ISSN 1814-4225 (print).
ISSN 2663-2012 (online)

УДК 004.415.2.05 **doi: 10.32620/reks.2019.1.07**

О. Є. ФЕДОРОВИЧ[1], Ол. В. БАБИЧ, Ол-р В. БАБИЧ[2]

[1] ***Національний аерокосмічний університет ім. М.Є. Жуковського «ХАІ», Україна***
[2] ***Полтавський політехнічний коледж, Україна***

ЗАБЕЗПЕЧЕННЯ ЯКОСТІ РОЗРОБКИ ПРОГРАМНИХ СИСТЕМ З ВИКОРИСТАННЯМ КОМПОНЕНТНОГО ПРОЕКТУВАННЯ ТА ТЕХНОЛОГІЇ P-MODELING FRAMEWORK

В публікації описано результати дослідження процесу створення сучасних програмних систем (ПС) з використанням компонентного підходу та технології проектування P-Modeling Framework. Обґрунтовано актуальність створення сучасного методу управління проектом зі створення ПС, який забезпечить якість процесу проектування та кінцевого продукту у вигляді програмної системи. Метою роботи є дослідження забезпечення якості проектування програмних продуктів та ПС шляхом розробки прикладної інформаційної технології для створення компонентної архітектури ПС з використанням сучасної технології P-Modeling Framework. За допомогою системного аналізу сформовано багаторівневу архітектуру ПС. Використано методи експертного оцінювання для прогнозування та оцінки якості створюваної ПС. Розроблено моделі оптимізації архітектури ПС. Використано методи теорії прецедентів для формування множини альтернативних варіантів ПС. Формалізовано та удосконалено метод зворотного семантичного трасування (RST) для забезпечення якості процесу проектування ПС. Описано структуру та особливості прикладної інформаційної технології, створеної з використанням P-Modeling Framework. Розроблені моделі та методи інформаційної технології є науково-методичною основою для вирішення завдань забезпечення якості в процесі створення програмних продуктів та систем з використанням позитивного досвіду минулих проектів у вигляді компонент повторного використання.

Ключові слова: *якість проектування; компонентне проектування; системна модель компонентного проектування; ітеративне проектування; зворотне семантичне трасування.*

Вступ

Більшість проблем, які виникають при проектуванні складних програмних продуктів спричинені помилками, що були допущені на ранніх етапах розробки. Тому більш уваги приділяють якості процесу проектування програмних систем, створенню методів забезпечення якості розробки. Тому тема публікації у якій ставиться та вирішується завдання розробки методів та інформаційної технології забезпечення якості розробки програмних систем (ПС) є актуальною.

1. Аналіз досліджень і публікацій

Існуючі на даний момент методи, методології, підходи та технології проектування програмних систем не забезпечують в повній мірі необхідні передумови для створення якісних програмних продуктів [1, 2]. В існуючих стандартах якості (ISO/IEC25010:2011) містяться вимоги та показники якості, але не вказано якими методами забезпечується якість створюваних програмних продуктів [3].

В останні роки на основі стандартів ISO серії 9000 сформувалась комплексна система управління якістю продукції TQM (Totally Quality Management). Застосування цього комплексу може служити основою для систем забезпечення якості ПС, однак потребує коригування та адаптації у відповідності до специфіки галузі [4]. Крім того, в ході розробки систем якості проектування ПС доцільним є залучення низки стандартів, які формально не стосуються цієї серії, однак регламентують показники якості, ЖЦ, верифікацію і тестування, документування ПС та інші аспекти, специфічні для галузі розробки ПС.

Від компаній-розробників замовники вимагають відповідності технологій проектування, розробки та якості продукції сучасним міжнародним стандартам, серед яких найчастіше згадуються методологія Microsoft Solutions Framework (MSF) – в гнучкому варіанті, або MSF CMMi, DevOps, UML, CMMi, тощо. Використання таких стандартів є необхідною умовою співпраці, гарантією отримання певної віддачі від інвестицій в проект, та дозволяє обрати найнадійніших партнерів для реалізації ПС. Нагадаємо, що інтегровану модель зрілості процесів

© О. Є. Федорович, Ол. В. Бабич, Ол-р В. Бабич, 2019

(CMMi) свого часу було створено саме для того, щоб гарантувати віддачу від капіталовкладень в оборонні проекти та полегшити вибір надійних підрядників з розробки ПС для Міністерства оборони США [5].

