

ФОРМАЛІЗАЦІЯ ХАРАКТЕРИСТИК ЯКОСТІ ПРОГРАМНИХ ПРОДУКТІВ ТА ЇХ УЗГОДЖЕННЯ ЗІ СПЕЦИФІКАЦІЯМИ ТЕСТУВАННЯ ДЛЯ СИСТЕМ КРИТИЧНОГО ТА КОМЕРЦІЙНОГО ЗАСТОСУВАННЯ

Вступ

Якість програмного забезпечення (ПЗ) сучасних комп'ютеризованих систем та комплексів є вадливою компонентою, що визначає конкурентоздатність програмних продуктів на ринку інженерії ПЗ. По-перше, через те, що наслідки постачання програмних продуктів (ПП) низької якості – це завжди матеріальні та фінансові втрати для користувачів, а по-друге, – це падіння престижу компанії-розробника. Ці проблеми, в свою чергу, негативно відображаються на конкурентоспроможності організації-розробника програмного продукту [1]. Відповідь на питання, як підвищити конкурентоспроможність ПП та як знизити ризик проєктів намагаються знайти не лише керівники організацій-розробників ПЗ та менеджери, але й замовники, що використовують ПП.

Постановка проблеми

Проблему забезпечення необхідного рівня якості ПЗ вже на стадії його розробки доцільно розглядати для двох окремих категорій ПЗ: створення якісних ПП: комерційного та критичного застосування. Варто відмітити той факт, що ПЗ критичного застосування має безпосереднє відношення до питань безпеки, оскільки кількість програмно реалізованих функцій в інформаційно-керуючих системах критичного застосування, до яких відносяться авіаційні та ракетно-космічні комплекси, атомні електростанції, нафтогазові комунікації, залізничний транспорт, медичні системи та ін., динамічно зростає. Крім того, збільшується частка аварій і катастроф, причиною яких є дефекти програмних засобів, які не були виявлені при розробці та верифікації ПЗ, однак проявилися при використанні системи за призначенням на етапі впровадження чи експлуатації. Як приклад досліджень у галузі якості та надійності систем критичного застосування можна відзначити роботи [2], в яких приведена класифікація відмов та їх причин для медичних програмно-апаратних комплексів, які відбулися внаслідок невиявлених на етапі тестування та верифікації дефектів ПЗ. У даному контексті необхідно підкреслити виняткову важливість якості та надійності ПЗ, оскільки від цього, по-перше залежить конкурентоздатність комерційних продуктів, а по-друге, безпечність систем критичного застосування для життя та здоров'я людини, навколишнього середовища та ін.

Більшість відомих робіт з моделювання якості таких відомих вчених, в Україні - Харченко В.С. [3], Коваль Г.І. [1], Тоценко В.Г. та ін.; у СНД - Липаев В.В. [4], Корольков [5] та ін.; за кордоном - Voehm [6] та ін., стосуються проблем забезпечення якості великих виробничих систем критичного застосування, що працюють у реальному часі.

Однак, ПЗ комерційного призначення має також широкий спектр застосування по відношенню до різноманітних сфер людського життя та на даному етапі розвитку вітчизняної програмної інженерії становить значну частку від програмних продуктів, що розробляються провідними ІТ-компаніями. Саме тому, **поставимо за мету** визначення та формалізацію характеристик якості ПЗ критичного та комерційного призначення для розв'язання задачі підвищення даного показника шляхом проходження продуктом процесу тестування. Для досягнення поставленої мети необхідно вирішити наступні задачі: 1) удосконалити поняття якості програмного продукту для формування узагальненого показника якості ПЗ; 2) проаналізувати спільні та відмінні риси існуючих стандартів якості ПЗ, як вітчизняних, так і міжнародних; 3) встановити взаємозв'язок між стандартами якості та специфікаціями тестування, що призначені для покриття основних характеристик якості ПЗ та, таким чином, забезпечують підвищення конкурентоздатності українських програмних продуктів на ринку інформаційних технологій.

