надійність (Reliability)

a. Зрілість (Maturity): здатність ПЗ уникати збоїв, пов'язаних з дефектами.

b. Стійкість до відмов (Fault Tolerance): здатність ПЗ підтримувати заданий рівень продуктивності у випадку збоїв або порушень.

c. Відновлюваність (Recoverability): здатність ПЗ відновлюватися після збоїв та відновлювати дані, що були пошкоджені.

d. Доступність (Availability): здатність ПЗ бути доступним для використання, коли це потрібно.

3. Зручність використання (Usability)

a. Зрозумілість (Understandability): легкість, з якою користувач може зрозуміти концепції та застосування ПЗ.

b. Навчальність (Learnability): легкість, з якою користувач може навчитися використовувати ПЗ.

c. Оперативність (Operability): легкість та ефективність керування ПЗ користувачем.

d. Привабливість (Attractiveness): ступінь, до якого ПЗ є привабливим для користувача.

e. Відповідність (Compliance): відповідність ПЗ стандартам або конвенціям, пов'язаним з зручністю використання.

4. Ефективність (Efficiency)

a. Часова характеристика (Time Behaviour): час відгуку та обробки, а також пропускна здатність ПЗ.

b. Використання ресурсів (Resource Utilization): обсяги та типи ресурсів, що використовуються ПЗ.

c. Відповідність (Compliance): відповідність ПЗ стандартам або конвенціям, пов'язаним з ефективністю.

5. Супроводжуваність (Maintainbility)

a. Аналізованість (Analyzability): легкість діагностики дефектів або причин збоїв.

b. Змінюваність (Changeability): легкість внесення модифікацій, усунення дефектів або адаптації ПЗ.

c. Стабільність (Stability): здатність ПЗ мінімізувати небажані ефекти від змін.
d. Тестованість (Testability): легкість, з якою можна протестувати ПЗ після змін.
e. Відповідність (Compliance): відповідність ПЗ стандартам або конвенціям, пов'язаним з супроводжуваністю.
6. Переносимість (Portability)
a. Адаптованість (Adaptability): здатність ПЗ бути адаптованим до різних середовищ без застосування інших дій або засобів.
b. Встановлюваність (Installability): легкість встановлення ПЗ в певному середовищі.
c. Співіснування (Co-Existence): здатність ПЗ співіснувати з іншими незалежними продуктами в загальному середовищі.
d. Замінність (Replaceability): здатність ПЗ замінювати інше ПЗ з аналогічною функціональністю в тому самому середовищі.
e. Відповідність (Compliance): відповідність ПЗ стандартам або конвенціям, пов'язаним з переносимістю.

Модель якості в процесі виконання:

1. Ефективність (Effectiveness): точність та повнота, з якою користувачі досягають своїх цілей.

2. Продуктивність (Productivity): ресурси, витрачені користувачами для досягнення цілей.

3. Безпека (Safety): здатність ПЗ досягати прийнятного рівня ризику шкоди.

4. Задоволеність (Satisfaction): ступінь задоволеності користувачів при використанні ПЗ.

Методологія TQM – концепція загального управління якістю, яка застосовується не тільки до виробничих процесів, але й до розробки ПЗ, з акцентом на задоволення потреб клієнтів через постійне покращення якості [23].

Основні принципи TQM:

1. Орієнтація на клієнта:
a. Задоволення потреб клієнта є найвищим пріоритетом.
b. Розуміння та перевищення очікувань клієнтів.
2. Залучення всіх співробітників
a. Командна робота та співпраця між співробітниками.
b. Підвищення кваліфікації та навчання персоналу.
c. Кожен співробітник відповідає за якість своєї роботи.
3. Підхід до процесів
a. Управління процесами замість управління результатами.
b. Ідентифікація та оптимізація ключових процесів.
4. Системний підхід до управління
a. Розуміння організації як єдиної системи, де всі процеси взаємопов'язані.
b. Координація та інтеграція процесів для досягнення цілей.
5. Постійне покращення
a. Безперервний розвиток та вдосконалення процесів, продуктів та послуг.
b. Використання методів, таких як PDCA (Plan-Do-Check-Act).
6. Прийняття рішень на основі фактів
a. Використання даних та аналізу для прийняття обґрунтованих рішень.
b. Моніторинг та вимірювання показників якості.
7. Взаємовигідні відносини з постачальниками
a. Партнерство з постачальниками та підрядниками.
b. Спільна робота над покращенням якості матеріалів та послуг.
8. Лідерство
a. Відданість керівництва принципам якості.
b. Встановлення чітких цілей та напрямків розвитку.