Сучасна методологія Microsoft Solutions Framework (MSF) та Unified Modeling Language (UML) дозволяють забезпечити якість програмних систем, що створюються. MSF має перевірені методики планування, проектування, розробки та впровадження успішних рішень в галузі інформаційних технологій (IT-рішень) [5]. MSF пропонує ітеративний підхід до процесу розробки та вимагає гнучкості під час проектування та документування ПС. В рамках MSF пропонується ряд шаблонів стандартних документів, які можуть бути використані для планування і контролю процесу розробки.

Говорячи про типові засоби керування проектами та проектування програмного забезпечення крім методології MSF та графічної нотації UML необхідно відокремити P-Modeling Framework (P-MF) – поки що маловідому надбудову над MSF, яка, проте, вже добре зарекомендувала себе на практиці, але потребує подальшого розвитку та формалізації [6, 7].

Незважаючи на наявність інструментарію та методологічної підтримки процесу проектування, відчувається брак формальних методів і технологій для забезпечення якості проектування програмних продуктів. P-Modeling Framework була спробою вирішити цю проблему, однак її розвиток і практичне впровадження значно сповільнились через відсутність формальних методів, моделей, критеріїв та технологій оцінки якості проектних артефактів [8 - 10]. Тому дослідження, яким ставиться та вирішується науково-прикладне завдання розробки інформаційної технології для забезпечення якості проектування створюваних програмних продуктів та систем, є актуальним.

2. Постановка завдання дослідження

Метою дослідження є розробка методів прикладної інформаційної технології забезпечення якості розробки програмних продуктів та систем з використанням компонентного проектування та сучасної технології P-Modeling Framework [11, 12].

Для досягнення мети дослідження необхідно:

1. Створити модель компонентного проектування для забезпечення якості процесу розробки програмної системи.

2. Формалізувати технологію проектування P-Modeling Framework.

3. Удосконалити метод зворотного семантичного трасування (RST).

4. Розробити прикладну інформаційну технологію забезпечення якості розробки програмних продуктів та систем.

Розроблені моделі, методи та прикладна інформаційна технологія є науково-методичною основою для вирішення практичних завдань забезпечення якості під час створення програмних продуктів з урахуванням компонентної архітектури програмної системи.

3. Рішення завдання дослідження

В ході розробки програмних систем (ПС) широко використовується компонентний підхід, який дозволяє скоротити тривалість процесу за рахунок повторного використання компонент. Компонентний підхід чудово поєднується з P-MF та мовою проектування UML, яка є фактичним стандартом у галузі проектування ПС.

Сучасна складна архітектура ПС вимагає вдосконалення методів і моделей компонентного проектування для керування та якісного виконання проектів вчасно, у рамках бюджету та з належним рівнем функціональності. Цей підхід підтримується більшістю сучасних методологій зі створення програмних продуктів, зокрема MSF.

Сучасні ПС є складними багаторівневими системами. Створення ПС є розробка складного продукту з архітектурою у вигляді ієрархії структурних елементів.

Прості компоненти є елементами нижнього рівня деталізації ПС (наприклад, утиліти або бібліотеки), складні – елементами середніх і верхнього рівнів (системи, підсистеми, фреймворки).

Компоненти в інформаційному поданні є елементарними інформаційними складовими проекту. В роботі виділені три типи компонент, які є елементами багаторівневої архітектури ПС: повторно використовувані компоненти (ПВК); нові компоненти (НК), які можуть забезпечити інноваційність проекту; складні компоненти (СК).

В реальному проекті багаторівневої інформаційної системи використовують усі типи перелічених компонент, а також їх комбінації. Наявність великої кількості нових компонент може вплинути на ризики та якість проектування, тобто на реалізованість IT-проекту.

На рис. 1 запропоновано системне уявлення проектування ПС, в яке інтегровано технологію P-MF для забезпечення якості, компонентну багаторівневу архітектуру ПС, та організаційну структуру управління проектом (рис. 1).

