

1. Функціональність (*Functionality*) — властивість ПС виконувати функції відповідно до встановлених і очікуваних потреб при використанні у вказаних умовах. Підхарактеристики:

– *функціональна повнота (suitability, іноді — функціональна придатність)* — властивість ПС надавати належну множину функцій для вирішення специфікованих завдань і досягнення цілей користувача;

– *точність (accuracy)* — властивість ПС забезпечувати правильні й узгоджені результати або впливи з необхідним ступенем точності;

– *здатність до взаємодії (interoperability)* — властивість ПС взаємодіяти з однією чи більш специфікованими системами;

– *захищеність (security)* — здатність ПС забезпечувати захист інформації від несанкціонованого доступу осіб або систем до читання або модифікації та доступність інформації для осіб і систем, що мають право доступу;

– *узгодженість із нормативними документами (compliance)*.

2. Надійність (*Reliability*) — властивість ПС зберігати рівень функціонування протягом роботи в зазначених умовах. Підхарактеристики:

– *завершеність (maturity, іноді — безвідмовність)* — властивість ПС уникати відмов через дефекти, що містяться в ній;

– *відмовостійкість (fault tolerance, або стійкість до відмов)* — властивість ПС підтримувати встановлений рівень функціонування в умовах прояву дефектів, а також помилок у даних або порушень специфікованого інтерфейсу;

– *відновлювальність (recoverability)* — властивість ПС відновлювати функціонування на заданому рівні, а також відновлювати пошкоджені програми і дані.

3. Зручність використання (*Usability*) — властивість ПС бути зрозумілою, освоюваною, зручною і привабливою для користувачів під час використання в зазначених умовах. Підхарактеристики:

– *зрозумілість (understandability)* — властивість ПС, яка дає змогу користувачу зрозуміти, чи дійсно ПС може задовольнити його потреби та як вона може використовуватися для вирішення визначених завдань і які умови її використання;

– *вучуваність (learnability, іноді — освоюваність)* — властивість ПС, що забезпечує користувачів можливістю освоїти прийоми її застосування;

– *керованість (operability)* — властивість ПС, яка надає можливість користувачу керувати або контролювати її дії;

– *привабливість (attractiveness)* — властивість ПС, яка забезпечує її привабливість для користувача.

4. Ефективність (Efficiency) — властивість ПС забезпечувати раціональне використання виділених ресурсів під час роботи у встановлених умовах. Підхарактеристики:

– *реактивність (time behavior, або ефективність у часі)* — властивість ПС забезпечувати належний час відповіді (відгуку) та оброблення завдань, а також рівень пропускну здатності під час виконання функцій ПС у встановлених умовах застосування;

– *використовуваність ресурсів (resource utilization)* — властивість ПС використовувати належні ресурси в потрібні періоди часу під час виконання своїх функцій у встановлених умовах застосування.

5. Супроводжуваність (Maintainability) — властивість ПС забезпечувати можливість її ефективної модифікації. Модифікація може включати коригування, вдосконалення або адаптацію ПС до змін середовища, вимог або функціональних специфікацій:

– *аналізованість (analyzability)* — властивість ПС, яка дає змогу діагностувати її недоліки або причини відмов, а також ідентифікувати частини, які мають модифікуватися;

– *модернізованість (changeability, або змінюваність)* — властивість ПС, що дає змогу виконувати встановлені види модифікацій;

– *стабільність (stability)* — властивість ПС мінімізувати несподівані ефекти модифікацій;

– *тестованість (testability)* — властивість ПС сприяти перевірці модифікованого ПЗ.

6. Мобільність або переносимість (Portability) — властивість ПС мати можливість бути перенесеною з одного середовища в інше:

– *адаптованість (adaptability)* — властивість ПС адаптуватися для застосування в різних специфікованих середовищах без використання дій і засобів, відмінних від спеціально призначених для цих цілей;

– *зручність установлення (installability, іноді — налаштовуваність)* — властивість ПС, що зумовлює її здатність до інсталяції у специфікованому середовищі;

– *здатність до співіснування (co-existence, або сумісність)* — властивість ПС співіснувати з іншими незалежними ПС у загальному середовищі з розподіленням загальних ресурсів;

– *заміщувальна здатність (replaceability)* — властивість ПС бути використаною замість інших специфікованих ПС у середовищі їх застосування.

Для кожної підхарактеристики якості існує декілька різних метрик якості, а тому в моделі якості запровадимо атрибути. Кожний атрибут і буде вимірюватися за допомогою відповідної метрики. Таким чином, *загальна формула моделі якості* виглядає так [68]:

$$Q = \left\{ C_i \left\{ S_{ij} \left\{ A_{ijk} (M_{ijk}, W_{ijk}) \right\}_{k=1}^{K_{ij}} \right\}_{j=1}^{J_i} \right\}_{i=1}^I, \quad (1.1)$$

де Q — множина взаємозалежних характеристик (властивостей) якості; C_i — i -а характеристика якості; S_{ij} — j -а підхарактеристика i -ї характеристики якості; A_{ijk} — k -й атрибут j -ї підхарактеристики i -ї характеристики якості; M_{ijk} — метрика k -го атрибута j -ї підхарактеристики i -ї характеристики якості; W_{ijk} — коефіцієнт ваги (значущості) k -го атрибута j -ї підхарактеристики i -ї характеристики якості; i, j, k — індекси; I — загальна кількість характеристик якості в моделі якості ($2 \leq I \leq 6$); J_i — загальна кількість підхарактеристик i -ї характеристики якості; K_{ij} — загальна кількість атрибутів j -ї підхарактеристики i -ї характеристики якості.

Після визначення всіх елементів моделі можна починати випробування, у ході яких обчислюються фактичні значення атрибутів підхарактеристик якості. Ці значення порівнюються з обмеженнями, заданими у вимогах, як показано у працях [68; 69; 70].

Якщо отримані фактичні значення показників якості відповідають вимогам, то подальше узагальнене оцінювання можна провести за інтегральним показником якості. Інтегральний показник якості застосовують для вибору кращого ПЗ із декількох конкуруючих у цій галузі за умови узгодження конфліктуючих атрибутів якості. Але, оскільки в моделі є показники з різними метриками, такими, як неперервні числові, бальні, якісні та інші, необхідно заздалегідь узгодити та унормувати метрики, наприклад, через введення шкал для якісних та категорійних критеріїв і задання вагових множників [69].

Таким чином, ураховуючи формулу (1.1), інтегральний (узгальнений) рівень якості ПС U_q можна обчислити як середньозважений показник:

$$U_q = \sum_{i=1}^I W_i \sum_{j=1}^{J_i} W_{ij} \sum_{k=1}^{K_{ij}} W_{ijk} Q_{ijk},$$

де W_i — коефіцієнт ваги i -ї характеристики якості; W_{ij} — коефіцієнт ваги j -ї підхарактеристики i -ї характеристики якості; W_{ijk} — коефіцієнт ваги k -го атрибута j -ї підхарактеристики i -ї характеристики якості; Q_{ijk} — рівень якості k -го атрибута j -ї підхарактеристики i -ї характеристики якості, який обчислюється за формулою [44]

$$Q_{ijk} = \frac{P_{ijk}}{PB_{ijk}},$$

де P_{ijk} — обчислений рівень якості атрибута; PB_{ijk} — базовий рівень якості згідно з обраною метрикою (максимально можливе значення міри атрибута).

Наведемо приклад обчислення рівня якості *атрибута*, який вимірюється відповідно до метрики *completeness* (закінченість, завершеність) підхарактеристики *suitability* (функціональна повнота) характеристики *функціональність*. Це перший атрибут першої підхарактеристики першої характеристики якості моделі, тобто Q_{111} . Міра цього атрибута вимірюється відповідно до метрики X . Рівень якості цього атрибута будемо обчислювати за формулою

$$Q_{111} = X = 1 - A/B,$$

де A — кількість нереалізованих функцій; B — кількість функцій, які мають бути реалізовані відповідно до специфікацій та описів ПС. Тут $0 \leq X \leq 1$. Природно, що чим X ближче до 1, тим вищий показник міри [50].

1.11.2. Модель експлуатаційної якості

Вимірювання якості зазвичай потрібні на всіх рівнях моделі якості, оскільки задоволеності критеріїв внутрішньої якості не достатньо для забезпечення гарантії досягнення критеріїв зовнішньої якості, а, у свою чергу, задоволеності критеріїв зовнішньої якості зазвичай не достатньо для забезпечення гарантії задоволення потрібного рівня експлуатаційної якості, що показано на рис. 1.12.

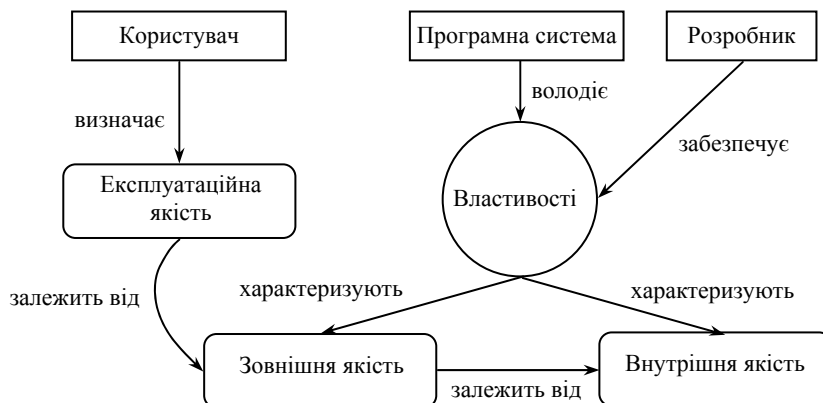


Рис. 1.12. Взаємозалежність понять під час оцінювання експлуатаційної якості

На відміну від моделі зовнішньої і внутрішньої якості для опису експлуатаційної якості використовують однорівневу модель, яка поділяє атрибути експлуатаційної якості за чотирьма характеристиками [52].

Характеристики експлуатаційної якості:

– *ефективність* (*іноді — результативність, effectiveness*) — ступінь, у якому користувач досягає заданих цілей щодо точності та повноти вирішення завдань у встановленому контексті використання ПС;

– *продуктивність* (*productivity*) — ступінь, у якому витрачаються ресурси на досягнення користувачем заданої ефективності у встановленому контексті використання ПС;

– *безпечність* (*safety*) — рівень ризику завдання матеріальних і моральних збитків, тобто шкоди здоров'ю людей, бізнесу, майну, навколишньому середовищу унаслідок використання ПС у встановленому контексті;

– *задоволеність* (*satisfaction*) — ступінь, у якому користувач задоволений ПС у певному контексті її використання.

1.11.3. Метрики в узагальненій моделі якості

Метрики в узагальненій моделі класифікуються за підхарактеристиками внутрішньої і зовнішньої якості, а також за характеристиками експлуатаційної якості, що описані у другій, третій і четвертій частинах стандарту ISO/IEC 9126 відповідно [50; 51; 52].

Опис метрик виконано за єдиною схемою із зазначенням такої інформації:

- 1) *ім'я метрики*. Відповідні метрики внутрішньої і зовнішньої якості мають однакові імена;
- 2) *призначення метрики*. Формулюється у вигляді питання, на яке дає відповідь метрика;
- 3) *метод застосування*. Містить правила отримання даних і схему їх використання в метриці;
- 4) *формула й елементи даних*. Указується формула обчислення і пояснюються елементи даних, які беруть у ній участь;
- 5) *інтерпретація вимірних даних*, діапазон значень і найбільше значення;
- 6) *тип шкали метрики* (див. п. 1.10.2);
- 7) *тип міри* (див. п. 1.10.3);
- 8) *вихідні дані*. Джерело даних, що використовується для вимірювання.
- 9) *процес ЖЦ*. Указується процес, у якому рекомендується застосування метрики для вимірювання.