Інструменти та методи TQM:

1. PDCA (Цикл Демінга):
a. Plan (Плануй): визначення проблеми, встановлення цілей та планування змін.
b. Do (Роби): впровадження запланованих змін у невеликому масштабі.
c. Check (Перевіряй): оцінка результатів та ефективності змін.
d. Act (Дій): впровадження успішних змін на постійній основі або повернення до планування.
2. Інструменти контролю якості:
a. Діаграми причинно-наслідкових зв'язків (Діаграми Ісікави).
b. Контрольні листи.
c. Гістограми.
d. Діаграми Парето.
e. Статистичний контроль процесів (SPC).
f. Діаграми розсіювання.

TQM не визначає конкретних математичних формул, але використовує набір метрик та інструментів для оцінювання якості, таких як статистичний контроль якості (SPC), аналіз дефектів та індекси ефективності.

Наприклад, рівень дефектів можна розраховувати як

$$D_{rate} = \frac{D}{U} \cdot 100\ \%,$$

де D – кількість дефектів, U – загальна кількість виготовлених одиниць.

Якщо із 1000 виготовлених одиниць виявлено 25 дефектів, то

$$D_{rate} = \frac{25}{1000} \cdot 100\,\% = 0{,}25\,\%.$$

Чим менший відсоток – тим кращий результат.

Методологія Six Sigma базується на методології DMAIC (Define, Measure, Analyze, Improve, Control), що дозволяє оцінювати і покращувати якість ПЗ через статистичний контроль та зменшення дефектів [24]. Основна мета – досягти рівня якості, при якому кількість дефектів не перевищує 3,4 на мільйон можливостей**.**

Методологія Six Sigma:

1. Визначення (Define): ідентифікація проблеми та встановлення цілей.
2. Вимірювання (Measure): збір даних та визначення поточного стану.
3. Аналіз (Analyze): виявлення кореневих причин проблем.
4. Покращення (Improve): розробка та впровадження рішень.
5. Контроль (Control): закріплення покращень та моніторинг результатів.

Основні принципи Six Sigma:

1. Орієнтація на клієнта: фокус на задоволенні потреб та очікувань клієнтів.
2. Управління на основі даних та фактів: використання статистичних даних для прийняття рішень.
3. Проактивне управління процесами: постійний моніторинг та покращення процесів.
4. Співпраця та залучення персоналу: участь усіх рівнів організації в процесі покращення.
5. Прагнення до досконалості: постійне вдосконалення та усунення причин дефектів.

Для розрахунку згідно цієї методології використовується набір статистичних методів і метрик для оцінювання ефективності процесів.

1. Дефекти на мільйон можливостей (*Defects Per Million Opportunities, DPMO*)

$$DPMO = \frac{D}{T} \cdot 1000000,$$

де D – кількість дефектів, T – кількість можливостей для дефектів у продукції.

Наприклад, якщо у 10,000 одиниць продукції знайдено 50 дефектів із 20 можливими дефектами на одиницю, то

$$DPMO = \frac{50}{10000 \times 20} \cdot 1000000 = 250\,.$$

2. Рівень Sigma (Sigma Level)

$$\sigma_{level} = \Phi^{-1}\left(1 - \frac{DPMO}{1000000}\right) + 1{,}5\,,$$

де Φ^{-1} — обернена функція стандартного нормального розподілу (*Z*-оцінка); 1,5 – коригування для типових зсувів процесу.