ISSN 1814-4225 (print).
ISSN 2663-2012 (online)

Виділено два зв'язаних процеси проектування, в яких задіяні різні команди:

Рис. 1 Системне уявлення багаторівневого компонентного проектування ПС

– процес створення базової архітектури ПС (верхній рівень);

– процес реалізації конкретного проекту ПС (нижній рівень).

Системну модель створення складної багаторівневої компонентної ПС з задоволенням вимог щодо забезпечення належного рівня якості можна представити у вигляді кортежу:

$$S = \{T, PK, NK, SK, A, B, KT, IP, VP, SV, Q\}, \quad (1)$$

де T – вимоги замовника;

PK – множина ПВК на всіх рівнях архітектури ПС;

NK – множина нових (створюваних) компонент на всіх рівнях ПС;

SK – множина складних компонент ПС;

A – базова архітектура ПС;

B – архітектура нової ПС;

KT – компонентна технологія створення ПС; IP – інтеграційний процес формування ПС з врахуванням групування компонент;

VP – високорівневе технологічне проектування на базі P-MF задля забезпечення якості розробки ПС;

SV – організаційна структура керування проектом;

Q – задоволення вимог якості на кожному етапі розробки ПС.

Запропоновано метод створення багаторівневої компонентної архітектури ПС, який використовує прецедентний підхід. Для цього сформовано багаторівневу базу прецедентів (БП) у вигляді перевірених проектних рішень та готових компонент, які відповідають рівням архітектури ПС. Пошук в БП компонент для нового проекту здійснюється «згори-донизу» за допомогою лексикографічного впорядкування вимог (якість, інноваційність, витрати, час, ризик), з урахуванням «близькості» варіантів (прецедентів) до потрібних компонент проекту.

Для складної багаторівневої архітектури запропоновано оптимізацію вибору варіантів з метою забезпечення якості ПС, що проектується. Використовується метод цілочисельного лінійного програмування з булевими змінними.

Введемо булеву змінну $x_{i_{je}} = 1$, якщо для і-го рівня архітектури нового проекту зі створення ПС та забезпечення j-ї функціональності з множини M_{ij} КПВ у БП взято е-й елемент зі своїми характеристиками, з врахуванням оцінки «близькості». В іншому випадку (якщо ми не зупинили вибір на е-му елементі) $x_{i_{je}} = 0$.

Цільова функція буде враховувати «близькість» (ступінь задоволення проектних вимог) і буде виглядати наступним чином:

$$F = \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{e=1}^{V_{ij}}\left(\alpha_1 P_{1_{ije}} + \alpha_2 P_{2_{ije}} + \alpha_3 P_{3_{ije}} + \alpha_4 P_{4_{ije}} + \alpha_5 P_{5_{ije}}\right) x_{i_{je}}, \quad (2)$$

де α_l – ваговий коефіцієнт, який враховує важливість «близькості», пов'язаної з l-м показником нового проекту зі створення ПС;

n – число рівнів ієрархії нової створюваної ПС;

n_i – число функціональних компонент на і-му рівні ПС;

V_{ij} – кількість КПВ в БП ПС для використання в нових проектах;

$P_{1_{ije}}$ – ступінь «близькості» (задоволення вимог) $r_{i_{je}}$ програмної компоненти, взятої у вигляді КПВ;

$P_{2_{ije}}$ – ступінь «близькості» $r_{i_{je}}$ значень технічних характеристик (ТХ);

$P_{3_{ije}}$ – ступінь «близькості» вимог з використання $r_{i_{je}}$;

$P_{4_{ije}}$ – ступінь «близькості» $r_{i_{je}}$ експлуатаційних вимог;

$P_{5_{ije}}$ – ступінь «близькості» $r_{i_{je}}$ вимог з масштабованості.