1.12. Стандарти інженерії якості та сертифікація програмної системи

1.12.1. Стандарти у сфері якості

Стандарти розробляються міжнародними та національними органами стандартизації.

Найупливовіші з них:

- ISO — International Standard Organization;
- IEC — International Electrotechnical Commission;
- EOQ — European Organization for Quality;
- ANSI — American National Standards Institute;
- BSI — British Standards Institution;
- IEEE — Institute of Electrical and Electronics Engineering;
- NASA — National Aeronautics Space Administration;
- NIST — National Institute of Standards and Technology;
- SCC — Standards Council of Canada;
- ДСТУ — Державний комітет із стандартизації метрології та сертифікації України;
- ГОСТ — Государственный комитет РФ по стандартизации и метрологии.

Базовими стандартами з управління якістю продукту і систем управління якістю організацій-розробників є міжнародні стандарти серії ISO 9000 [71–78]. Стандарт ISO 9000-3 регламентує управління якістю під час розроблення, постачання та супроводження програмних засобів [74]. Сім'ю стандартів ISO 9000 більш детально розглянуто в підрозд. 2.1 і 2.4, зокрема у пп. 2.1.1 і 2.4.1.

Міжнародні стандарти в галузі інформаційних технологій розробляються *об'єднаним технічним комітетом* JTC1 (Joint technical Committee) *Information Technology*, заснованим ISO та IEC. Він включає 36 підкомітетів SC (*SubCommittee*), а за розроблення міжнародних стандартів з програмної інженерії (*Software Engineering — SE*) відповідає робочий підкомітет ISO/IEC SC7 *Software and System Engineering*. Для роботи над проектами стандартів у підкомітеті SC7 створено робочі групи (WG, Working Group) [17].

Базовий стандарт з управління якістю ПЗ:

ДСТУ ISO 9000-3 — Стандарти з управління якістю та забезпечення якості.

Частина 3. Настанови щодо застосування ДСТУ ISO 9001–95 під час розроблення, постачання та супроводження програмних засобів [77].

Чинні стандарти у сфері інженерії якості:

– ДСТУ ISO 9000–2001. Системи управління якістю. Основні положення та словник [75].

– ДСТУ ISO 9001–2001. Системи управління якістю. Вимоги [76].

– ДСТУ ISO 9004–2001. Системи управління якістю. Настанови щодо поліпшення діяльності [78].

– ISO/IEC 9126 (1991). *Information Technology (IT) — Software product evaluation — Quality characteristics and guidelines for their use* [79].

– ISO/IEC 12119 (1994) — IT — *Software packages — Quality requirements and testing* [80].

Вітчизняний стандарт ДСТУ ISO/IEC 12119 «Пакети програм — Тестування і вимоги до якості» [81] гармонізований зі стандартом [80]. Однак стандарт [81] перероблено і доповнено, оскільки він використовує вже модель якості ISO/IEC 9126-1 (2001 р., а не 1991 р. як [80]) для оцінювання якості й тестування програмного продукту.

- ISO/IEC 12207 (1995) — IT — *Software life cycle processes* [1].
(Процеси ЖЦ ПЗ).
- Група стандартів з оцінювання ПП — ISO/IEC 14598 [82–87]:
 - 14598-1 (1999) — IT — Оцінювання продукту — Частина 1.
Загальний огляд.
 - 14598-2 (2000) — SE — Оцінювання продукту — Частина 2.
Планування й керування.
 - 14598-3 (2000) — SE — Оцінювання продукту — Частина 3.
Процес для розробників.
 - 14598-4 (2000) — SE — Оцінювання продукту — Частина 4.
Процес для покупців.
 - 14598-5 (1998) — IT — Оцінювання продукту — Частина 5.
Процес для оцінювачів.
 - 14598-6 (2001) — SE — Оцінювання продукту — Частина 6.
Документація модулів оцінювання (специфікації спеціальних модулів, призначених для оцінювання ПЗ):
 - ISO/IEC 14764 (1999) — IT — *Software maintenance* [33].
(Супроводження ПЗ).
 - ISO/IEC 9126-1 (2001) *Software engineering — Product quality — Part 1: Quality model* [49].
 - ISO/IEC TR 9126-2 (2003) *Software engineering — Product quality — Part 2: External metrics* [50].
 - ISO/IEC TR 9126-3 (2003) *Software engineering — Product quality — Part 3: Internal metrics* [51].
 - ISO/IEC TR 9126-4 (2004) *Software engineering — Product quality — Part 4: Quality in use metrics* [52].
- IEEE Std. 610:12 (1990) — Глосарій термінології з програмної інженерії [88].
- IEEE Std. 12207.0 (1996) — Процеси життєвого циклу ПЗ [9].
- IEEE Std. 12207.1 (1997) — Дані життєвого циклу ПЗ [10].
- IEEE Std. 730 (1998) — Плани забезпечення гарантії якості ПЗ (Підходи до забезпечення якості відповідно до SQA) [53].
- IEEE Std. 830 (1999) — *Recommended Practice for Software Requirements Specifications* [12]. (Рекомендовані прийоми специфікації вимог до ПЗ).
- NASA std. 2201 (1993). Забезпечення гарантії ПЗ [55].
- NASA std. 8719.13A (1997). Безпека функціонування ПЗ [89].

ДСТУ 2844 (1994). Програмні засоби ЕОМ. Забезпечення якості. Терміни та визначення [45].

ДСТУ 2850 (1994). Програмні засоби ЕОМ. Показники та методи оцінювання якості [44]. (Стандарт гармонізовано з ISO/IEC 9126-1991).

ДСТУ 2851 (1994). Програмні засоби ЕОМ. Документування результатів випробовувань [48].

ДСТУ 2853 (1994). Програмні засоби ЕОМ. Підготовка та проведення випробувань [47].

ДСТУ 2462 (1994). Сертифікація. Основні поняття. Терміни та визначення [46].

ДСТУ 3918 (1999). Інформаційні технології. Процеси життєвого циклу ПЗ. [2]. (Гармонізовано з ISO/IEC 12207-1995).

ДСТУ 3919 (1999). Інформаційні технології. Основні напрямки оцінювання та відбору CASE-інструментів [30]. (Гармонізовано з ISO/IEC 14102).

ГОСТ 34.602-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы [90].

ГОСТ РФ 28195-89. Оценка качества программных средств. Общие положения [91].

1.12.2. Сертифікація програмних систем

Сертифікація — це процедура перевірки відповідності ПС вимогам замовника, розробника, стандартів з якості ПЗ і галузевих стандартів. Основні визначення і поняття в сфері сертифікації ПЗ викладено в стандарті ДСТУ 2462 (1994) — Сертифікація. Основні поняття. Терміни та визначення [46].

Наведемо визначення сертифікації згідно з ДСТУ 2462:

Сертифікація відповідності означає дії третьої сторони (*органу сертифікації та випробувальної лабораторії*), спрямовані на підтвердження відповідності ПЗ установленим вимогам стандартів та інших нормативних документів.

Сертифікація ПЗ виконується акредитованими незалежними *органами сертифікації*. Вона покликана підтвердити, що продукція, яка випускається, відповідає нормативним документам і вимогам замовника. Результат сертифікації оформляється у вигляді сертифіката.

Сертифікація може бути обов'язковою і добровільною. *Обов'язкова сертифікація* застосовується до ПЗ, а також до систем якості організації-розробника, задіяних під час розроблення *критичних ПС*, тобто ПС, пов'язаних із безпекою життєдіяльності. Необхідність обов'язкової сертифікації визначається замовником ПС, а в інших випадках рекомендується *добровільна сертифікація* [17].

У галузі сертифікації систем якості, а також продукції та послуг, функціонує державна система сертифікації продукції УкрСЕПРО. Вона об'єднує близько 150 акредитованих органів сертифікації та понад 800 виконавчих випробувальних (сертифікаційних) лабораторій.

Сертифікація систем якості організацій-розробників ПЗ регламентована в стандарті ДСТУ 3419 — Система сертифікації УкрСЕПРО — Сертифікація систем якості — Порядок проведення [92].

Останнім часом нова державна організація виконує основні дії зі стандартизації та сертифікації продуктів та послуг. Цією організацією став Укрметртестстандарт. Однією із базових організацій із сертифікації ПЗ є Софт-Рейтинг (*SoftRating*).

Лабораторія сертифікації (випробувальна лабораторія) проводить випробування (у тому числі сертифікаційні, атестаційні та ін.), а *орган сертифікації* займається висновками та прийняттям рішення і в кінцевому підсумку видачею *сертифікатів відповідності*. Сертифікат відповідності видається УКРСЕРТСОФТ (орган сертифікації, з яким співпрацює Софт-Рейтинг) на період 1, 2 або 3 роки залежно від обраної схеми сертифікації (рис. 1.13).

Обов'язки, діяльність та вимоги до компетенції випробувальних лабораторій задано в стандарті ISO/IEC 17025 — *General requirements for the competence of testing and calibration laboratories* [93].

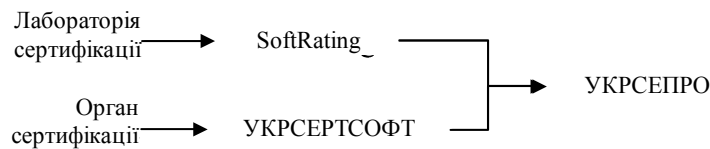


Рис. 1.13. Схема сертифікації програмних продуктів

1.12.3. Проект SQuaRE

Починаючи з 2005 р., проводяться роботи зі створення нової парадигми якості. У межах цих робіт запропоновано створити проєкт SQuaRE (*Software Quality Requirements and Evaluation*) шляхом об'єднання груп стандартів ISO/IEC 9126 (parts 1–4) [49–52] та ISO/IEC 14598 (parts 1–6) [82–87]. Серія SQuaRE (*SQuaRE Series of Standards*) складається з п'яти груп стандартів:

1. 2500n — *SQuaRE Quality Management Division* — управління якістю:
 - огляд, структура і довідник із SQuaRE [94];
 - планування й управління якістю продукту [95].
2. 2501n — *SQuaRE Quality Model Division* — модель якості:
 - модель якості [96];
 - указівки до застосування моделі якості;
 - модель якості даних [97].
3. 2502n — *SQuaRE Quality Measurement Division* — вимірювання якості:
 - еталонна модель вимірювання якості та вказівки до застосування [98];
 - базові та похідні елементи мір якості [99];
 - вимірювання внутрішньої якості [100];
 - вимірювання зовнішньої якості [101];
 - вимірювання якості у використанні [102];
 - документування модулів оцінювання [103].
4. 2503n — *SQuaRE Quality Requirements Division* — вимоги до якості:
 - вимоги до якості ПЗ [104];
 - указівки до визначення якості.
5. 2504n — *SQuaRE Quality Evaluation Division* — оцінювання якості програмного продукту:
 - модель і керівництво з процесів оцінювання [105];
 - модулі оцінювання [106];
 - процес оцінювання для розробників [107];
 - процес оцінювання для покупців (споживачів) [108];
 - процес для оцінювачів [109].

Подальші розділи від 2505n до 2509n залишені для розвитку проекту SQuaRE. Унаслідок цього появився стандарт з вимог до якості комерційних ПП та керівництво з тестування: ISO/IEC 25051 — Requirements for Quality of Commercial Off-The-Shelf (COTS) Software Product and Instructions for Testing [110], а також стандарт із загальних шаблонів використання звітів з тестування: ISO/IEC 25062 — *Common Industry Format (CIF) for Usability Test Reports* [111]. Структуру серії стандартів SQuaRE показано на рис. 1.14.