Наприклад, якщо $DPMO = 250$, то σ_{level} приблизно дорівнює 5.5.

3. Коефіцієнт якості процесу (Process Capability Index)

$$C_p = \frac{USL - LSP}{6\sigma}\,,$$

де USL – верхня допустима межа (*Upper Specification Limit*), *LSL* – нижня допустима межа (*Lower Specification Limit*), σ – стандартне відхилення.

Наприклад, якщо USL = 100, LSL = 80, а стандартне відхилення = 3, то

$$C_p = \frac{100 - 80}{6 \cdot 3} = 1{,}11.$$

4. Коефіцієнт відповідності процесу (Process Performance Index)

$$C_{pk} = min\left(\frac{USL - \mu}{3\sigma}, \frac{\mu - LSP}{3\sigma}\right),$$

де μ – середнє значення процесу, σ – стандартне відхилення.

Наприклад, якщо *USL* = 100, *LSL* = 80, μ = 90, σ = 3, то

$$C_{pk} = min\left(\frac{100 - 90}{3 \cdot 3}, \frac{90 - 80}{3 \cdot 3}\right) = min\left(\frac{10}{9}, \frac{10}{9}\right) = 1{,}11$$

.

5. Рівень виконання процесу (*Process Yield*)

$$Y_{process} = \left(1 - \frac{D}{T}\right) \cdot 100\,\%\,,$$

де D – кількість дефектів, T – загальна кількість одиниць.

Наприклад, якщо з 10000 одиниць знайдено 50 дефектів, то

$$Y_{process} = \left(1 - \frac{50}{10000}\right) \cdot 100\,\% = 99{,}5\,\%\,.$$

6. Коефіцієнт ефективності процесу (*Rolled Throughput Yield, RTY*). *RTY* враховує втрати на кожному етапі процесу:

$$RTY = \prod_{i=1}^{n}\left(1 - \frac{D_i}{U_i}\right),$$

де D_i – кількість дефектів, U_i – загальна кількість одиниць.

Наприклад, якщо є три етапи з такими дефектами: 5/1000, 10/990, 8/980, то

$$Y_{process} = \left(1 - \frac{5}{1000}\right) \cdot \left(1 - \frac{10}{990}\right) \cdot \left(1 - \frac{8}{980}\right) =$$

$$= 0{,}995 \cdot 0{,}9899 \cdot 0{,}9918 \cdot 0{,}976.$$

7. Індекс витрат через низьку якість (*Cost of Poor Quality, COPQ*)

$$COPQ = \frac{C}{T} \cdot 100\,\%,$$

де C – вартість дефектів, T – загальна вартість.

Наприклад, якщо дефекти обійшлися в \$50,000 із загального доходу \$1000000, то

$$COPQ = \frac{50000}{1000000} \cdot 100\,\% = 5\,\%.$$

8. Загальна продуктивність процесу (*Overall Equipment Effectiveness, OEE*). Вимірює ефективність обладнання на основі доступності, продуктивності й якості.

$$OEE = A \cdot P \cdot Q,$$

де A – доступність, P – продуктивність, Q – якість.

Наприклад, якщо доступність = 90 %, продуктивність = 95 %, якість = 98 %, то

$$OEE = 0{,}9 \cdot 0{,}95 \cdot 0{,}98 = 0{,}837 (83{,}7\,\%).$$

Ці формули дозволяють вимірювати ефективність процесів та рівень якості в межах Six Sigma. Вони підтримують ідеологію зменшення кількості дефектів, поліпшення продуктивності й оптимізації процесів.

Визначимо критерії порівняння методів та моделей між собою у контексті ІКС:

1. Надійність (Reliability): багато моделей оцінки якості включають цей критерій, оскільки надійність є ключовим показником для будь-якого ПЗ. Зазвичай для цього використовуються метрики відмов, час до відмови (MTTF), час відновлення (MTTR), які моделюються та оцінюються за допомогою моделей, як-от ISO/IEC 25010 або FURPS.