Обмеження, в порядку їх значимості, будуть виглядати наступним чином:

Вимоги з забезпечення якості створюваної ПС:

$$Q = \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{e=1}^{V_{ij}} q_{i_{je}} x_{i_{je}} .$$

Обмеження стосовно витрат на адаптацію відібраних для нового проекту КПВ:

$$W = \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{e=1}^{V_{ij}} w_{i_{je}} x_{i_{je}} . \qquad (3)$$

Вимоги стосовно термінів розробки ПС:

$$T = \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{e=1}^{V_{ij}} t_{i_{je}} x_{i_{je}} . \qquad (4)$$

Вимоги стосовно ризиків, пов'язаних з розробкою ПС:

$$R = \sum_{i=1}^{n}\sum_{j=1}^{n_i}\sum_{e=1}^{V_{ij}} \mu_{i_{je}} x_{i_{je}} , \qquad (5)$$

де $q_{i_{je}}$ – експертна оцінка якості програмної $r_{i_{je}}$ КПВ;

$w_{i_{je}}$ – витрати на придбання та адаптацію $r_{i_{je}}$ КПВ;

$t_{i_{je}}$ – час, витрачений на адаптацію $r_{i_{je}}$ КПВ для конкретного проекту;

$\mu_{i_{je}}$ – ризики з адаптації $r_{i_{je}}$ КПВ для конкретного проекту.

Під час оптимізації, слід максимально задовольнити вимоги нового проекту зі створення ПС (тобто здійснити максимізацію «близькості» F): $\max F$ за умови виконання наступних обмежень:

$$Q \geq Q', W \leq W', T \leq T', R \leq R' ,$$

де Q', W', T', R' – допустимі значення показників якості, витрат, часу та ризиків проекту зі створення нової ПС.

Для забезпечення якості в ході проектування програмної системи сформована послідовність дій розробників при створенні багаторівневої архітектури системи. Дії виконуються «згори-донизу» з пошуком в багаторівневій базі прецедентів «близьких» до потрібних в проекті ПС компонент починаючи з верхніх рівнів архітектури. На кожному і-му рівні багаторівневої архітектури програмної системи проводиться аналіз наявності в базі прецедентів повторно використовуваних компонент. Відсутність їх приводить до необхідності розробки нових, що пов'язані з ризиками проекту.

Далі проводиться прогнозування і оцінка якості ПС, що створюється. В загальному випадку, для компонентної архітектури нової ПС, багаторівневий склад якої сформовано за допомогою множини КПВ, можна отримати оцінку у вигляді прогнозу можливості задоволення вимог стосовно якості наступним чином:

$$P_S = \lambda_S \left[\prod_{v=1}^{V} \mu_v \cdot \prod_{w=1}^{W} P_w \right],$$

де μ_v – ймовірнісна оцінка експертів задоволення вимог, яка стосується підсистем, взятих у вигляді готових КПВ з БПР та, можливо, адаптованих до нової ПС;

P_w – стосується підсистем, які побудовані шляхом комплексування КПВ, починаючи з найнижчих рівнів деталізації нової ПС.

Удосконалено метод зворотного семантичного трасування (RST). Сформовано наступну послідовність дій проектувальників:

1) використовуючи початковий опис (ПО) програмної системи, зусиллями команди розробників базової платформи (БП) створюється модель системи для нової ПС у вигляді множини діаграм (класів, прецедентів, послідовностей, тощо);

2) далі використовується удосконалений метод RST, в якому команда розробників (інша команда), члени якої детально не ознайомлені з вимогами та описом предметної області ПС, повинні на основі моделі, створеної під час першого етапу, відновити початковий опис системи (ВО) для нового проекту (НП);

3) наступний етап – перевірка відповідності ПО та ВО, отриманих розробниками БП та НП. Якщо ступінь співпадання цих документів достатньо великий (вводиться критерій «близькості»), то у відповідності до RST приймається рішення про початок роботи над проектом командою виконавців НП.

В іншому випадку (відсутність близькості) ПО та ВО, здійснюється доопрацювання моделі системи командою розробників БП. Після цього процедура RST повторюється, і якщо відповідності ПО та ВО знову не спостерігається, уточнення моделі та процедура RST повторюються знову (чергова ітерація процесу), аж поки ми не отримаємо близькі документи.

Для оцінки близькості описів ПС, отриманих від розробників БП та НП, для кожної з ознак p_i користуємось як кількісними, так і якісними оцінками. При використанні якісних оцінок експертів формується лексикографічно упорядкований список «слів» для оцінки «близькості» описів

ISSN 1814-4225 (print).
ISSN 2663-2012 (online)

різних команд розробників ПС до введеного «контрольного слова близькості».