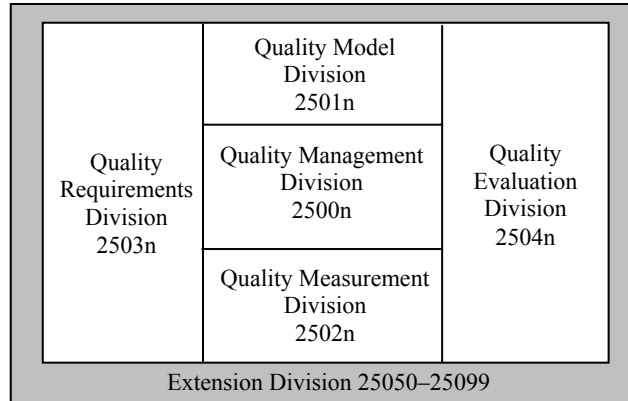


Рис. 1.14. Організація серії міжнародних стандартів SQuaRE

Загальну структуру моделі якості продукту (комп'ютерної системи або ПЗ) відповідно до стандарту ISO/IEC 25010 SQuaRE — *Quality Model* [96] показано на рис. 1.15.

1. Функціональна придатність (*Functional Suitability*) — ступінь, у якому продукт (виріб) або система забезпечують виконання своїх функцій відповідно до встановлених і очікуваних потреб під час використання у вказаних умовах. Підхарактеристики:

– *функціональна повнота, завершеність (functional completeness)* — ступінь (міра), у якому належна множина функцій системи охоплює всі вказані (специфіковані) завдання і цілі користувача;

– *функціональна правильність, точність (functional correctness)* — ступінь, у якому продукт (виріб) або система забезпечує правильні результати з необхідним ступенем точності;

– *функціональна відповідність, доречність (functional appropriateness)* — ступінь, у якому належні функції полегшують або спрощують виконання вказаних (специфікованих) завдань та досягнення цілей користувача.

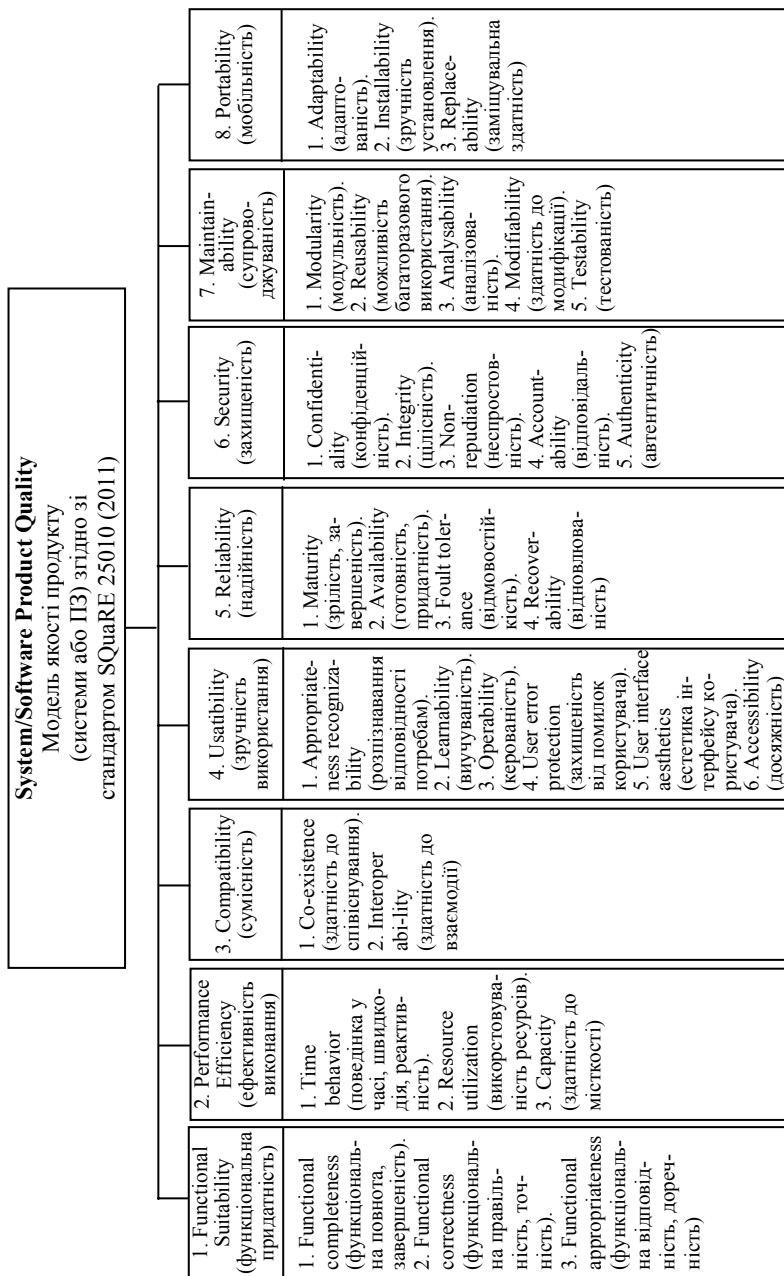


Рис. 1.15. Модель якості продукту (системи або ПЗ) згідно зі стандартом SQuaRE

2. Ефективність виконання (*Performance Efficiency*) — ефективність виконання відповідно до кількості виділених ресурсів під час роботи у встановлених умовах. Підхарактеристики:

– *поведінка у часі (ефективність у часі), швидкодія, реактивність (time behavior)* — ступінь, у якому продукт або система відповідають вимогам щодо забезпечення належного часу відгуку та оброблення завдань, а також рівня пропускну здатності під час виконання своїх функцій;

– *використання ресурсів (resource utilization)* — ступінь, у якому кількість та категорії ресурсів, що використовує продукт або система під час виконання своїх функцій, відповідають установленим вимогам;

– *здатність до місткості (capacity)* — ступінь, у якому максимальні граничні значення параметрів продукту або системи відповідають установленим вимогам.

3. Сумісність (*Compatibility*) — ступінь, у якому продукт, система або компонент можуть обмінюватися інформацією з іншими продуктами, системами або компонентами і/або виконувати належні функції, розділяючи апаратні ресурси та операційне середовище. Підхарактеристики:

– *здатність до співіснування (co-existence)* — ступінь, у якому продукт (система) може ефективно виконувати належні йому функції, розділяючи загальне середовище та ресурси з іншими продуктами без шкідливого впливу на будь-який інший продукт;

– *здатність до взаємодії (interoperability)* — ступінь, у якому дві або більше систем, продуктів або компонентів можуть обмінюватися інформацією та використовувати інформацію, (прийнята або переслана).

4. Зручність використання (придатність до використання) (*Usability*) — ступінь, у якому система або продукт можуть бути використані встановленими користувачами (споживачами) для досягнення належних цілей щодо ефективності, продуктивності та задоволеності під час використання в зазначених умовах. Підхарактеристики:

– *розпізнавання відповідності потребам (appropriateness recogni-zability)* — ступінь, у якому користувачі можуть розпізнати, чи дійсно продукт або система можуть задовольнити їх потреби;

– *вучуваність (learnability, іноді — освоюваність)* — ступінь, у якому продукт або система можуть бути використані встановленими

користувачами для досягнення належних цілей щодо вивчення, маючи на меті використання продукту або системи з ефективністю, продуктивністю, свободою від ризику та задоволеністю під час роботи в зазначених умовах;

– *керованість, придатність до роботи (operability)* — ступінь, у якому продукт або система мають атрибути, що полегшують користувачу використання та керування продуктом або системою;

– *захищеність від помилок користувача (user error protection)* — ступінь, у якому система захищає користувачів від допущення ними помилок;

– *естетика інтерфейсу користувача (user interface aesthetics)* — ступінь, у якому користувацький інтерфейс забезпечує взаємодію для користувача привабливою і такою, що викликає задоволення;

– *досяжність (accessibility)* — ступінь, у якому продукт або система можуть бути використані людьми із широким діапазоном характеристик і здібностей для досягнення встановлених цілей у зазначеному контексті використання продукту або системи.

5. Надійність (Reliability) — ступінь, у якому система, продукт або компонент виконують указані функції під час роботи у зазначених умовах протягом установленого періоду часу. Підхарактеристики:

– *зрілість або завершеність (maturity, іноді — безвідмовність)* — ступінь відповідності системи потребам у надійності за нормально-го функціонування у вказаних умовах;

– *готовність або придатність (availability)* — ступінь, у якому система, продукт або компонент є працездатними та досяжними, у разі потреби, для використання;

– *відмовостійкість або стійкість до відмов (fault tolerance)* — ступінь, у якому система, продукт або компонент працюють як встановлено (як слід), незважаючи на наявність і/або прояв дефектів у ПЗ або апаратних платформах чи засобах;

– *відновлюваність (recoverability)* — ступінь, у якому система чи продукт у випадку переривання або відмови може безпосередньо відновити пошкоджені дані та поновити бажаний стан функціонування системи.

6. Захищеність, безпека (Security) — ступінь, у якому продукт чи система захищають інформацію та дані таким чином, щоб будь-які особи або інші продукти чи системи мали рівень доступу до даних, що відповідає їх типу та рівню авторизації. Підхарактеристики:

– *конфіденційність (confidentiality)* — ступінь, у якому продукт чи система гарантують доступність даних лише для осіб або інших продуктів чи систем, які уповноважені мати доступ;

– *цілісність (integrity)* — ступінь, у якому система, продукт чи компонент запобігають несанкціонованому доступу або модифікації комп'ютерних програм і/або даних;

– *неспростовність, незворотність (неанульованість, некасованість) (non-repudiation)* — ступінь, у якому дії або події, щодо яких можна довести, що вони відбувалися в системі, надалі не можуть бути спростовані (анульовані, скасовані) унаслідок інших дій;

– *відповідальність, здатність до обліку (accountability)* — ступінь, у якому дії об'єкта в роботі системи можуть бути відслідковані та унікально прив'язані до цього об'єкта;

– *автентичність, достовірність (authenticity)* — ступінь, у якому ідентичність об'єкта (предмета) або ресурсу може бути доведена належним чином (як потрібно).

7. Супроводжуваність (Maintainability) — ступінь ефективності та продуктивності, у якому продукт чи система можуть бути змінені (модифіковані) призначеними особами, що виконують супроводження.

Модульність (modularity) — ступінь, у якому система або комп'ютерна програма складена (скомпонована) з дискретних компонентів (модулів) так, що зміна одного компонента має мінімальний вплив на інші компоненти.

Можливість багаторазового використання (reusability) — ступінь, у якому робочі продукти системи (документи вимог до системи, вихідний код модулів системи тощо) можуть бути використані у більш ніж одній системі, чи для формування (побудови) інших робочих продуктів.

Аналізованість (analysability) — ступінь ефективності та продуктивності, у якому можливо оцінити вплив на продукт чи систему призначеної зміни однієї чи більше її частин, або можна діагностувати її недоліки чи причини відмов, а також ідентифікувати частини, які повинні модифікуватися.

Здатність до модифікації (modifiability) — ступінь, у якому продукт або система можуть бути ефективно та продуктивно змінені (модифіковані) без унесення дефектів і погіршення існуючої якості продукту.

Тестованість (testability) — ступені ефективності та продуктивності, у яких для системи, програми чи компонента можуть бути встановлені критерії тестування (перевірки), і тести можуть бути виконані для визначення, чи були досягнуті ці критерії.

8. Мобільність або переносимість (Portability) — ступені ефективності та продуктивності, у яких система, продукт чи компонент можуть бути переміщені з одного апаратного, програмного, або іншого операційного чи користувацького середовища до іншого.