В області якості існує безліч стандартів - національних, галузевих, відомчих, корпоративних. Міжнародним стандартом якості ПЗ є ISO 9126:2001 [7], який складається з 4 частин під загальним заголовком "Інформаційна технологія - характеристики та метрики якості програмного забезпечення". В Німеччині розглядають такі властивості якості ПЗ як: надійність, продуктивність, модифікуємість та ергономічність (зручність у використанні). У Франції J. Larrie запропонував розглядати гарантоване виконання специфікованих послуг (dependability), як аналог якості.

Для більш детального аналізу вимог щодо забезпечення якості будь-якої системи необхідно проаналізувати та порівняти вітчизняні та міжнародні стандарти якості ПЗ. Як відомо, стандарти якості ISO/IEC 9126 та ДСТУ 2850-94 [8] включають по шість базових (однакових) характеристик якості та містять множини певних підхарактеристик, які відрізняються між собою. На рис 1. відображено порівняльний аналіз стандартів якості ДСТУ 2850:1994 та ISO/IEC 9126-1:2001, останній з яких орієнтований на оцінку якості самого програмного продукту, та в якому виділяються такі складові якості ПЗ як: внутрішня, зовнішня якість та якість у використанні.

Модель зовнішньої та внутрішньої якості

ISO/IEC 9126-1

ДСТУ 2850-94



Рис.1. Порівняльний аналіз стандартів якості ДСТУ 2850:1994 та ISO/IEC 9126-1:2001

Для ґрунтового аналізу з використанням апарату теорії множин продемонструємо порівняння спільних та відмінних складових характеристик двох стандартів якості ПЗ: міжнародного та вітчизняного. Загальну ієрархічну модель якості ISO/IEC 9126 запишемо у вигляді множини:

$$Q = \left\{ C_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=6} \right\}_{i=1}^{\max(i)=6}, \quad (1)$$

де C_i – i -та характеристика (*characteristics*), для яких $i = [1.. \max(i) = 6]$; S_{ij} – j -та підхарактеристика (*sub-characteristic*) i -ї характеристики якості, для яких $j = [1.. \max(j) = 6]$; T_{ijk} – метрики вимірювання (*estimation techniques*) підхарактеристик якості, для яких $k = [1.. \max(k)]$.

Загальне визначення якості ПЗ у відповідності до складових її атрибутів на 1-му рівні, яка є спільною для ISO/IEC 9126 та ДСТУ 2850–94 має вигляд:

$$Q(I|U) = \{F_i(I|U), R_i(I|U), U_i(I|U), E_i(I|U), M_i(I|U), P_i(I|U)\}_{i=1}^6, \quad (2)$$

Першу характеристику якості ПЗ *функціональність (Functionality)* можна описати для ISO/IEC 9126 F_I та ДСТУ 2850–94 F_U відповідно: $F_I = \{S_t, A, I, S_c, C\}$ та $F_U = \{S_t, I, S_c, R_g, C, T_h\}$. Об'єднання - $F_I \cup F_U = \{S_t, A, I, S_c, R_g, C, T_h\}$, перетин - $F_I \cap F_U = \{S_t, I, S_c, C\}$ та асиметрична різниця $F_I \setminus F_U = \{A\}$, $F_U \setminus F_I = \{R_g, T_h\}$ для двох запропонованих стандартів якості ПЗ.

Наступна характеристика якості – *надійність (Reliability)* описується наступним чином: $R_I = \{M_t, F_t, R_c, C\}$ та $R_U = \{M_t, F_t, R_c, A, R_s, C\}$. Об'єднання - $R_I \cup R_U = \{M_t, F_t, R_c, A, R_s, C\}$, перетин - $R_I \cap R_U = \{M_t, F_t, R_c, C\}$ та асиметрична різниця $R_I \setminus R_U = \emptyset$, $R_U \setminus R_I = \{R_s, A\}$ для двох стандартів якості ПЗ, що аналізуються.