Розроблені моделі та методи було використано як формальний математичний інструментарій для створення прикладної інформаційної технології забезпечення якості програмних систем (ПІТЗЯПС). Створена прикладна інформаційні технологія забезпечення якості ПС складається з таких компонент:

– блок файлового вводу-виводу;

– блок відсіювання незначущих слів в документах ПО та ВО;

– блок підрахунку кількості сутностей в ПО та ВО;

– блок прогнозування і оцінки якості для здійснення RST;

– блок візуалізації результатів оцінки якості проектування.

На рис. 2 представлено структурно-логічну схему ПІТЗЯПС. На відміну від існуючих, розроблена прикладна інформаційна технологія оцінки якості проектування програмних продуктів базується на методологічних засадах багаторівневого компонентного проектування, моделей оптимізації архітектури програмної системи з урахуванням вимог якості, багаторівневої бази прецедентів для вибору та використання кращих програмних компонент, методі зворотного семантичного трасування, що дозволяє забезпечити контроль якості проектування програмних систем.

Розроблена прикладна інформаційна технологія розповсюджується на безоплатній основі та є продуктом з відкритими програмними кодами.

Подальший розвиток результатів дослідження та впровадження прикладної інформаційної технології, яка напрямлена на забезпечення якості програмних продуктів та систем, тісно пов'язаний з підвищенням конкурентоспроможності вітчизняних програмних розробок та використанням їх не тільки в Україні, але й за кордоном.

Висновки

В роботі проведено дослідження процесу створення програмних систем з використанням компонентного підходу та технології P-Modeling Framework, що дозволяє забезпечити якість проектування ПС. Сформульоване системне уявлення та модель компонентного проектування. Розроблена модель формування якісного складу компонент архітектури ПС. Удосконалено та формалізовано метод для планування та контролю дій проектувальників ПС. Розроблена прикладна інформаційна технологія оцінки та забезпечення якості проектування.

Програмне забезпечення прикладної інформаційної технології є функціонально достатнім для підтримки процесів якісної розробки у нових проектах створення ПС.

Література

1. Revindasari, Fony. Traceability between business process and software component using Probabilistic Latent Semantic Analysis. Informatics and Computing (ICIC) [Text] / Revindasari Fony, Sarno Riyanarto, Solichah Adhatus // International Conference on. IEEE. – 2016. – P. 245-250.

2. Broy, Manfred. A logical approach to systems engineering artifacts: semantic relationships and dependencies beyond traceability-from requirements to functional and architectural views [Text] / Manfred Broy // Software & Systems Modeling. – 2018. – vol. 17, no. 2. – P. 365-393.

3. ISO/IEC/IEEE Systems and Software Engineering – System Life Cycle Processes / ISO/IEC JTC 1/SC 7 Software and systems engineering. – Geneva, Switzerland : ISO Central Secretariat, 2015. – 84 p.

4. Martínez-Lorente, Angel R. Total Quality Management: Origins and Evolution of the Term [Text] / Angel R. Martínez-Lorente, F. Dewhurst, B. G. Dale // The TQM Magazine. – Bingley, United Kingdom : MCB University Publishers Ltd, 1998. – Vol. 10, № 5. – P. 378-386.

5. MSF for CMMI Process Improvement: Process Guidance [Electronic resource]. – Microsoft Corp., v. 5.0., 2010. – Mode of access: https://lwoodyiii.files.wordpress.com/2010/12/msf-for-cmmi-process-improvement-v5-process-guidance2.doc. – 10.09.2016.

6. OMG Certified UML Professional [Electronic resource]. – Object Management Group, UML Technology Institute Co., Ltd.(UTI), 2014. – Mode of access: http://www.omg.org/ocup-2/index.htm. – 10.09.2018.

7. INTSPEI P-Modeling Framework For Software Companies Whitepaper [Text] / INTSPEI Inc., v. 0.3 Beta. – Lake Success, NY, USA, 2007. – 59 p.

8. Єльчанінов, Д. Б. Технології автоматизації проектування програмних систем [Текст] / Д. Б. Єльчанінов, М. С. Косіло, Н. В. Бєлова // Системи обробки інформації. – 2014. – № 6(122). – С. 135-140.

9. Desfray, Philippe. Modeling enterprise architecture with TOGAF: A practical guide using UML and BPMN [Text] / Philippe Desfray, Gilbert Raymond. – Morgan Kaufmann, San Francisco, CA, USA, 2014. – 350 p.