Адаптованість (adaptability) — ступінь, у якому продукт або система можуть бути фактично та ефективно адаптовані (приспосовані) для застосування в різних або еволюціонуючих апаратних, програмних, або інших операційних чи користувацьких середовищах.

Зручність встановлення, здатність до встановлення (Installability) — ступені ефективності та продуктивності, у яких продукт або система можуть бути успішно інстальовані (встановлені) і/або деінстальовані у вказаному (специфікованому) середовищі.

Заміщувальна здатність (replaceability) — ступінь, у якому продукт може бути замінений іншим специфікованим програмним продуктом для однієї і тієї самої мети в одному і тому самому середовищі застосування.

Крім загальної моделі якості продукту, у стандарт вводиться розгляд моделі якості у використанні, яка у SQuaRE стає системною якістю, але вона, крім того, відображає результати взаємодії з комп'ютерною системою, у першу чергу, замовників і користувачів, та їх погляд на якість продукту. Якість у використанні також характеризує вплив, який продукт або система чинить на її користувачів. Модель якості у використанні показано на рис. 1.16.

1. Ефективність або результативність (Effectiveness) — точність і завершеність, з якими користувачі досягають указаних (специфікованих) цілей під час роботи у встановлених умовах.

2. Продуктивність (Efficiency) — ресурси, що витрачені для досягнення точності та завершеності, з якими користувачі досягають указаних (специфікованих) цілей під час роботи у встановлених умовах.

3. Задоволеність (Satisfaction) — ступінь, у якому продукт або система задовольняє потреби користувачів у разі використання системи у встановлених умовах. Підхарактеристики:

– *повноцінність (usefulness)* — ступінь, у якому користувач задоволений сприйнятим досягненням своїх прагматичних цілей,

включаючи результати та наслідки використання системи чи продукту у встановлених умовах;

– *довіра (trust)* — ступінь, у якому користувач, замовник або інший співвласник продукту чи системи (або інша зацікавлена сторона) мають довіру, що цей продукт чи система будуть поводитися так, як визначено під час роботи у встановлених умовах;

– *задоволення (pleasure)* — ступінь, у якому користувач задоволений реалізацією своїх потреб, використовуючи продукт або систему;

– *комфорт (comfort)* — ступінь, у якому користувач задоволений своїм фізичним комфортом.

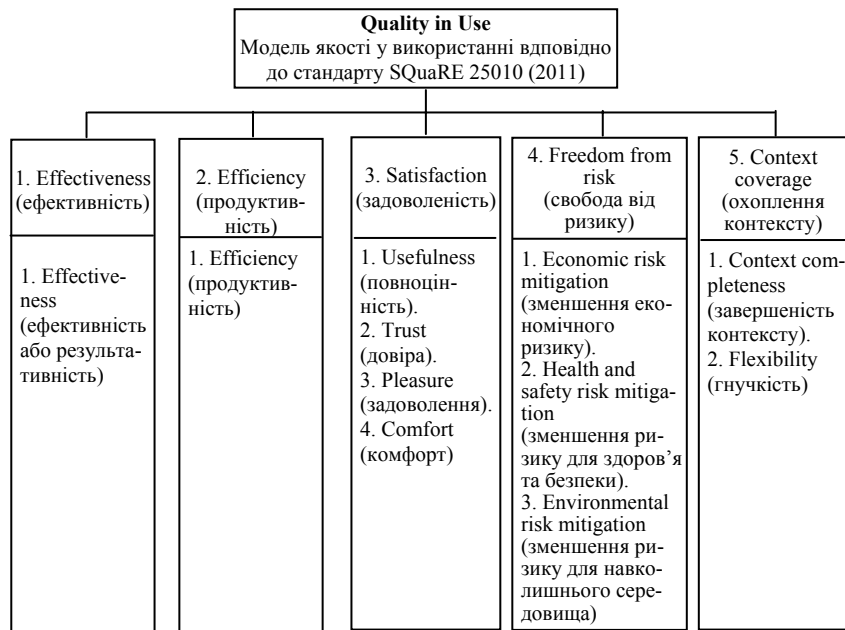


Рис. 1.16. Модель якості у використанні згідно зі стандартом SQuaRE

4. Свобода від ризику (*Freedom from risk*) — ступінь, у якому продукт або система зменшують (пом'якшують) потенційний ризик економічному стану (статусу), життю і здоров'ю людей та навколишньому середовищу під час використання продукту або системи у встановлених умовах. Підхарактеристики:

– зменшення економічного ризику (*economic risk mitigation*) — ступінь, у якому продукт або система зменшують (пом'якшують) потенційний ризик фінансового стану, ефективності дій та операцій, комерційній власності, репутації або іншим ресурсам у призначеному контексті використання продукту чи системи;

– зменшення ризику для здоров'я та безпеки (*health and safety risk mitigation*) — ступінь, у якому продукт або система зменшують (пом'якшують) потенційний ризик для людей у призначеному контексті використання продукту чи системи;

– зменшення ризику для навколишнього середовища (*environmental risk mitigation*) — ступінь, у якому продукт або система зменшують (пом'якшують) потенційний ризик власності або навколишньому середовищу у призначеному контексті використання продукту чи системи.

5. Охоплення контексту (*Context coverage*) — ступінь, у якому продукт або система можуть бути використані з ефективністю, продуктивністю, свободою від ризику і задоволеністю у вказаних контекстах використання та у контекстах за межами цих початково явно ідентифікованих контекстів. Підхарактеристики:

– *завершеність контексту (*context completeness*)* — ступінь, у якому продукт або система можуть бути використані з ефективністю, продуктивністю, свободою від ризику і задоволеністю у всіх специфікованих (вказаних) контекстах їх використання;

– *гнучкість (*flexibility*)* — ступінь, у якому продукт або система можуть бути використані з ефективністю, продуктивністю, свободою від ризику і задоволеністю у контекстах за межами початково визначеними у вимогах до системи.

Співвласники цільової комп'ютерної системи, що розробляється (замовники, власники, розробники, група якості, оцінювачі, персонал супроводження, користувачі), зацікавлені у досягненні високого рівня якості кінцевої системи, зокрема її ПЗ.

Це досягається за допомогою побудови моделі якості продукту та моделі якості у використанні з подальшим оцінюванням рівня якості атрибутів характеристик та підхарактеристик якості цих моделей.

Модель якості продукту відображає рівень якості цільової комп'ютерної системи та її ПЗ, а модель якості у використанні, крім того, характеризує вплив цієї системи та її ПЗ на інформацій-

ну систему в цілому, на використовуване середовище та на первинних, вторинних і непрямих користувачів системи, а також на інші зацікавлені сторони.

1.13. Тестування програм і систем

Концепція проектування ПС містить базові визначення тестування, відмов, дефектів і помилок, а також стратегії, пов'язані з етапами ЖЦ, зокрема з процесом тестування та його елементами.

Тестування — це спосіб семантичної перевірки (налагодження) програми, що складається з опрацювання програмою послідовності різноманітних контрольних тестових наборів даних (ТНД) з наперед відомими результатами.

Проведення випробувань та їх документування регламентовано стандартами [47; 48]. Широке коло питань, пов'язаних із тестуванням, розглянуто у працях С. Канера, В. Ліпаєва, Г. Майерса [112; 113; 114]. Відповідно до стандарту [47] випробування поділяються на *попередні, приймальні, інсталяційні та експлуатаційні*. Документування випробувань, окрім стандарту [48], розглянуто також у стандарті IEEE Std. 829 — *IEEE Standard for Software Test Documentation* [115].

Тестові набори даних підбирають так, щоб вони охоплювали найбільшу кількість типових ситуацій, які виконуються у процесі роботи програми, тобто, щоб програма використовувала в ідеалі всі допустимі шляхи графу передавання керування.

Тестування охоплює такі види робіт:

- розроблення ПП на відповідних етапах ЖЦ та верифікацію результатів на кожному етапі;
- створення плану тестування і підготовка ТНД для перевірки окремих елементів програми і програми в цілому;
- керування проходженням тестів і аналіз результатів тестування.

Як правило, тестування починається з останньої фази проектування і містить розроблення планів і тестів перевірки правильності функціонування вихідного коду. Тестування становить від 30 до 50 % трудомісткості робіт зі створення коду ПС. Історично першим різновидом тестування було налагодження програм.

Налагодження — це перевірка ПП (об'єкта) на наявність у ньому помилок і наступне їх усунення.

Під час сертифікації ПС кінцевою метою тестування є підтвердження правильності функціонування компонентів та надійності ПС у цілому [46]. У разі успішного проходження *сертифікаційних випробувань* організація-розробник отримує сертифікат відповідності на розроблену ПС.

1.13.1. Методи тестування

Функціональне тестування, як правило, проводять за планом, розробленим на етапі проектування ПС. Головна мета — виявити невідповідності між фактичною поведінкою реалізованих функцій і очікуваною їх поведінкою відповідно до специфікацій на ПЗ. Тестування ПЗ згідно із зовнішніми специфікаціями проводять відповідно до вимог, сформульованих на етапі аналізу предметної галузі.

Тести відображають структури даних, алгоритми компонентів та інтерфейси між окремими компонентами системи.

Методи функціонального тестування поділяються на *статичні* та *динамічні*.

Статичні методи використовують під час проведення інспекцій вихідного коду й аналізу специфікацій компонентів без виконання програм на комп'ютерній платформі.

Динамічні методи використовують у процесі виконання програми на комп'ютері. Вони ґрунтуються на побудові графу, що пов'язує причини помилок з очікуваним реагуванням на ці помилки. У процесі динамічного тестування нагромаджується інформація про помилки, яку надалі використовують для оцінювання надійності програми [114].

1.13.2. Статичні методи тестування

До *статичних методів* належать методи аналізу послідовностей операторів і аналізу перетворень типів даних. Техніка аналізу полягає в методичному перегляді та аналізі структури програм, а також у доведенні правильності програм формальними методами.

Статичний аналіз спрямований на аналіз документів, створених на всіх етапах ЖЦ, і полягає в *інспекції вихідного коду програми* та *наскрізному контролі*. *Інспекція вихідного коду* складається зі спільного розгляду документів незалежними експертами за участю розробників.

Спочатку перевіряється повнота, цілісність, однозначність і сумісність документів на ПС порівняно з вихідними вимогами. На етапі реалізації ПС під інспекцією розуміють аналіз текстів програм щодо відповідності вимогам стандартів і керівних нормативних документів технології програмування.

Під час проведення інспекції на проблему намагаються подивитися «збоку» і її проаналізувати [112; 114]. Проведення *формальних інспекцій* і *наскрізного контролю* (або *перегляду*) регламентується стандартом *NASA Std. 2202 — NASA Software Formal Inspections Standard* [116], а також *NASA A302 — NASA Software Formal Inspections Guidebook* [117].

Під час *наскрізного контролю* виконують безпосередній перегляд коду програми (ручна імітація), що дозволяє виявити помилки в логіці. У свою чергу статичні методи поділяють на такі методи:

- *метод простого структурного аналізу* (Дейкстра і Хоар) — орієнтований на аналіз структури програми (поток керування та даних). Для проведення аналізу структуру програми подають у формі графової моделі, де кожна вершина — оператор, а дуга — передавання керування. Цей підхід дозволяє виявити логічні помилки;

- *метод доведення правильності програм* (Флойд і Наур) — метод надійний, однак надто трудомісткий і потребує колосальних ресурсів;

- *метод аналізу дерева відмов* — вибір *ситуації відмови* в окремому компоненті системи та подальше відстеження подієвих ланцюжків, які можуть призвести до цієї ситуації;

- *метод аналізу інтерфейсів* та ін.

1.13.3. Динамічні методи тестування

Під *динамічним тестуванням* розуміють перевірку ПС за допомогою виконання її на комп'ютерній платформі.