Так як, в стандарті якості ISO/IEC 9126 кожна характеристика містить одну і ту ж підхарактеристику – *узгодженість (Compliance)*, тобто $F_i, R_i, U_i, E_i, M_i, P_i \supseteq \{C\}$, то будемо позначати її однаково як C у всіх записах для атрибутів якості ПЗ, що аналізуються. В таблиці 1 приведений детальний аналіз всіх характеристик якості ПЗ для двох моделей, що аналізуються.

Таблиця 1. Порівняльний аналіз стандартів якості ПЗ для ISO/IEC 9126-1:2001 ДСТУ 2850–94

i	Характеристики стандартів якості ПЗ		Операцій над множинами характеристик моделей якості ПЗ		
	ISO/IEC 9126	ДСТУ 2850–94	Об'єднання (<i>max</i>)	Перетин (<i>min</i>)	Асиметрична різниця
1	$F_I \Rightarrow \{S_t, A, I, S_c, C\}$	$F_U \Rightarrow \{S_t, I, S_c, R_g, C, T_h\}$	$F_I \cup F_U \Rightarrow \{S_t, A, I, S_c, R_g, T_h, C\}$	$F_I \cap F_U \Rightarrow \{S_t, I, S_c, C\}$	$F_I \setminus F_U \Rightarrow \{A\}$ $F_U \setminus F_I \Rightarrow \{R_g, T_h\}$
2	$R_I \Rightarrow \{M_t, F_t, R_c, C\}$	$R_U \Rightarrow \{M_t, F_t, R_c, A, R_s, C\}$	$R_I \cup R_U \Rightarrow \{M_t, F_t, R_c, A, R_s, C\}$	$R_I \cap R_U \Rightarrow \{M_t, F_t, R_c, C\}$	$R_I \setminus R_U \Rightarrow \emptyset$ $R_U \setminus R_I \Rightarrow \{R_s, A\}$
3	$U_I \Rightarrow \{U_n, L, O, A_t, C\}$	$U_U \Rightarrow \{T_r, P_r, O, R_a, D\}$	$U_I \cup U_U \Rightarrow \{U_n, L, O, A_t, T_r, P_r, R_a, D, C\}$	$U_I \cap U_U \Rightarrow \{O\}$	$U_I \setminus U_U \Rightarrow \{U_n, L, A_t, C\}$ $U_U \setminus U_I \Rightarrow \{T_r, P_r, R_a, D\}$
4	$E_I \Rightarrow \{T_b, R_u, C\}$	$E_U \Rightarrow \{R_o, R_u\}$	$E_I \cup E_U \Rightarrow \{T_b, R_u, R_o, C\}$	$E_I \cap E_U \Rightarrow \{R_u\}$	$E_I \setminus E_U \Rightarrow \{T_b, C\}$ $E_U \setminus E_I \Rightarrow \{R_o\}$
5	$M_I \Rightarrow \{A_n, C_h, S_b, T_s, C\}$	$M_U \Rightarrow \{A_n, C_r, M_b, T_s\}$	$M_I \cup M_U \Rightarrow \{A_n, C_h, S_b, T_s, M_b, C_r, C\}$	$M_I \cap M_U \Rightarrow \{A_n, T_s\}$	$M_I \setminus M_U \Rightarrow \{C_h, S_b, C\}$ $M_U \setminus M_I \Rightarrow \{C_r, M_b\}$
6	$P_I \Rightarrow \{A_d, I_n, C_x, R_p, C\}$	$P_U \Rightarrow \{A_d, I_n\}$	$(P_I \cup P_U) = P_I \Rightarrow \{A_d, I_n, C_x, R_p, C\}$	$(P_I \cap P_U) = P_U \Rightarrow \{A_d, I_n\}$	$P_I \setminus P_U \Rightarrow \{C_x, R_p, C\}$ $P_U \setminus P_I \Rightarrow \emptyset$