10. Garlan, David. Software architecture: a travelogue [Text] / David Garlan // Proceedings of the on Future of Software Engineering by ACM. – 2014. – P. 29-39.

11. Федорович, О. Є. Метод багаторівневого компонентного проектування для забезпечення

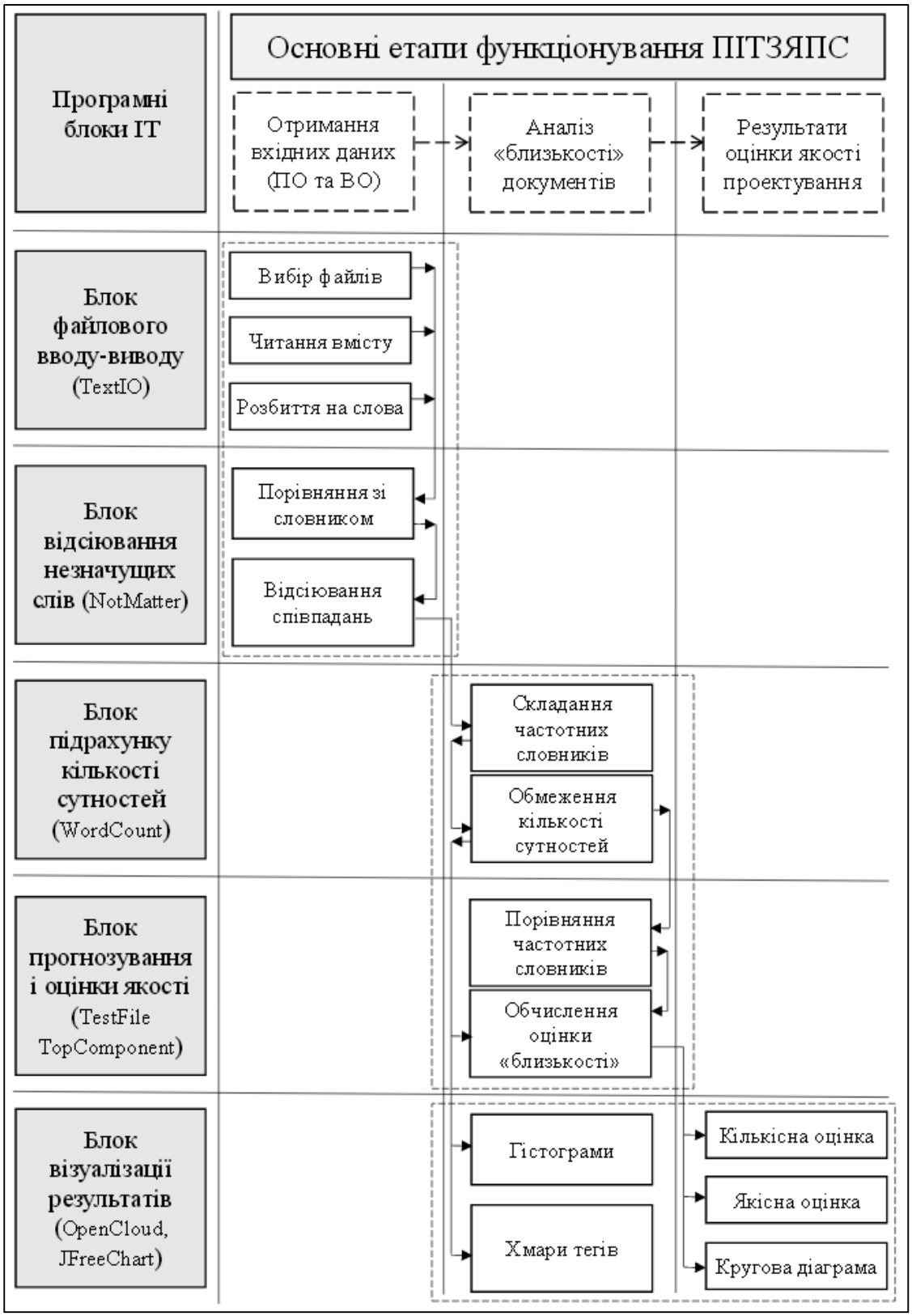

Рис. 2. Структурно-логічна схема ПІТЗЯПС

ISSN 1814-4225 (print).
ISSN 2663-2012 (online)

якості створюваної програмної системи [Текст] / О. Є. Федорович, О. В. Бабич // Радіоелектронні і комп'ютерні системи. – 2016. – № 1(75). – С. 119–124.