Динамічне тестування ґрунтується на *систематичних, статистичних та імітаційних методах*.

Систематичні методи:

- кожна програма розглядається як «чорний ящик» (використовується інформація про виконуване завдання);

- програми розглядаються як «скляний ящик», або «білий ящик» (використовується інформація про структуру програми).

Найповніше систематичне модульне тестування висвітлено у стандарті *ANSI/IEEE Std. 1008 — IEEE Standard for software Unit Testing* [118].

Тестування за принципом «чорного ящика» інакше називають тестуванням через керування входом/виходом (вхідні дані/вихідні дані). Мета тестування — дослідити обставини, за яких поведінка програми не відповідає її специфікаціям, при цьому кількість виявлених помилок і є критерієм оцінювання. У свою чергу методи «чорного ящика» поділяють на такі [112; 114]:

- метод еквівалентного розбиття;
- метод аналізу граничних значень;
- метод функціональних діаграм;
- інші методи.

Еквівалентне розбиття — це розбиття вхідної області даних програми на скінченну кількість класів еквівалентності з метою покриття різних маршрутів виконання в програмах і їх логічного оброблення.

Тестовий набір даних розробляють за *методом еквівалентних розбиттів* за два етапи: 1) виділення класів еквівалентності; 2) побудова тестів. Правильні класи еквівалентності містять правильні вхідні дані програми і акцентують увагу на правильних умовах, а неправильні класи — неправильні дані (акцентують увагу на неправильних умовах) [112; 114].

Наприклад, для програми, яка обробляє інформацію про політ повітряного корабля з метою контролю пілотування, необхідно перевірити, чи виконувалась умова додержання висоти польоту згідно з предикатами для барометричної висоти (H_6): $800 \leq H_6 \leq 1200$ (у метрах). У цьому разі отримаємо два неправильні класи еквівалентності: $H_6 \leq 800$ (I) і $H_6 \geq 1200$ (III) та один правильний клас еквівалентності: $800 \leq H_6 \leq 1200$ (II), як показано на рис. 1.17.

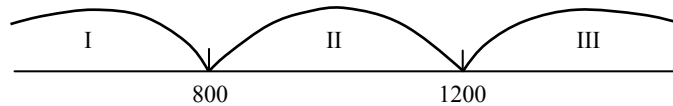


Рис. 1.17. Правильний та неправильні класи еквівалентності

Метод еквівалентних розбиттів враховує синтаксичні та семантичні правила побудови програм, а також спосіб задання даних. Тому він є одним з найкорисніших з методів «чорного ящика».

Метод аналізу граничних значень ґрунтується на обранні областей тестових значень поблизу меж класів еквівалентності. Цей метод є продовженням методу *еквівалентних розбиттів*.

Для прикладу з попереднього методу отримуємо нові граничні області (класи IV і V), як показано на рис. 1.18.

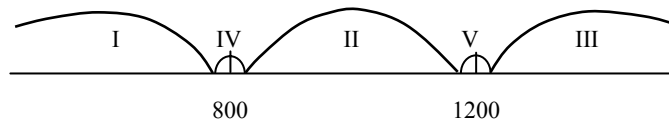


Рис. 1.18. Класи еквівалентності та граничні області

Підготовка ТНД для тестування модулів ПС згідно з *методом аналізу граничних значень* зводиться до включення граничних умов, для чого в тести вводять дані, розміщені безпосередньо вище чи нижче від границь вхідних чи вихідних класів еквівалентності.

Метод функціональних діаграм складається переважно з набору алгоритмічних правил для вибору значень операндів в операторах програми і визначення очікуваних вихідних значень для кожного застосованого тесту (стовпці таблиці рішень [114]). Метод корисний для визначення побічних наслідків, що призводять до помилок, але є громіздким у реалізації.

Методи «білого ящика» дозволяють дослідити внутрішню структуру програми, причому виявити всі помилки можливо лише в разі повного тестування всіх маршрутів її графу передавання керування.

Існують такі критерії та методи:

- критерій покриття операторів — ТНД повинні забезпечити проходження кожного оператора не менше ніж один раз;
- критерій покриття рішень і/або переходів — ТНД має забезпечити проходження кожної гілки програми хоча б один раз;
- комбіновані методи (перший критерій з варіантами другого);
- метод тестування умов.

Методи тестування за принципом «білого ящика» орієнтовані на перевірку всіх шляхів програми, а методи «чорного ящика» оперують методами функціонального тестування ПЗ [114].

Окремі тести поєднують у тестові сценарії, що орієнтовані на перевірку рівня якості вирішення всіх функціональних завдань ПС.

1.14. Спосіб оцінювання рівня якості програмних систем під час їх сертифікації

1. Згідно з ДСТУ 2462–94 [46] та ДСТУ 2844–94 [45] під сертифікацією ПС розуміють дії незалежного органу з перевірки відповідності фактичних властивостей ПС тим, що заявлені в документації, або встановлені в нормативних документах. Натепер в інженерії ПЗ для опису властивостей ПС використовують в основному корпоративні системи класифікації. У них заявлені властивості або вимоги до ПС, які перевіряються у процесі сертифікації, задаються на різному рівні узагальнення і без використання загальноприйнятої системи класифікації, що зумовлює неоднозначність трактувань цих вимог та ускладнює процес оцінювання, оскільки у вимогах можуть бути відсутні, пов'язані з властивостями, атрибути та метрики їх вимірювання. Крім того, це створює незручності, зумовлені неоднозначним тлумаченням належності властивостей до класів, і в кінцевому підсумку може призвести до невизнання отриманих результатів дослідження рівня якості ПС або неможливості порівняння ПС одного класу.

Вирішити ці проблеми можна введенням у використання загальноприйнятої системи класифікації властивостей ПС, яка базується на узагальненій моделі якості, запропонованій у стандарті ISO/IEC 9126 (parts 1–4) [49–52]. Процедури оцінювання властивостей ПС логічно будувати відповідно до рекомендацій стандартів ISO/IEC 14498 (parts 1–6) [82–87], які узгоджені з ISO/IEC 9126.

Вимоги або заявлені властивості, відповідність яким перевіряється, формулюються замовником і розробником ПС і викладені в документації на ПС. Сертифікація ПС ґрунтується на результатах випробувань (ДСТУ 2851–94, ДСТУ 2853–94 [48; 47]) і полягає в оцінюванні відповідності виявлених властивостей розробленої системи вимогам до неї або заявленим у документації властивостям.

2. Організації, які проводять сертифікацію ПС у системі УкрСЕПРО, використовують технологію, яка базується на стандарті ДСТУ ISO/IEC 12119:2003 [81]. Цей стандарт містить настанови з формалізованого подання заявлених властивостей ПС та процедури тестування. Однак набір класів, відповідно до якого проводиться класифікація властивостей ПС, обмежується лише характеристиками ПС, які визначені в ISO/IEC 9126–1 [49], без урахування метрик.

Використання однорівневої класифікації в умовах наявності лише досить загальних нечітких ознак належності властивості до класу може приводити до перетину класів та інших неоднозначностей.

Пропонований спосіб розроблений з використанням наведених у п. 1 принципів положень і дає змогу подолати вказані проблеми. Першим кроком технології оцінювання є класифікація заявлених властивостей ПС, які оцінюються відповідно до рекомендацій ISO/IEC 9126.

Беручи до уваги пропозиції ISO/IEC 9126–1 [49], визначимо трирівневу ієрархічну систему класів властивостей ПС. На першому рівні визначимо клас характеристик C^C , на другому — підхарактеристик C^S , на третьому — атрибутів і/або метрик $C^{a.m}$. Ієрархічність виражається так: $C_{ij}^S \in C_i^C$, $i = 1, 2, \dots, 6$, $j = 1, 2, \dots, J_i$; $C_{ijk}^{a.m} \in C_{ij}^S$, $k \in K_{ij}$.

Трирівнева ітераційна процедура класифікації враховує характеристики (перший рівень), підхарактеристики (другий рівень) та атрибути з їх метриками (третій рівень). Процедура класифікує властивість у напрямку зверху вниз, якщо цю властивість описано в загальному вигляді, і знизу вгору, якщо задано конкретну міру властивості.

У першому випадку згідно з описом визначається характеристика, підхарактеристика та атрибут, і якщо для вимірювання цієї властивості підходить яка-небудь метрика даної підхарактеристики, то класифікація завершена, а якщо ні, то потрібно визначити нову підхарактеристику і так далі. Для властивостей, які не мають відповідності у класах за назвою, шукають аналог за змістом. У другому випадку для заданої міри підбирається відповідна метрика, яка і визначає для цієї властивості підхарактеристику та характеристику.

Для виконання робіт з оцінювання рівня якості ПС заявники подають документ на заявлені властивості ПС у формі, передбаченій у підрозд. 1.15.

3. На другому кроці технологія передбачає побудову сертифікаційної моделі якості. Сукупність обраних у п. 2 під час класифікації характеристик, підхарактеристик та атрибутів з їх метриками складе сертифікаційну модель якості.

$$Q_{Ce} = \left\{ C_i \left\{ S_{ij} \left\{ A_{ijk} \left(M_{ijk}, B_{Ce_{ijk}}, W_{ijk} \right), A_{ijk} \in P_{ijk} \right\}_{k=1}^{K_{ij}^{Ce}} \right\}_{j=1}^{J_i^{Ce}} \right\}_{i=1}^{I^{Ce}}, \quad (1.2)$$

де C_i — i -а характеристика; S_{ij} — j -а підхарактеристика i -ї характеристики; A_{ijk} — k -й атрибут j -ї підхарактеристики i -ї характеристики якості ($2 \leq I^{Ce} \leq 6$).

У цій моделі відповідність фактичних показників атрибутів вимогам залежить від обмежень сертифікації $B_{Ce_{ijk}}$, що задані у нормативних документах. Якщо галузевих нормативних документів немає, то вибір атрибутів варто залишити на розсуд експертів випробувальної лабораторії, які повинні керуватися загальними стандартами якості, а також вимогами ТЗ на розроблення ПС.

Для того щоб модель (1.2) була завершеною, необхідно обрати метрики M_{ijk} вимірювання атрибутів підхарактеристик та вагові коефіцієнти W_{ijk} .

Для кожного атрибута сертифікаційної моделі (1.2) має бути відповідна вимога або властивість. Додаткові характеристики можуть бути введені в модель за погодженням із замовником сертифікації. При цьому метрики M_{ijk} вимірювання атрибутів обирають з урахуванням рекомендацій стандарту ISO/IEC 9126-2 [50], а також особливостей сфери застосування ПС, оскільки обмеження $B_{Ce_{ijk}}$ задані в галузевих стандартах і нормативних документах.

Після цього залежно від специфіки предметної галузі атрибути поділяють на критичні і другорядні, призначають вагові коефіцієнти W_{ijk} . Атрибути A_{ijk} повинні належати множині допустимих атрибутів P_{ijk} (критерій — задоволення атрибутів A_{ijk} обмеженням сертифікації $B_{Ce_{ijk}}$).

Сертифікація полягає у перевірці належності значень атрибутів моделі допустимій множині [формула (1.2)]. Ця перевірка здійснюється шляхом сертифікаційних випробувань. Запропонований підхід (проведення класифікації заявлених властивостей на множині уніфікованих характеристик якості загальних стандартів) можна застосовувати для будь-яких класів ПС.

4. Після визначення всіх елементів моделі (1.2) переходять до сертифікаційних випробувань, у ході яких обчислюються фактичні значення атрибутів підхарактеристик якості, після чого ці значення порівнюються з нормативними обмеженнями.