2. Продуктивність (Performance): цей критерій можна оцінити у більшості моделей через аналіз швидкості відгуку, затримок, продуктивності на навантаження. Він є особливо важливим у моделі ISO/IEC 25010, яка має продуктивність як окремий фактор.

3. Юзабіліті (Usability): існують конкретні моделі та стандарти, такі як ISO/IEC 25010 і FURPS, де юзабіліті охоплює субкритерії, як-от зрозумілість, навченість і зручність інтерфейсу.

4. Безпека (Security): цей критерій часто застосовується через стандарти ISO/IEC 25010 і Common Criteria (ISO/IEC 15408), які дають змогу оцінювати захищеність систем від внутрішніх та зовнішніх загроз.

5. Масштабованість (Scalability): масштабованість може бути врахована через продуктивність і архітектурну модульність. Хоча вона не завжди є окремим критерієм, її можна інтегрувати в оцінки через моделі, як-от ATAM (Architecture Tradeoff Analysis Method).

6. Модульність (Modularity): модульність враховується через структурованість і гнучкість архітектури. Модель FURPS+ включає цей критерій у додаткових характеристиках для архітектурних оцінок.

7. Інтероперабельність (Interoperability): це одна з характеристик, що безпосередньо включена у ISO/IEC 25010 і описує здатність ПЗ до взаємодії з іншими системами та компонентами.

8. Підтримка та обслуговування (Maintainability): важливий критерій, який охоплює такі підкритерії, як гнучкість, тестованість і доступність до документації. ISO/IEC 25010 і FURPS+ також включають цей критерій.

9. Гнучкість (Flexibility): гнучкість частково охоплена моделлю FURPS+ через критерії надійності та обслуговуваності, хоча її іноді розглядають окремо в більш спеціалізованих моделях, як-от ATAM.

10.Ефективність з точки зору витрат (Cost-effectiveness): це специфічний критерій, який може бути врахований за допомогою методів ROI (Return on Investment) або TCO (Total Cost of Ownership), а також деяких моделей на кшталт CMMI де якість розглядається у взаємозв'язку з витратами.

Визначені критерії базуються на існуючих підходах, але акцентують увагу на їх актуальності для сучасних ІКС.

Перераховані критерії є фундаментальними для сучасних ІКС, оскільки вони відображають ключові аспекти ефективної роботи програмного забезпечення в умовах постійної технологічної еволюції.

На основі сформованих критеріїв створено таблицю, що дозволяє порівняти методи й моделі оцінки якості та визначити, які з них найбільше відповідають сучасним вимогам ІКС.

У комірках відображено наявність або відсутність кожного критерію в певній моделі.

Позначення: ✅ – критерій враховується, ❌ – критерій не враховується, ⚠ – критерій враховується частково.

В існуючих методах та моделях оцінки якості ПЗ для ІКС є кілька напрямків, які можна покращити. Ось деякі з них:

1. Адаптивність до нових технологій та вимог. Сучасні ІКС постійно змінюються, з'являються нові технології, такі як хмарні обчислення, IoT, 5G та інші. Багато існуючих методів та

моделей оцінки якості недостатньо адаптовані до цих нових технологій та не враховують нові параметри якості, що можуть виникати в таких умовах (наприклад, низька затримка передачі даних в 5G або кібербезпека в IoT).