Для спрощення запису (2) з врахуванням того, що реалізується загальний показник якості ПЗ для ISO/IEC 9126 та ДСТУ 2850–94 запишемо на 2-му рівні в загальному вигляді визначення якості:

$$Q = \left\{ \begin{array}{l} F_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)}, R_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)}, U_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)}, \\ E_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)}, M_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)}, P_i \left\{ S_{ij} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)} \end{array} \right\}_{i=1}^{\max(i)=6}, \quad (3)$$

Враховуючи, по-перше, часткову взаємозамінюємість деяких підхарактеристик якості ПЗ та по-друге, обмеженість ресурсів проекту, необхідно використовувати загальне визначення якості на 3-му рівні з позиції перетину всіх підхарактеристик атрибутів якості, тобто $\bigcap_{j=1}^{\max(j)} S_{ij}$:

$$Q_{\cap} = \left\{ \begin{array}{l} F_i \left\{ \{S_t, I, S_c, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=4}, R_i \left\{ \{M_t, F_t, R_c, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=4}, \\ U_i \left\{ \{O\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=1}, E_i \left\{ \{R\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=1}, \\ M_i \left\{ \{A_n, T_s\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=2}, P_i \left\{ \{A_d, I_n\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=2} \end{array} \right\}_{i=1}^{\max(i)=6}, \quad (4)$$

Хоча, останнє формулювання узагальненого показника якості ПЗ може виявитися недостатньо повним, по відношенню до того чи іншого стандарту якості у відповідності до вимог користувача. Тому, варто представити формування узагальненого показника якості ПЗ на 3-му рівні з позиції об'єднання всіх підхарактеристик атрибутів якості $\bigcup_{j=1}^{\max(j)} S_{ij}$:

$$Q_{\cup} = \left\{ \begin{array}{l} F_i \left\{ \{S_t, A, I, S_c, R_g, T_h, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=7}, \\ R_i \left\{ \{M_t, F_t, R_c, A, R_s, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=6}, \\ U_i \left\{ \{U_n, L, O, A_t, T_r, P_r, R_a, D, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=9}, \\ E_i \left\{ \{T_b, R_u, R_o, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=4}, \\ M_i \left\{ \{A_n, C_h, S_b, T_s, M_b, C_r, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=7}, \\ P_i \left\{ \{A_d, I_n, C_x, R_p, C\} \left\{ T_{ijk} \right\}_{k=1}^{\max(k)} \right\}_{j=1}^{\max(j)=5} \end{array} \right\}_{i=1}^{\max(i)=6}, \quad (5)$$

Недоліком останнього представлення якості ПЗ є можливість появи збитковості підхарактеристик якості, що необхідно забезпечити по відношенню до того чи іншого стандарту якості.

Як результат проведеного аналізу з двох стандартів якості ПЗ необхідно відзначити наступне: обидва стандарти якості місять однакові характеристики внутрішньої якості ПЗ, однак різні вимоги щодо якості за характеристикою – *зручність використання (Usability)*, тобто дана характеристика є спільною за найменшою кількістю підхарактеристик; найбільш подібні для обох стандартів якості ПЗ такі характеристики як: *функціональність (Functionality)* та *надійність (Reliability)*.

Для прикладу, детально опишемо основні характеристики та підхарактеристики якості стандарту ISO/IEC 9126-1:2001 [10], що впливають на формування узагальненого показника якості ПЗ критичного застосування:

- **Функціональність.** Підхарактеристики *функціональна повнота (Suitability)* і *точність (Accuracy)* є критичними, а *здатність до взаємодії (Interoperability)* – не критична, оскільки вона не впливає на оцінку функціонування системи в цілому. *Захищеність (Security)* можна опустити через те, що критичні системи працюють у сеансі з експертами, а не з користувачами. Зауважимо, що *узгодженість функціональності (Functionality Compliance)* з галузевими вимогами проводиться в обов'язковому порядку. Перші дві підхарактеристики (*Suitability та Accuracy*) звичайно з'являються після відображення галузевих вимог і вимог користувача на множині показників стандарту ISO/IEC 9126-1:2001 [7].