Якщо отримані фактичні значення показників якості відповідають нормативним вимогам, то подальшу оцінку можна виконати, використовуючи інтегральний показник якості, у якому вага кожного критичного показника має бути більшою від суми ваг другорядних. Інтегральний показник якості можна застосувати для вибору кращої ПС із декількох конкуруючих за умови врахування значущості критичних характеристик, вимоги до яких потрібно виконувати обов'язково. Оскільки в моделі є показники з різними метриками (неперервні числові, бальні, якісні та ін.), необхідно попередньо узгодити та нормувати метрики. Це можна зробити, наприклад, шляхом уведення шкал для якісних та категорійних критеріїв і заданням вагових множників.

Диференціальний метод не може забезпечити отримання інтегральної оцінки якості ПС, а тому обирають комплексний метод оцінювання рівня якості, що ґрунтується на використанні узагальненого показника якості із залученням ранжування для оцінювання ступеня задоволення вимогам. Метод полягає в тому, що для оцінювання рівня якості ПС використовують узагальнений показник якості U , що обчислюється як середній зважений арифметичний показник:

$$U = \sum_{i=1}^N Q_i W_i, \quad (1.3)$$

де Q_i — відносний показник якості, що визначається зі співвідношення $Q_i = \frac{P_i}{P_{ib}}$; P_i — рівень якості i -го елемента показника якості

($i = \overline{1, N}$); P_{ib} — базове значення i -го елемента; W_i — ваговий коефіцієнт (параметр значущості) i -го елемента показника якості.

Для використання цього підходу пропонуємо задавати параметр значущості кожного критичного показника більшим ніж сума вагових коефіцієнтів усіх другорядних показників. Базове значення P_{ib} пропонується брати рівним одиниці для всіх показників, а рівень якості P_i можна остаточно обчислити відповідно до принципу кла-

сифікаційної метрики (тобто P_i перебуває або не перебуває у допустимих межах).

5. Для розрахунку трирівневого узагальненого інтегрального показника якості U_G врахуємо вираз (1.3), а також формулу із ДСТУ 2850–94 [44]. За багаторівневої ієрархічної структури номенклатури показників якості розрахунок рівня якості виконується знизу вгору:

$$U_G = \sum_{i=1}^I W_i \sum_{j=1}^{J_i} W_{ij} \sum_{k=1}^{K_{ij}} W_{ijk} Q_{ijk},$$

де I — кількість характеристик; J_i — кількість підхарактеристик i -ї характеристики; K_{ij} — кількість атрибутів j -ї підхарактеристики i -ї характеристики якості.

Для розрахунку рівня якості ПС пропонується обрати такі метрики: об'єктивні (оскільки вони рахункові) і номінальні (класифікаційні). Номінальні метрики обрано тому, що для оцінювання критичних показників важлива наявність такої властивості (1 — очікувана функція виконується), чи її відсутність (0 — очікувана функція не виконується). У цьому випадку важливо фіксувати повне виконання, а тому для остаточної оцінки недоцільно застосовувати абсолютні шкали. Будемо використовувати зовнішні метрики (ISO/IEC 9126–2 [50]), оскільки за допомогою тестування можна отримати фактичні показники атрибутів якості ПС на стадії сертифікаційних випробувань. Для розрахунку рівня якості ПС перевагу пропонується віддати номінальним шкалам оцінювання мір атрибутів з використанням вагових коефіцієнтів для ранжування. Однак для попередньої оцінки атрибута можна користуватися будь-якими шкалами вимірювання (частіше використовують відносну шкалу). У процесі попереднього оцінювання необхідно зафіксувати досягнення граничного задовільного рівня якості за кожним атрибутом, тобто класифікувати наявність властивості у ПС.

1.15. Форма опису заявлених властивостей програмних систем

1.15.1. Галузь застосування

Ця форма встановлює базові вимоги до складу, змісту та оформлення документів опису заявлених властивостей ПС для підготовки ПС до сертифікації відповідності. Вимоги цієї форми є обов'язковими для сертифікації ПС згідно із запропонованим способом оцінювання рівня якості.

Форму розроблено на підставі державних стандартів України ДСТУ 2462, ДСТУ 2844, ДСТУ 2850, ДСТУ 2851, ДСТУ 2853 та стандартів ISO/IEC 9126-1 і ISO/IEC 9126-2 (повні назви стандартів указано в нормативних посиланнях) [46; 45; 44; 48; 47; 49; 50].

1.15.2. Визначення

1. Документ вимог (*requirements document*) — зовнішній документ, що містить вимоги, рекомендації, обмеження або інструкції щодо нормованих характеристик ПС. Такими документами можуть бути технічні галузеві стандарти та інші нормативні документи для певного класу ПС, а також перелік вимог або ТЗ на розроблення ПС.

2. Опис програмної системи (*program system description*) — виготовлений розробником і замовником ПС внутрішній документ, що містить опис заявлених властивостей ПС. Документ має на меті надання допомоги потенційним користувачам в оцінюванні придатності ПС для задоволення їх потреб.

3. Документація користувача (*user documentation*) — повний набір документації у друкованій або недрукованій формі, що призначений для застосування або використання ПС користувачем.

1.15.3. Загальні вимоги

1. Заявник надає на сертифікацію по два комплекти опису ПС та документації користувача.

2. Заявлені властивості, що подаються в описі ПС, повинні бути приведені у відповідність з характеристиками стандарту ISO/IEC 9126. Властивостям ПС мають бути поставлені у відповідність атрибути підхарактеристик. Бажано, аби для показників властивостей були встановлені граничні умови.

3. Опис ПС та документація користувача вважаються приведеними у відповідність з міжнародним стандартом ISO/IEC 9126, якщо вони відповідають обов'язковим вимогам, наведеним у пп. 1.15.4 і 1.15.5 цієї форми. Вимоги, які надалі зазначені у дієслівній формі «повинний» є обов'язковими, а рекомендації, котрі зазначені у формі «може» — необов'язковими.

4. Пункти 1.15.4 і 1.15.5 містять:

– вимоги до опису ПС. Опис ПС повинен містити встановлену цією формою інформацію, а всі заяви опису повинні бути такими,

що перевіряються і є коректними (такими, що не мають взаємних суперечностей);

– вимоги до документації користувача.

1.15.4. Вимоги до опису програмної системи

Загальні вимоги до опису програмної системи

1. Кожна ПС повинна мати опис, який є частиною документації на ПС. Опис ПС є основою для проведення випробувань.

2. Опис ПС повинен бути доступним для всіх зацікавлених сторін.

3. Опис ПС повинен бути повним, зрозумілим, легким для перегляду і вільним від внутрішніх суперечностей.

4. Кожний термін опису повинен мати однаковий зміст.

5. Увесь подальший зміст і текст пп. 1.15.4–1.15.5 визначає, що повинно бути включеним і що доцільно включити в опис ПС.

Позначення та загальні відомості про програмну систему

1. Опис ПС повинен її ідентифікувати. Крім того, сам опис повинен позначатися унікальним ідентифікатором документа.

2. Ідентифікатор ПС повинен складатися щонайменше з назви ПС, номера її версії і/або дати виготовлення ПС.

3. Опис ПС повинен містити ім'я й адресу хоча б одного постачальника.

4. В описі ПС повинні бути ідентифіковані та описані всі варіанти використання, виконання яких забезпечує ПС.

5. В описі ПС повинні бути ідентифіковані всі програми, що постачаються замовнику ПС. Крім того, повинен бути визначений вид постачання: тексти програм, об'єктні або завантажувальні модулі.

6. Повинна бути наведена необхідна конфігурація технічних та програмних засобів, необхідних для функціонування ПС. Конфігурація повинна включати ім'я виробника і типові ідентифікатори всіх її частин.

7. Опис ПС може посилатися на документи вимог, яким повинна відповідати програмна система. У цьому випадку видання цих документів вимог повинні бути ідентифіковані в тексті опису ПС.

8. Заявником властивостей ПС можуть бути рекомендовані ті чи інші метрики зі стандарту ISO/IEC 9126-2 [50] для розрахунку показників атрибутів (або їх елементів) відповідних підхарактеристик

моделі якості ISO/IEC 9126-1 [49]. Остаточне рішення щодо вибору метрики приймає випробувальна лабораторія.

Заяви про функціональність

1. Заяви про функціональність (характеристика *functionality*) повинні описувати властивості ПС, що обумовлюють здатність ПС виконувати функції, які відповідають установленим потребам при її використанні в зазначених умовах.

2. Опис ПС повинен містити заяву про функціональну повноту (характеристику *functionality*, підхарактеристику *suitability*), тобто здатність ПС надавати користувачу належну множину функцій для вирішення специфікованих завдань і досягнення його цілей. В описі можуть надаватися рекомендації для розрахунку фактичних значень показників атрибутів (або їх елементів) підхарактеристики «функціональна повнота» з використання однієї з чотирьох метрик, наведених у табл. 8.1.1 стандарту ISO/IEC 9126-2.

3. Опис ПС повинен забезпечувати змістовний огляд кожної функції з належної множини функцій підхарактеристики *suitability*, у тому числі назву функції, короткий опис алгоритму (або визначених дій функції), вхідні та вихідні дані.

4. Опис ПС повинен містити заяву про точність (підхарактеристику *accuracy*) функцій ПС, тобто властивість, що надає ПС здатність забезпечувати правильні чи узгоджені результати або впливи з необхідним ступенем точності. Метрики підхарактеристики «точність» описано в табл. 8.1.2 стандарту ISO/IEC 9126-2.

5. Опис ПС повинен містити заяву про здатність ПС до взаємодії (підхарактеристику *interoperability*), тобто властивість, яка надає ПС здатність взаємодіяти з однією чи більше специфікованими системами. Метрики підхарактеристики «здатність до взаємодії» описано в табл. 8.1.3 стандарту ISO/IEC 9126-2.

6. Опис ПС повинен містити визначення систем для підхарактеристики *interoperability* та інтерфейсів між ними і ПС, що проходить сертифікацію.

7. Опис ПС повинен містити заяву про захищеність (підхарактеристику *security*), тобто властивість, яка надає ПС здатність забезпечити захист інформації від несанкціонованого доступу осіб чи систем на читання та модифікацію, та її доступність для осіб чи систем, які мають право доступу. Метрики підхарактеристики «захищеність» наведено в табл. 8.1.4 (*тут і надалі у підрозд. 1.15 наводяться посилання на таблиці з ISO/IEC 9126-2*).

8. Опис ПС повинен містити інформацію про засоби для запобігання неавторизованого доступу до програм і даних, якщо вони використовуються.

9. Опис ПС повинен містити заяву про узгодженість характеристики *функціональність* з нормативними документами (підхарактеристику *functionality compliance*), тобто властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для цього класу ПС. Метрики підхарактеристики «узгодженість функціональності» наведено в табл. 8.1.5.

Заяви про надійність

1. Заяви про надійність (характеристика *reliability*) повинні описувати властивості ПС, що зумовлюють здатність ПС зберігати рівень функціонування під час її роботи в зазначених умовах.

2. Опис ПС повинен містити заяву про завершеність (підхарактеристику *maturity*), тобто здатність ПС уникнути відмов через дефекти, що містяться в ній. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «завершеність» одну з восьми метрик, наведених в табл. 8.2.1.

3. Опис ПС повинен містити інформацію про засоби збереження даних у разі виникнення відмови ПС.

4. Опис ПС повинен містити заяву про стійкість до відмов (підхарактеристику *fault tolerance*), тобто властивість, що надає ПС здатності підтримувати встановлений рівень функціонування в умовах прояву дефектів у ПС, помилок даних або порушень специфікованого інтерфейсу. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «стійкість до відмов» метрики, наведених у табл. 8.2.2.