Таблиця 1

Матриця порівняння методів та моделей

Модель/Метод Критерій	ISO/IEC 25010 (SQuaRE)	CMMI	McCall	Boehm	Dromey	FURPS/ FURPS+	GQM	ISO/IEC 9126	TQM	Six Sigma
Надійність	✅	⚠	✅	✅	✅	✅	❌	✅	⚠	⚠
Продуктивність	✅	❌	⚠	⚠	⚠	✅	❌	⚠	⚠	✅
Юзабіліті	✅	⚠	✅	✅	⚠	✅	❌	✅	⚠	❌
Безпека	✅	⚠	❌	⚠	❌	⚠	❌	⚠	⚠	❌
Масштабованість	⚠	⚠	❌	⚠	❌	⚠	⚠	❌	✅	⚠
Модульність	⚠	✅	✅	⚠	✅	⚠	❌	✅	✅	⚠
Інтероперабельність	✅	⚠	❌	❌	⚠	⚠	❌	⚠	✅	⚠
Підтримка та обслуговування	✅	✅	✅	⚠	✅	✅	✅	✅	✅	⚠
Гнучкість	⚠	✅	⚠	✅	✅	⚠	⚠	⚠	⚠	✅
Ефективність з точки зору витрат	⚠	✅	❌	❌	⚠	❌	⚠	❌	✅	✅

Покращення: розробка адаптивних методів та моделей, що можуть динамічно враховувати нові технологічні вимоги та стандарти.

2. Інтеграція показників безпеки. У багатьох методах та моделях оцінки якості ПЗ для ІКС недостатньо уваги приділяється показникам безпеки. У зв'язку з ростом кіберзагроз, питання захищеності ПЗ стає одним з ключових.

Покращення: інтеграція специфічних показників кібербезпеки в загальну систему оцінки якості ПЗ.

3. Гнучкість у врахуванні потреб кінцевих користувачів. Існуючі методи та моделі часто зосереджуються на технічних характеристиках, таких як продуктивність, надійність і масштабованість, але не завжди приділяють належну увагу якості взаємодії з кінцевими користувачами, їхньому досвіду використання системи (UX).

Покращення: інтеграція показників якості з погляду користувачів, зокрема вимірювання зручності користування, часу відгуку системи для користувача, доступності та задоволення потреб.

4. Автоматизація та підтримка штучного інтелекту. Процеси оцінки якості ПЗ часто потребують значних людських ресурсів. Однак, існуючі методи та моделі могли б отримати більше користі від використання методів автоматизації та штучного інтелекту для виявлення проблем і оптимізації процесів тестування.

Покращення: розробка методів та моделей, які використовують методи машинного навчання та AI для автоматичного оцінювання якості ПЗ та прогнозування потенційних проблем на етапі розробки.

5. Оцінка надійності та масштабованості у реальних умовах. Тестування якості ПЗ часто виконується у лабораторних умовах, що може не відображати реальні умови експлуатації. Особливо це стосується ІКС, які можуть працювати під різними навантаженнями, у різних мережевих умовах або бути частиною складної гетерогенної архітектури.

Покращення: створення методів та моделей, які можуть точніше оцінювати надійність і масштабованість ПЗ в умовах реальної експлуатації, враховуючи реальні сценарії використання та мережеві умови.

6. Комплексні методи та моделі з урахуванням міждисциплінарного підходу. ІКС включають різні аспекти – від мережевих технологій до програмної архітектури, що ускладнює оцінку якості. Багато існуючих методів та моделей зосереджені на вузьких аспектах (наприклад, тільки на продуктивності або безпеці), тоді як комплексна оцінка може вимагати міждисциплінарного підходу.

Покращення: створення комплексних методів та моделей, які охоплюють не лише технічні аспекти, але й враховують організаційні, економічні та користувацькі фактори.

7. Врахування енергоспоживання та екологічних факторів. У сучасних умовах важливо враховувати екологічні аспекти роботи ПЗ, зокрема

енергоспоживання. Для ІКС це може мати суттєвий вплив, особливо якщо система масштабована або використовує багато обчислювальних ресурсів.

Покращення: розробка методів та моделей, що включають метрики енергоспоживання та екологічної стійкості.

Ці покращення можуть допомогти зробити методи та моделі оцінки якості ПЗ більш актуальними та корисними для сучасних і майбутніх ІКС.