- **Надійність. Безвідмовність (Maturity)** – критична підхарактеристика, а стійкість до відмов (*Fault Tolerance*) і відновлюваність (*Recoverability*) можуть бути як критичними, так і ні. Це залежить від того, де програмний продукт використовується: в режимі реального часу, чи після функціонування об'єкта.
- **Зручність використання. Здатність до вивчення (Learnability) і зручність інтерфейсу (Attractiveness)** є другорядними підхарактеристиками (хоча інколи їх можна віднести і до критичних, особливо *оперуємість (Operability)*). *Узгодженість зручності використання (Usability Compliance)* являється критичною підхарактеристикою, бо потребує погодженості з цільовими вимогами користувача до ПЗ.
- **Ефективність.** Дана характеристика якості (*Efficiency*), в цілому, та її підхарактеристики (*Time Behavior*) та (*Resource Utilization*), зокрема, не впливають на властивості ПЗ, що відображають його внутрішню якість.
- **Супроводжуваність. Змінюваність (Changeability)** – критична підхарактеристика, бо змінюваність коду не повинна впливати на якість функціонування ПЗ в цілому.
- **Переносимість. Адаптуємість (Adaptability) та встановлюємість (Installability)** є другорядними підхарактеристиками, які необхідно відобразити у моделі, а на інші можна не враховувати.

Отже, сформуємо узагальнений показник якості для систем критичного застосування:

$$Q_{CRITICAL} = \left\{ \begin{array}{l} F_i \left\{ \{S_t, A, C\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=3}, R_i \left\{ \{M_t, R_c\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=2} \right\}, \\ U_i \left\{ \{L, O, A_t\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=3}, M_i \left\{ \{C_h\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=1} \right\}, \\ P_i \left\{ \{A_d, I_n\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=2} \right\} \end{array} \right\}_{i=1}^{\max(i)=6}, \quad (6)$$

Як приклад програмного продукту комерційного застосування, розглянемо один із видів комерційного ПЗ, що призначене для обробки даних та *націлене на вирішення задач таких класів* [9]: автоматизації адміністративної діяльності (системи організаційного управління); вирішення структурованих задач (розрахунково-оптимізаційні системи); пошуку і видачі необхідної інформації, наприклад інформаційно-пошукові системи); системи підтримки прийняття рішень та ін.

Вимоги до забезпечення якості ПЗ обробки даних можуть бути різними, але завжди спрямовані на досягнення таких основних підхарактеристик:

- **Функціональність. (Suitability, Security, Accuracy, Functionality Compliance)** – інформація повинна бути достовірною, точною та захищеною від спотворення, більш того надання доступу до інформації повинно виконуватися згідно з визначеними привілеями користувачів;
- **Надійність. (Fault Tolerance, Recoverability)** – система повинна бути стійкою до відмов та легко відновлювати втрачені дані;
- **Зручність використання. (Understandability, Learnability, Usability Compliance)** – система повинна бути зрозумілою користувачу, тобто зручною для роботи та легкою для вивчення, більш того, інформація та функції ПЗ після обробки повинні бути доступними і готовими для роботи;
- **Ефективність. (Resource Utilization)** – система повинна оптимально використовувати ресурси, враховуючи масштабність застосування та обсяги обробки інформації такими комерційними системами;
- **Супроводжуваність. (Changeability, Stability)** – система повинна працювати стабільно, однак бути гнучкою щодо внесення необхідних змін;
- **Переносимість. (Adaptability, Installability)** – система повинна легко встановлюватися та володіти властивостями адаптації до тих чи інших особливостей робочого середовища.

$$Q_{COMMERCIAL} = \left\{ \begin{array}{l} F_i \left\{ \{S_t, S_c, A, C\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=3}, R_i \left\{ \{F_t, R_c\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=2} \right\}, \\ U_i \left\{ \{U_n, L, C\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=3}, E_i \left\{ \{R_u\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=1} \right\}, \\ M_i \left\{ \{C_h, S_b\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=2}, P_i \left\{ \{A_d, I_n\} \{T_{ijk}\}_{k=1}^{\max(k)} \}_{j=1}^{\max(j)=2} \right\} \end{array} \right\}_{i=1}^{\max(i)=6}, \quad (7)$$