5. Опис ПС повинен містити заяву про відновлюваність (підхарактеристику *recoverability*), тобто властивість, що надає ПС здатності відновлювати функціонування на заданому рівні та відновлювати пошкоджені програми і дані. В описі можуть надаватися рекомендації для розрахунку фактичних значень показників атрибутів підхарактеристики «відновлюваність» з використання метрик, що описані в табл. 8.2.3.

6. Опис ПС може містити інформацію про засоби перевірки введення даних, захисту від істотних помилок користувача та відновлення після відмов.

7. Опис ПС повинен містити заяву про узгодженість характеристики *надійність* з нормативними документами (підхарактеристику *reliability compliance*), тобто властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для цього класу ПС. Метрики підхарактеристики «узгодженість» характеристики «надійність» наведено в табл. 8.2.4.

Заяви зі зручності використання

1. Заяви зі зручності використання (характеристика *usability*, іноді — здатність до використання) повинні описувати властивості ПС, що обумовлюють здатність ПС бути легко зрозумілою, освоєваною, зручною та привабливою для користувача при її використанні в зазначених умовах.

2. Опис ПС повинен містити заяву про зрозумілість (підхарактеристику *understandability*), тобто здатність ПС забезпечити користувачу можливість зрозуміти, чи дійсно ПС може задовольнити його потреби, як її можна застосовуватися для вирішення визначених завдань та які умови її використання. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «зрозумілість» одну з метрик, що описані в табл. 8.3.1.

3. Опис ПС повинен містити інформацію про знання, що необхідні користувачу для застосування ПС, зокрема знання в технічній сфері, знання операційної системи та інші спеціалізовані знання.

4. Опис ПС повинен містити заяву про освоєваність (підхарактеристику *learnability*), тобто властивість, що надає ПС здатності забезпечити користувачу можливість засвоїти прийоми її застосування. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «освоєваність» одну з метрик, що описані в табл. 8.3.2.

5. Опис ПС повинен містити заяву про керованість (підхарактеристику *operability*), тобто властивість, що надає ПС здатності забезпечити користувачу можливість керувати або контролювати її дії. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «керованість» одну з метрик, що описані в табл. 8.3.3.

6. Для забезпечення опису властивості «керованість» (підхарактеристика *operability*) опис ПС повинен містити інформацію про типи інтерфейсу користувача ПС (меню, вікна, командний рядок, функціональні клавіші), а також про функцію допомоги (*help*).

7. Опис ПС повинен містити заяву про привабливість (підхарактеристику *attractiveness*), тобто властивість ПС, яка забезпечує її привабливість для користувача. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «привабливість» одну з метрик, що описані в табл. 8.3.4.

8. Опис ПС повинен містити заяву про узгодженість характеристики *зручність використання* з нормативними документами (підхарактеристику *usability compliance*), тобто властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для цього класу ПС. Метрики підхарактеристики «узгодженість зручності» використання наведено в табл. 8.3.5.

Заяви про ефективність

1. Заяви про ефективність (характеристика *efficiency*) повинні описувати властивості ПС, що зумовлюють здатність ПС забезпечувати раціональне використання виділених ресурсів для роботи у встановлених умовах.

2. Опис ПС повинен містити заяву про реактивність (підхарактеристику *time behavior*), тобто здатність ПС забезпечувати належний час відгуку (відповіді) і оброблення завдань, а також рівень пропускну здатності під час виконання функцій у встановлених умовах їх використання, заданих системними специфікаціями. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «реактивність» одну з метрик, що описані в табл. 8.4.1.

3. Опис ПС повинен містити заяву про використовуваність ресурсів (підхарактеристику *resource utilization*), тобто властивість, що надає ПС здатність використовувати належні ресурси в потрібні періоди часу при виконанні своїх функцій у встановлених умовах використання. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «використовуваність ресурсів» одну з метрик, що описані в табл. 8.4.2.

4. Опис ПС повинен містити заяву про узгодженість характеристики *ефективність* з нормативними документами (підхарактеристику *efficiency compliance*), тобто властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для певного класу ПС. Метрики підхарактеристики «узгодженість ефективності» наведено в табл. 8.4.3.

Заяви про супроводжуваність

1. Заяви про супроводжуваність (характеристика *maintainability*) повинні описувати властивості ПС, що обумовлюють здатність ПС до ефективної модифікації. Модифікація може включати коригування, удосконалення або адаптацію ПС до змін середовища, вимог і функціональних специфікацій.

2. Опис ПС повинен містити заяву про аналізованість (підхарактеристику *analyzability*), тобто здатність ПС забезпечувати можливість діагностування її недоліків або причин відмов, а також ідентифікації частин, які повинні бути модифіковані. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «аналізованість» одну з метрик, що описані в табл. 8.5.1.

3. Опис ПС повинен містити заяву про модернізованість (підхарактеристику *changeability*), тобто властивість, що надає ПС здатність виконувати встановлені види модифікації. Метрики підхарактеристики «модернізованість» наведено в табл. 8.5.2.

4. Опис ПС повинен містити заяву про стабільність (підхарактеристику *stability*), тобто властивість, що надає ПС здатності мінімізувати несподівані ефекти модифікацій. Метрики підхарактеристики «стабільність» наведено в табл. 8.5.3.

5. Опис ПС повинен містити заяву про тестованість (підхарактеристику *testability*), тобто властивість, що надає їй здатності сприяти перевірці модифікованих програм. Метрики підхарактеристики «тестованість» наведено в табл. 8.5.4.

6. Опис ПС повинен містити заяву про узгодженість характеристики *супроводжуваність* з нормативними документами (підхарактеристику *maintainability compliance*), тобто про властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для певного класу ПС. Метрики підхарактеристики *узгодженість супроводжуваності* наведено в табл. 8.5.5.

Заяви про переносимість

1. Заяви про переносимість (характеристика *portability*) повинні описувати властивості ПС, що зумовлюють здатність ПС бути такою, яка може переноситися з одного середовища в інше.

2. Опис ПС повинен містити заяву про адаптованість (підхарактеристику *adaptability*), тобто властивість ПС, яка зумовлює її здатність адаптуватися для використання в різних специфікованих

середовищах без використання дій або засобів, що відрізняються від тих, які спеціально призначені для цих цілей. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «адаптованість» метрики, наведені в табл. 8.6.1 стандарту ISO/IEC 9126-2.

3. У разі якщо ПС може бути адаптована користувачем, опис ПС повинен містити інформацію про засоби для адаптації й умови їх використання (наприклад, заміна алгоритмів обчислень, зміна параметрів, зміна керувальних впливів, перепризначення функціональних клавіш тощо).

4. Опис ПС повинен містити заяву про налагоджуваність (підхарактеристику *installability*), тобто властивість ПС, яка зумовлює її здатність до інсталяції у специфікованому середовищі. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «налагоджуваність» метрики, що наведені в табл. 8.6.2 стандарту ISO/IEC 9126-2.

5. Опис ПС повинен містити інформацію про те, ким і як проводиться інсталяція ПС (розробником або користувачем). Повинно також бути пояснено, чи пропонується розробником супроводження ПС у процесі експлуатації, чи ні.

6. Опис ПС повинен містити заяву про сумісність (підхарактеристику *co-existence*), тобто властивість ПС, яка зумовлює її здатності співіснувати з іншими незалежними ПС у єдиному середовищі, розподіляючи спільні ресурси. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «сумісність» метрики, що наведено в табл. 8.6.3 стандарту ISO/IEC 9126-2.

7. Опис ПС повинен містити заяву про здатність до заміщення (підхарактеристику *replaceability*), тобто властивість, яка надає ПС здатності використовуватися замість інших специфікованих ПС у середовищі їх застосування. Опис може рекомендувати для розрахунку фактичних значень показників атрибутів підхарактеристики «здатність до заміщення» метрики, що наведено в табл. 8.6.4.

8. Опис ПС повинен містити заяву про узгодженість характеристики *переносимість* з нормативними документами (підхарактеристику *portability compliance*), тобто про властивість, яка забезпечує відповідність цієї характеристики вимогам відповідних стандартів для цього класу ПС. Метрики підхарактеристики «узгодженість переносимості» наведено в табл. 8.6.5.

1.15.5. Вимоги до документації користувача

1. Кожна ПС повинна мати документацію користувача, яка повинна бути повною, тобто містити всю інформацію, яка необхідна для використання ПС.

2. Усі функції, що заявлені в описі ПС, а також усі функції користувача повинні бути повністю описані в документації користувача.

3. У разі якщо інсталяція ПС може бути виконана користувачем, документація користувача повинна містити повну настанову з інсталяції.

4. У разі якщо супроводження ПС може бути виконано користувачем, документація користувача повинна містити повну настанову із супроводження.

5. Документація користувача повинна бути зрозуміла усіма користувачами, які правильно виконують заявлені варіанти використання ПС. Це може бути досягнуто за допомогою відповідного вибору термінів, коментарів та пояснень, а також шляхом посилань на корисні джерела інформації.

6. Доцільно аби документація користувача легко переглядалась та забезпечувала розуміння взаємозв'язку її частин (розділів).

7. Документація користувача повинна бути несуперечливою.



Контрольні запитання та завдання

1. Життєвий цикл ПС (ЖЦ ПС). Які настанови стандарту ISO/IEC12207: Information Technology — Software life cycle processes щодо інженерії вимог до ПС?
2. Каскадна та спіральна модель ЖЦ ПС. Як характеризується етап формування вимог до ПС?
3. Етап формування вимог до ПС. Які повна назва, структура та настанови стандарту IEEE-830?
4. Стандарт IEEE-830: Recommended practice for Software Requirements Specification. Зміст та основні засади стандарту. Як визначаються *S*-вимоги та *D*-вимоги?
5. Інженерія вимог як процес. Збирання вимог та аналіз вимог до ПС. Що таке *S*-вимоги та *D*-вимоги?
6. Аналіз вимог до ПС. Що таке функціональні та нефункціональні вимоги до ПС? Навести приклади.
7. Концептуальне моделювання проблеми. Визначення мети. Онтологія домену проблемної галузі. Навести приклад домену.

8. Моделі динамічних явищ домену та моделі алгоритмів. Навести приклад.
9. Об'єктно-орієнтована інженерія вимог. Характеристики та властивості. У чому полягає парадигма об'єктно-орієнтованого підходу до проектування та розроблення ПС?
10. Аспекти визначення якості ПС. Характеристики якості. Які три типи якості ви знаєте? Наведіть їх стислу характеристику.
11. Визначення якості ПС. Основні стандарти з якості ПС. Модель якості. Які цілі моделювання якості ПС?
12. Визначення моделі якості ПС. Зовнішня, внутрішня та експлуатаційна якість ПС. Визначення метрик якості. Типи метрик та шкал. Навести приклади.
13. Ядро професійних знань інженерії якості. Що таке SWEBOOK і PMBOOK? Який існує зв'язок між SWEBOOK і PMBOOK та ядром професійних знань інженерії якості?
14. Концепція підвищення якості ПС. Які принципи документа CMM та стандартів серії ISO9000? Ієрархічна модель якості ПС.
15. Основні чинні стандарти в галузі інженерії якості. У чому полягає базова мета інженерії якості ПС?



Завдання і вправи

I. Користуючись рекомендаціями стандарту ISO/IEC 9126-1, побудувати ієрархічну модель зовнішньої якості ПС (рис. 1.11), яка створюється для автоматизації функцій персоналу в межах заданої предметної галузі. Модель якості ПС має складатися з характеристик, підхарактеристик та атрибутів якості, що вимірюються за допомогою метрик. Обґрунтувати вибір та вилучення характеристик та підхарактеристик у процесі створення моделі якості ПС. Побудована модель якості має відповідати формулі (1.1). Знайти та узгодити попарно конфліктуючі між собою властивості якості (не менше ніж два атрибути підхарактеристик якості). Обрати вагові коефіцієнти для атрибутів підхарактеристик та характеристик якості. Обрати метрики для визначення рівня якості кожного атрибута відповідно до рекомендацій стандарту ISO/IEC 9126-2 і задати величини та міри цільової якості ПС. Для кожного атрибута побудувати шкалу оцінювання якості, рівні ранжування і визначити інтегральний показник цільової якості ПС.