Висновки

У ході дослідження було проаналізовано існуючі методи та моделі оцінки якості ПЗ в ІКС. Було сформовано список критеріїв яким мали би відповідати сучасні методи та моделі, а також порівняно їх за цими критеріями. В результаті чого було виявлено, що існуючі методи та моделі потребують вдосконалення з метою відповідності сучасним технологічним викликам.

Основними напрямами покращення є:

1. Адаптація до новітніх технологій: включення до моделей критеріїв, що враховують особливості хмарних обчислень, мікросервісної архітектури, IoT та штучного інтелекту.

2. Інтеграція аспектів безпеки: розробка метрик та методів оцінки, що дозволяють ефективно оцінювати безпеку ПЗ в умовах зростання кіберзагроз.

3. Фокус на користувацькому досвіді: врахування показників юзабіліті та задоволеності користувачів для підвищення якості взаємодії з системою.

4. Автоматизація оцінки якості: впровадження інструментів штучного інтелекту та машинного навчання для підвищення ефективності процесів тестування та оцінки якості.

5. Комплексний міждисциплінарний підхід: створення моделей, що охоплюють технічні, організаційні, економічні та екологічні аспекти якості ПЗ.

Подальші дослідження в цій галузі мають бути спрямовані на розробку універсальних та адаптивних методів та моделей оцінки якості ПЗ, які б враховували сучасні тенденції та забезпечували високу надійність, безпеку та ефективність ІКС.

ЛІТЕРАТУРА

[1] Okumoto K., Mijumbi R., Asthana A. Software Quality Assurance. *Telecommunication Networks – Trends and Developments.* IntechOpen, 2019. DOI: 10.5772/intechopen.79839.

[2] Lee M.-C. Software quality factors and software quality metrics to enhance software quality assurance. *British Journal of Applied Science & Technology*. 2014. Vol. 4, No. 21. P. 3069–3095.

[3] Humayed A., Lin J., Li F., Luo B. Cyber-Physical Systems Security–A Survey. *IEEE Internet of Things Journal*. 2017. Vol. 4, No. 6. P. 1802–1831. DOI: 10.1109/JIOT.2017.2703172.

[4] Sampaio S., Souto P., Vasques F. A review of scalability and topological stability issues in IEEE 802.11s wireless mesh networks deployments. *International Journal of Communication Systems*. 2016. Vol. 29. P. 671–693. DOI: 10.1002/dac.2929.

[5] Luntovskyy A., Globa L. Performance, Reliability and Scalability for IoT. 2019 International Conference on Information and Digital Technologies (IDT). Zilina, Slovakia, 2019. P. 316–321. DOI: 10.1109/DT.2019.8813679.

[6] Radjenović D., Heričko M., Torkar R., Živkovič A. Software Fault Prediction Metrics: A Systematic Literature Review. *Information and Software Technology*. 2013. Vol. 55, No. 8. P. 1397–1418. DOI: 10.1016/j.infsof.2013.02.009.

[7] Li Z. et al. VulDeePecker: A Deep Learning-Based System for Vulnerability Detection. Proceedings of the 25th Annual Network and Distributed System Security Symposium (NDSS). 2018. DOI: 10.14722/ndss.2018.23158.

[8] Desmal A., Madan Z. Measuring the Service Quality of Mobile Smart Devices: A Framework for Best Practices. IntechOpen, 2024. DOI: 10.5772/intechopen.113993.

[9] Shmatko O., Kolomiitsev O., Fedorchenko V., Mykhailenko I., Tretiak V. Integrating analytical statistical models, sequential pattern mining, and fuzzy set theory for advanced mobile app reliability assessment. *Innovative Technologies and Scientific Solutions for Industries*. 2023. No. 26. P. 78–86. DOI: 10.30837/ITSSI.2023.26.078.

[10] Wen J. et al. Code Ownership in Open-Source AI Software Security. arXiv preprint. 2023. URL: https://arxiv.org/abs/2312.10861.