Фахівці в галузі інженерії ПЗ та користувачі сходяться в поглядах на програмний продукт низької якості як на такий, що: не забезпечує підтримку потреб користувачів або не достатньо надійний, гнучкий, зручний та супроводжуваний або надто дорогий і надто довго розроблюється [10]. Для **забезпечення необхідного рівня якості ПЗ** в міжнародній практиці знаходять застосування два підходи: орієнтований на продукт, що розробляється; орієнтований на процес розробки.

В першому акцент робиться на **оцінку якості шляхом тестування готового програмного продукту**. Цей підхід базується на припущенні, що чим більше знайдено та усунуто дефектів в ПЗ при тестуванні, тим вище його якість. Недоліки даного підходу полягають в наступному: усунення дефектів в готовому продукті на етапі тестування обходиться набагато дорожче, ніж їх своєчасне усунення на ранніх етапах ЖЦ ПЗ; відсутні методи і засоби тестування ПЗ, які гарантують вичерпне виявлення дефектів шляхом тестування [11].

При підході, що орієнтований на процес розробки, наголос робиться на **вживання заходів щодо запобігання, своєчасного виявлення і усунення дефектів в ПЗ**, починаючи з ранніх етапів життєвого циклу. Цей підхід покладений в основу концепції якості ПЗ міжнародних організацій ISO/IEC і реалізується в серії міжнародних стандартів, проектів стандартів і в матеріалах цих організацій.

Сучасний стан розвитку програмної інженерії, впровадження процесо-орієнтованого підходу до розробки ПЗ дозволяють сформулювати **ряд вимог до тестування ПЗ**, головними з яких є:

- проведення підготовки до тестування, починаючи з ранніх стадій життєвого циклу проекту;
- вибір та використання стратегій тестування, адекватних об'єктам, рівням тестування та ресурсним обмеженням проектів;
- стандартизація процесу тестування з урахуванням вимог процесо-орієнтованого підходу;
- визначення кількісних критеріїв завершення тестування.

Застосування традиційних підходів до тестування, орієнтованих на завершальні стадії розробки, призводить до тривалого ітераційного процесу масових доробок і повторного тестування. Це спричинює зростання ризиків, стосовних якості та надійності ПЗ, а також зрив проекту. Це стосується в першу чергу некритичних систем, на розробку та тестування яких (на відміну від критичних систем з підвищеними вимогами до надійності) виділяються обмежені ресурси щодо вартості проекту та часу.

Для проведення відповідності між характеристиками якості ПЗ та методами їх забезпечення – тестуванням ПЗ, проаналізовано та приведено на рис.2. основні специфікації тестування. Як видно, специфікації тестування лише частково забезпечують перевірку характеристик якості ПЗ [12].



Рис.2. Відповідність специфікацій тестування основним характеристикам якості ПЗ

Для **ПЗ комерційного застосування** для забезпечення покриття основних характеристик якості ПЗ, та у відповідності до специфікацій тестування [12], що приведені на рис.3. необхідно проводити наступні види тестування з використанням, наприклад інструментів IBM Rational:

- **Функціональне тестування (Functional testing)**: цілісності даних (Data integrity testing); на різних платформах (Configuration testing); відмовостійкості (Failover & recovery testing); доступу

(Security testing); інсталяційне (Installation testing); інтерфейсу користувача (User interface testing).

- Навантажувальне тестування (Load testing): профілювання продуктивності (Performance profiling); циклу роботи (Business cycle testing); при великому користувальницькому навантаженні (Stress testing); на великих обсягах даних (Volume testing).