На наступному кроці побудувати експлуатаційну модель якості, користуючись рекомендаціями стандарту ISO/IEC 9126-4. Для цього необхідно обрати характеристики та метрики якості, запропоно-

вані в стандарті ISO/IEC 9126-4, після чого оцінити рівень цільової експлуатаційної якості ПС з точки зору користувача.

Пропоновані дії виконати для таких предметних галузей:

1. *Проектна установа*. Ліцензована на розроблення архітектурних проектів і складається з декількох відділів. Кожен відділ очолює завідувач і в ньому працюють співробітники на певних посадах. Відділи мають назву (архітектурний, конструкторський тощо) і розташовані у певних приміщеннях. Співробітники працюють у конкретних приміщеннях і мають телефон. Приміщення мають номер, назву, площу, кількість місць. Співробітники виконують проекти, які мають номер і назву, дату початку і закінчення, а також керівника. Співробітник може виконувати декілька проектів. Виконані проекти затверджуються Державною архітектурною комісією, котра має голову, членів, адресу, телефон і розглядає та приймає (або відхиляє) архітектурні проекти в зазначені терміни.

2. *Акціонерне товариство*. Акціонерне товариство випустило акції, які купують власники акцій (акціонери). Збори акціонерів обирають правління акціонерного товариства, яке має прізвище, ім'я, по батькові (ПБ) голови правління, кількість членів, термін діяльності. Акціонерне товариство має назву, номер ліцензії, адресу, номер телефону. Акції характеризуються номером, категорією, датою емісії, номіналом. Власники акцій характеризуються номером, ПБ, адресою, номером паспорта. Власник може мати декілька акцій, а одна акція може мати декілька власників. За кожною акцією виплачуються дивіденди, які мають розмір, назву, дати виплат.

3. *Товари у кредит*. Меблевий магазин надає послуги з придбання товарів у кредит. Для оформлення кредиту покупець має надати довідку про доходи, а адміністратор магазину приймає рішення про видачу кредиту чи відхилення запиту. Під час прийняття рішення про видачу кредиту визначаються строк погашення кредиту та відсоткова ставка. Відповідно до цих даних формується графік погашення кредиту, у якому визначаються дати виплат за кредитом та суми виплат. Сформований графік передається до адміністрації магазину. У разі якщо клієнт не виплачує внесок у визначену дату, нараховується пеня за кожний день затримки. Якщо виплати за кредитом не проводяться протягом трьох місяців, менеджер магазину інформує адміністрацію, яка переводить дані боржника до колекторської фірми. Щомісяця менеджер формує звіт про

клієнтів магазину із зазначенням стану виплат. Указані вище дані менеджер заносить у базу даних (БД) інформаційної системи меблевого магазину та формує щоквартальний фінансовий звіт діяльності магазину, який подає адміністрації.

4. *Житлові комунальні контори (ЖКК).* У кожному місті є ЖКК, котрі здійснюють обслуговування жителів будинків, поставляючи газ, електроенергію та воду; ЖКК має номер, адресу, телефон, ПІБ директора. У ЖКК є декілька відділів (дирекція, бухгалтерія, інспектори із прийому громадян, водії та робітники різних спеціальностей), причому персонал має табельний номер, посаду, фах. Жителі будинків сплачують вартість комунальних послуг і мають ПІБ, ідентифікаційний код, адресу, домашній та мобільний телефон, списки номерів квитанцій про сплату. На балансі ЖКК, як правило, є декілька будинків, що характеризуються, адресою, кількістю поверхів, кількістю квартир, кількістю мешканців, наявністю котельної. На виконання робіт жителі будинку подають заявку, у якій вказують номер, дату, адресу й телефони мешканця та перелік замовлених робіт. На виконання робіт майстер оформляє накладну, у якій зазначає: номер, дату, адресу, ПІБ замовника, наявність пільг, ПІБ майстра, список матеріалів та перелік виконаних робіт із зазначенням сумарної вартості робіт.

5. *Робота студентської бібліотеки.* До студентської бібліотеки звертаються студенти, які бажають отримати навчальну літературу. Для оформлення замовлення необхідно отримати бланк замовлення, у якому зазначаються точне найменування книжок, автори та рік видання. Бланк замовлення можна отримати у бібліотекаря відповідного відділу. Якщо деякі реквізити книжок невідомі, то студент визначає їх, користуючись БД інформаційної системи бібліотеки. Заповнений бланк передається до відділу бібліотеки, у якому відбувається перевірка студента. Якщо студент має на руках більше ніж чотири книжки, то в отриманні п'ятої йому відмовляють. Перевірені заявки передаються до бібліотечного сховища, де за кожною заявкою працівник сховища або видає книжку, або у бланку зазначає інформацію про причину її відсутності. Розглянуті заявки студентів передаються до залу видачі книжок, де їх отримують студенти від бібліотекаря під запис у формуляр студента та підпис студента. Дані з формулярів студентів переносить у БД адміністратор бібліотеки. Щомісяця адміністратор формує звіт з описом книжок, виданих студентам, а також утрачених, загублених та зіпсованих книг.

6. *Пункт прокату.* На користування автомобілем клієнту бухгалтерією пункту прокату видається документ. Пункт прокату має ліцензію, назву, адресу, номер телефону. Бухгалтерія характеризується ПІБ головного бухгалтера, телефоном, кімнатою. Автомобіль має марку, дату випуска, реєстраційний номер, дату реєстрації. Клієнт має ПІБ, адресу, номер телефону, рік народження, серію та номер посвідчення водія. Для видачі документа надаються паспорт та посвідчення водія. Після цього бухгалтерія виписує чек, який оплачується користувачем у касі пункту прокату і пред'являється в бухгалтерії. Документ на користування має номер, назву, реєстраційний номер автомобіля, дані користувача, дату видачі. На один автомобіль може бути видано декілька документів на користування, один користувач може мати документи на декілька автомобілів.

7. *Приватні підприємства, що розробляють програмні продукти.* Характеризуються назвою, номером державної ліцензії, адресою, банківським рахунком. Підприємство має декілька відділів, де працюють співробітники (дирекція, бухгалтерія, відділ розроблення ПП, відділ тестування ПП, відділ супроводження та ін.). Програмний продукт проходить усі фази ЖЦ (формування вимог, проектування, реалізація, тестування, супроводження). Процес формування вимог складається із підпроцесів (збирання вимог, аналіз вимог, побудова формальних специфікацій вимог), які регламентовані в часі і мають документуватися. Програмний продукт поставляється замовнику і має назву, список розробників, вартість, документацію, дистрибутив.

II. Користуючись рекомендаціями стандарту ISO/IEC 25010 побудувати ієрархічну модель якості ПС (див. рис. 1.15), що створюється для автоматизації діяльності персоналу в межах заданої предметної галузі. Модель цільової якості ПС має містити характеристики та підхарактеристики якості. Обґрунтувати вибір та вилучення характеристик і підхарактеристик у процесі створення моделі якості ПС. Знайти та узгодити попарно конфліктуючі між собою властивості якості (не менше ніж дві підхарактеристики якості). Обрати вагові коефіцієнти для підхарактеристик та характеристик якості. На наступному етапі побудувати експлуатаційну модель якості ПС (див. рис. 1.16), користуючись рекомендаціями стандарту ISO/IEC 25010. Для оцінювання рівня цільової якості ПС з точки

зору користувача обрати характеристики та підхарактеристики якості, запропоновані в стандарті. Вибрати вагові коефіцієнти для підхарактеристик та характеристик якості. Зазначені дії виконати для таких предметних галузей:

1. *Підприємство «Будинок музики».* Це підприємство займається продажами, обслуговуванням і ремонтом музичних інструментів (електрогітар, клавішних та ударних інструментів), апаратури та обладнання. Крім того, підприємство організовує і проводить заходи, пов'язані з музичною діяльністю своїх клієнтів (концерти, джем-сейшени, майстер-класи тощо). Основні напрями діяльності: виконання продажів музичних інструментів, ремонт інструментів і обладнання відповідно до замовлень клієнтів, організація музичних заходів, навчання в групах та індивідуально. Покупець оформляє замовлення інструментів або послуг у продавця або через Інтернет.

2. *Видавництво.* У видавництві, що видає друковану продукцію, клієнти роблять замовлення. Видавництво має назву, номер ліцензії, адресу, ПІБ директора. Співробітники видавництва мають ПІБ, табельний номер, посаду, фах і працюють у різних відділах, котрі мають назву, список технічних засобів для друку, список комп'ютерної техніки. Перш ніж оформити замовлення, замовники вивчають прайси, у яких перераховані різні види й найменування продукції (книжки, журнали, газети, листівки, конверти, плакати та ін.), із зазначенням обсягів і термінів робіт, а також вартістю одиниці продукції. Замовники характеризуються іменем, адресою, кодом платника податків, рахунком у банку. Замовлення має номер, дату, список печатної продукції із зазначенням вартості одиниці продукції, термінів виконання робіт, сумою виконаних послуг, банківськими рахунками виконавця та замовника.

3. *Продаж залізничних квитків.* Продаж залізничних квитків здійснюється таким чином: клієнт запитує про наявність квитків на певну дату та напрям, додатково визначаючи тип вагона (плацкарт, купе тощо). Касир здійснює пошук у БД інформаційної системи за вказаними параметрами і повідомляє клієнту можливі варіанти. При цьому обрані позиції блокуються до закінчення продажу. Клієнт обирає з можливих варіантів ті, що його задовольняють, та повідомляє касиру паспортні дані для оформлення квитків. Касир оновлює дані в БД (придбані та розблоковані позиції) та отримує гроші за квитки. Також в інформаційній системі існує можливість

повернення квитка — у цьому випадку касир повертає клієнту гроші (за винятком деякої пені) та вилучає у клієнта оформлений квиток, після чого помічає його у базі як незайнятий. Пеня за квитки, які здають за місяць до дати використання, становить 1 % вартості, за 15 днів — 5 %, за 3 доби — 10 %, у день від'їзду — 50 %. Наприкінці кожного робочого дня касир формує звіт з інформацією про продані та повернуті квитки і передає звіт до центрального апарату адміністрації залізничних кас.

III. Користуючись рекомендаціями стандарту ISO/IEC 9126-1, побудувати ієрархічну модель зовнішньої якості, що складається з характеристик та підхарактеристик якості, для ПП MS PowerPoint 2013. Обґрунтувати вибір та вилучення характеристик і підхарактеристик у процесі створення моделі якості ПП. Побудована модель якості має відповідати формулі (1.1). Знайти та узгодити попарно конфліктуєчі між собою властивості якості (не менше ніж два атрибути підхарактеристик якості).

Обрати вагові коефіцієнти для атрибутів підхарактеристик та характеристик якості. Обрати метрики для визначення рівня якості кожного атрибута відповідно до рекомендацій стандарту ISO/IEC 9126-2. Порівняти рівень властивостей якості MS PowerPoint 2013 з відповідними властивостями MS PowerPoint 2010.

На наступному етапі побудувати експлуатаційну модель якості, користуючись рекомендаціями стандарту ISO/IEC 9126-4. Для цього необхідно обрати характеристики та метрики якості, запропоновані в стандарті ISO/IEC 9126-4, після чого оцінити рівень якості продукту MS PowerPoint 2013 з точки зору користувача. Порівняти рівень якості MS PowerPoint 2013 з якістю MS PowerPoint 2010.