[11] Challa J. S., Paul A., Dada Y., Nerella V., Srivastava P. R., Singh A. P. Integrated Software Quality Evaluation: A Fuzzy Multi-Criteria Approach. Journal of Information Processing Systems. 2011. Vol. 7, No. 3. P. 473–518. DOI: 10.3745/JIPS.2011.7.3.473.

[12] Verma A., Agarwal A., Rathore M., Bisht S., Singh D. Preferential Selection of Software Quality Models Based on a Multi-Criteria Decision-Making Approach. *International Journal of Software Innovation (IJSI)*. 2023. Vol. 11, No. 1. P. 1–13. DOI: 10.4018/IJSI.315739.

[13] Harputlugil T., Prins M., Gültekin A., Topcu I. Conceptual framework for potential implementations of multi criteria decision making (MCDM) methods for design quality assessment. 2011.

[14] Mishra A., Linh N. T. D., Bhardwaj M., Pinto C. M. A. Multi-Criteria decision modelцs in software reliability: Methods and Applications. Information Technology, Management and

Operations Research Practices. 2022. DOI: 10.1201/9780367816414.
[15] Systems and software engineering–Systems and software Quality Requirements and Evaluation (SQuaRE)–System and software quality models: ISO/IEC 25010:2023. Geneva: International Organization for Standardization, 2023.
[16] CMMI for Development, Version 2.0 / CMMI Institute. CMMI Institute, 2018.
[17] McCall J. A., Richards P. K., Walters G. F. Factors in Software Quality. General Electric, 1977.
[18] Boehm B. W., Brown J. R., Kaspar H. Characteristics of software quality. North-Holland Pub. Co., 1978.
[19] Dromey R. G. A model for software product quality. IEEE Transactions on Software Engineering. 1995. Vol. 21, No. 2. P. 146–162. DOI: 10.1109/32.345830.
[20] Ziemek M. Documenting non-functional requirements using FURPS+ [Electronic resource]. URL: https://marcinziemek.com/blog/content/ articles/ 8/article_en.html.
[21] Basili V. R., Caldiera G., Rombach H. D. The Goal Question Metric Approach. Encyclopedia of Software Engineering. 1994. P. 528–532.
[22] Software engineering–Product quality–Part 1: Quality model: ISO/IEC 9126-1:2001. Geneva: International Organization for Standardization, 2001.
[23] Oakland J. S. Total Quality Management: Text with Cases. Routledge, 2003. 496 p.
[24] Pyzdek T., Keller P. A. The Six Sigma Handbook. 4th ed. McGraw-Hill, 2014.

Гамрецький Р. М., Гнатюк В. О.

ОГЛЯД ТА АНАЛІЗ МЕТОДІВ ТА МОДЕЛЕЙ ОЦІНКИ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМАХ

Сучасні інформаційно-комунікаційні системи (ІКС) є фундаментом цифрового суспільства, забезпечуючи ефективну взаємодію між організаціями, клієнтами та технологіями. Актуальність дослідження зумовлена зростаючою складністю ІКС та необхідністю забезпечення високої якості програмного забезпечення (ПЗ), яке є їх основою. Традиційні моделі оцінки якості ПЗ, такі як ISO/IEC 25010, CMMI, McCall та інші, не завжди відповідають сучасним технологічним викликам, оскільки вони були розроблені в контексті раніше існуючих технологій і не враховують специфіку хмарних обчислень, мікросервісної архітектури, IoT та штучного інтелекту.

Постановка проблеми полягає в тому, що існуючі моделі оцінки якості ПЗ можуть бути недостатньо ефективними для повноцінного оцінювання якості сучасних ІКС. Вони не враховують критично важливі аспекти, такі як адаптивність до швидких змін, масштабованість у глобальному масштабі, інтеграція з різноманітними сервісами та системами, а також підвищені вимоги до безпеки в умовах зростання кіберзагроз. Це може призвести до зниження надійності та безпеки ІКС, що негативно впливає на бізнес-процеси та довіру користувачів.