Для **систем критичного призначення** необхідно проводити наступні види тестування [13], обов'язково враховуючи вимоги до надійності такого виду систем: структурне (покриття розгалужень, функцій); надійності (функціональність, перевірка компонентів); функціональне (відповідність функціональним вимогам); регресійне (перевірка можливості внесення змін); навантажувальне (оцінка статистичних характеристик системи); стресове (коректність роботи при граничних навантаженнях). Перелік видів тестування для систем критичного застосування можна значно розширити, у відповідності до вимог замовника, однак необхідно враховувати обмеження на вартість і час реалізації проекту.

Висновки

Отже, результатом проведеного дослідження є математична формалізація характеристик якості ПЗ у відповідності до міжнародного та вітчизняного стандартів ISO/IEC 9126:2001 та ДСТУ 2850:94. Особливістю застосування отриманих результатів є їх узгодженість з підходами щодо визначення якості програмного забезпечення критичного та комерційного призначення.

В процесі проведення дослідження: проаналізовано поняття якості програмного продукту з метою формування узагальненого показника якості ПЗ для програмного забезпечення в цілому та комп'ютеризованих систем критичного та комерційного застосування зокрема; 2) відзначено спільні та відмінні риси існуючих вітчизняних та міжнародних стандартів якості ПЗ; 3) приведено загальноприйняті специфікації тестування ПЗ у відповідності до міжнародних стандартів, що призначені для покриття основних характеристик якості ПЗ; 4) встановлено взаємозв'язок між стандартами якості та специфікаціями тестування, взаємоузгодженість яких забезпечить підвищення конкурентоздатності українських ПЗ на ринку інформаційних технологій; 5) запропоновано перелік видів тестування для систем критичного та комерційного призначення з метою забезпечення якості ПЗ. В подальшому пропонується розширення методик підвищення якості ПЗ шляхом застосування новітніх методологій інженерії програмних продуктів, таких як Test Driven Development та ін.

Література

1. Основы инженерии качества программных систем / Ф.И. Андон, Г.И. Коваль, Т.М. Коротун, В.Ю. Суслов / Под ред. И.В. Сергиенко. – К.: Академперіодика. - 2002. - 504 с.
2. Volkova S.O., Trunov O.M. Medical systems: quality and safety in robotic surgery, Collected Abstracts 10th All-Russian Scientific and Technological Conference with International Participation "Extreme Robotics", April 8-9, Saint-Petersburg, 2008, pp. 74-75.
3. Харченко В.С., Тарасюк О.М. Оценка экспертизы программного обеспечения: показатели, методика и инструментальные средства. Информационные технологии и безопасность // Сб. научн. тр. – К.НАНУ, Ін-т проблем реєстрації інформації, 2003. – Вып. 4. – С. 128-139.
4. Липаев В.В. Обеспечение качества программных средств. Методы и стандарты. – М.: СИНТЕГ, 2001. – 380 с.
5. Корольков Ю. Д.. Математические модели качества программных средств. – Иркутск: Издательство иркутского университета, 1995. – 160с.
6. W. Boehm, J. R. Brown, H. Kaspar, M. Lipow, G. MacLeod, and M. J. Merritt. Characteristics of Software Quality. North Holland, 1978.
7. ISO/IEC 9126-1. Software engineering – Product quality – Part 1: Quality model, 2001. – 26 p.; ISO/IEC TR 9126-2. Software engineering – Product quality – Part 2: External metrics, 2003. – 86 p.; ISO/IEC TR 9126-3. Software engineering – Product quality – Part 3: Internal metrics, 2003. – 66 p.; ISO/IEC TR 9126-4. Software engineering – Product quality – Part 4: Quality in use metrics, 2004. – 70 p.
8. ДСТУ 2850-94. Програмні засоби ЕОМ. Показники та методи оцінювання якості. – Чинний від 01.01.96. –К.: Дежстандарт України, 1994. – 20 с.
9. Бернатович О.В., Коваль Г.И., Коротун Т.М. Функциональное тестирование программного обеспечения информационно-расчетного типа // Проблемы программирования. – 1998. - № 3. – с. 51-58.
10. Канер С., Фолк Д., Нгуен Е.К. Тестирование программного обеспечения: Пер с англ. - К.: DiaSoft. – 2000. – 544 с.
11. Майерс Г. Искусство тестирования программ. – М: Финансы и статистика. - 1982. – 176 с.
12. Software Engineering Body of Knowledge (SWEBOK). // ISO/IEC JTC1/SC7 N2517. Software & System Engineering Secretariat, Canada, 2001. – 220 p.
13. S.O. Volkova, O.M. Trunov The analysis of methods of software quality and reliability estimation for medical diagnostic systems, Abstracts of IV International Conference «Theoretical and Applied Aspects of Program Systems Development» (TAAPSD'2007), 04.09.2007 – 09.09.2007 // с.63-68.