Шляхи вирішення проблеми включають вдосконалення існуючих моделей або розробку нових, які б відповідали сучасним викликам та тенденціям. У статті проведено аналіз останніх досліджень і публікацій, де підкреслюється необхідність адаптації методів забезпечення якості ПЗ до специфіки телекомунікаційних мереж та ІКС, інтеграції безпекових аспектів, використання методів штучного інтелекту для автоматизації процесів оцінки якості, а також врахування користувацького досвіду та зручності використання.

Результати дослідження включають детальний аналіз існуючих методів та моделей оцінки якості ПЗ, таких як ISO/IEC 25010, CMMI, McCall, Boehm, Dromey, FURPS/FURPS+, GQM, ISO/IEC 9126, TQM та Six Sigma. Проведено порівняння цих моделей за критеріями, що є важливими для сучасних ІКС, включаючи надійність, продуктивність, юзабіліті, безпеку, масштабованість та інші. Виявлено, що жодна з існуючих моделей не повністю відповідає всім сучасним вимогам, що підтверджує необхідність їх удосконалення.

У висновках зазначено, що основними напрямами покращення є адаптація до новітніх технологій, інтеграція аспектів безпеки, фокус на користувацькому досвіді, автоматизація оцінки якості та застосування комплексного міждисциплінарного підходу. Рекомендовано розробку універсальних та адаптивних моделей оцінки якості ПЗ, які б забезпечували високу надійність, безпеку та ефективність ІКС в умовах динамічного технологічного середовища.

Ключові слова: інформаційно-комунікаційні системи, оцінка якості програмного забезпечення, методи та моделі якості, адаптація до сучасних технологій, кібербезпека.

Hamretskyi R., Gnatyuk V.

REVIEW AND ANALYSIS OF METHODS AND MODELS FOR ASSESSING THE QUALITY OF SOFTWARE IN INFORMATION AND COMMUNICATION SYSTEMS

Modern information and communication systems (ICS) are the foundation of the digital society, ensuring effective interaction between organizations, clients, and technologies. The relevance of this research is driven by the increasing complexity of ICS and the necessity to ensure high-quality software (SW) that underpins these systems. Traditional software quality evaluation models, such as ISO/IEC 25010, CMMI, McCall, and others, do not always meet current

technological challenges, as they were developed in the context of earlier technologies and do not account for the specifics of cloud computing, microservice architecture, IoT, and artificial intelligence.

The problem statement highlights that existing software quality evaluation models may be insufficient for fully assessing the quality of modern ICS. They do not consider critically important aspects such as adaptability to rapid changes, global scalability, integration with various services and systems, and heightened security requirements amid growing cyber threats. This may lead to decreased reliability and security of ICS, negatively affecting business processes and user trust.

Solutions to the problem involve improving existing models or developing new ones that correspond to current challenges and trends. The article analyzes recent research and publications, emphasizing the need to adapt software quality assurance methods to the specifics of telecommunication networks and ICS, integrate security aspects, utilize artificial intelligence methods for automating quality assessment processes, and consider user experience and usability.

The research results include a detailed analysis of existing software quality evaluation methods and models, such as ISO/IEC 25010, CMMI, McCall, Boehm, Dromey, FURPS/FURPS+, GQM, ISO/IEC 9126, TQM, and Six Sigma. These models are compared based on criteria important for modern ICS, including reliability, performance, usability, security, scalability, and others. It was found that none of the existing models fully meet all current requirements, confirming the need for their improvement.

The conclusions state that the main directions for improvement are adaptation to new technologies, integration of security aspects, focus on user experience, automation of quality assessment, and application of a comprehensive interdisciplinary approach. The development of universal and adaptive software quality evaluation models is recommended to ensure high reliability, security, and efficiency of ICS in a dynamic technological environment.

Keywords: information and communication systems, software quality evaluation, quality methods and models, adaptation to modern technologies, cybersecurity.

Стаття надійшла до редакції 02.11.2024 р.
Прийнято до друку 11.12.2024 р.