С.О. Волкова, О.В. Гнездьонова

ФОРМАЛІЗАЦІЯ ХАРАКТЕРИСТИК ЯКОСТІ ПРОГРАМНИХ ПРОДУКТІВ ТА ЇХ УЗГОДЖЕННЯ ЗІ СПЕЦИФІКАЦІЯМИ ТЕСТУВАННЯ ДЛЯ СИСТЕМ КРИТИЧНОГО ТА КОМЕРЦІЙНОГО ЗАСТОСУВАННЯ

Анотація

В статті удосконалено поняття якості програмного забезпечення та формалізовано характеристики внутрішньої якості програмних продуктів. Описано узагальнений показник якості для систем критичного та комерційного призначення шляхом порівняння міжнародного та вітчизняного стандарту якості. Визначено поняття тестування, як один з основних методів забезпечення якості програмних продуктів. Проаналізовано основні специфікації тестування та приведено види тестування для систем критичного та комерційного застосування з метою забезпечення базових характеристик якості програмних продуктів.

Ключові слова: якість та надійність програмного забезпечення, тестування програмного продукту, специфікації тестування, зовнішня та внутрішня якість, дефект, відмова, розробка продукту через тестування.

С.А. Волкова, О.В. Гнездьонова

ФОРМАЛИЗАЦИЯ ХАРАКТЕРИСТИК КАЧЕСТВА ПРОГРАММНЫХ ПРОДУКТОВ И ИХ СОГЛАСОВАНИЕ СО СПЕЦИФИКАЦИЯМИ ТЕСТИРОВАНИЯ ДЛЯ СИСТЕМ КРИТИЧЕСКОГО И КОММЕРЧЕСКОГО ПРИМЕНЕНИЯ

Аннотация

В статье усовершенствовано понятие качества программного обеспечения и формализованы характеристики внутреннего качества программных продуктов. Описан обобщенный показатель качества для систем критического и коммерческого назначения путем сравнения международного и отечественного стандарта качества. Определено понятие тестирования, как один из основных методов обеспечения качества программных продуктов. Проанализированы основные спецификации тестирования и приведены виды тестирования для систем критического и коммерческого применения с целью обеспечения базовых характеристик качества программных продуктов.

Ключевые слова: качество и надежность программного обеспечения, тестирование программного продукта, спецификации тестирования, внешнее и внутреннее качество, дефект, отказ, разработка продукта через тестирование.

S.O. Volkova, O.V. Gnezdyonova

SOFTWARE QUALITY ATTRIBUTES' FORMALIZATION AND ITS AGREEMENT WITH TESTING SPECIFICATION FOR COMMERCIAL AND CRITICAL SYSTEMS

Annotation

The conception of software quality is improved. The software internal quality attributes are formalized. The overall quality index for commercial and critical software, by means of comparing different quality standards, is described. The conception of software testing, as main method for software quality providing, is defined. Main software testing specifications, methodologies and types for commercial and critical software are analyzed and devoted to guaranteeing of essential software quality attributes.

Key words: software engineering, software quality and reliability, software testing, internal and external quality, software quality in use, testing specifications, software failure, fault and defect, test driven development.