12. Бабич, А. В. Управление процессом разработки программной системы с использованием методологи P-Modeling Framework и обратной семантической трассировки [Текст] / А. В. Бабич // Системи озброєння і військова техніка. – 2016. – № 3(47). – С. 112-115.

References

1. Revindasari, Fony., Sarno, Riyanarto., Solichah, Adhatus. Traceability between business process and software component using Probabilistic Latent Semantic Analysis. *Informatics and Computing (ICIC), International Conference on. IEEE,* 2016, pp. 245-250.

2. Manfred, Broy. A logical approach to systems engineering artifacts: semantic relationships and dependencies beyond traceability—from requirements to functional and architectural views. *Software & Systems Modeling*, 2018, vol. 17, no. 2, pp. 365-393.

3. *ISO/IEC/IEEE Systems and Software Engineering – System Life Cycle Processes / ISO/IEC JTC 1/SC 7 Software and systems engineering.* – Geneva, Switzerland: ISO Central Secretariat, 2015. 84 p.

4. Martínez-Lorente Angel R., Dewhurst, F., Dale, B. G. Total Quality Management: Origins and Evolution of the Term. *The TQM Magazine,* Bingley, United Kingdom, MCB University Publishers Ltd, 1998, vol. 10, no. 5, pp. 378-386.

5. *MSF for CMMI Process Improvement: Process Guidance. Microsoft Corp., v. 5.0*, 2010. Available at: https://lwoodyiii.files.wordpress.com/2010/12/msf-for-cmmi-process-improvement-v5-process-guidance2.doc. (accessed 10.09.2018).

6. *OMG Certified UML Professional*. Object Management Group, UML Technology Institute Co., Ltd.(UTI), 2014. Available at: http://www.omg.org/ocup-2/index.htm. (accessed 10.09.2018).

7. *INTSPEI P-Modeling Framework For Software Companies Whitepaper*. INTSPEI Inc., v. 0.3 Beta, Lake Success, NY, USA, 2007. 59 p.

8. Yel′chaninov, D. B. *Tekhnolohiyi avtomatyzatsiyi proektuvannya prohramnykh system* [Technologies of automation of programming of software systems] Systemy OBROBKY ynformatsyy. - 2014. - № 6 (122). - p. 135-140.

9. Desfray, Philippe., Raymond, Gilbert. *Modeling enterprise architecture with TOGAF: A practical guide using UML and BPMN,* San Francisco, CA, USA, 2014. 350 p.

10. Garlan, David. Software architecture: a travelogue. *Proceedings of the on Future of Software Engineering by ACM*, 2014, pp. 29-39.

11. Fedorovich, O. E., Babich, O. V. Metod bahatorivnevoho komponentnoho proektuvannya dlya zabezpechennya yakosti stvoryuvanoyi prohramnoyi systemy [Method of multilevel component design to ensure the quality of created software system]. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2016, no. 1(75), pp. 119-124.

12. Babych, A. V. Upravlenye protsessom razrabotky prohrammnoy systemy s yspol′zovanyem metodolohy R-Modeling Framework y obratnoy semantycheskoy trassyrovky [Management of the process of development of the software system using the methodology of the P-Modeling Framework and the reverse semantic tracing]. *Systemy ozbroyennya i viys'kova tekhnika,* 2016, no.3(47), pp. 112-115.

Поступила в редакцию 12.02.2019, рассмотрена на редколлегии 15.03.2019

ОБЕСПЕЧЕНИЕ КАЧЕСТВА РАЗРАБОТКИ ПРОГРАММНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ КОМПОНЕНТНОГО ПРОЕКТИРОВАНИЯ И ТЕХНОЛОГИИ P-MODELING FRAMEWORK

О. Е. Федорович, Е. В. Бабич, А. В. Бабич

В публикации описаны результаты исследования процесса создания современных программных систем (ПС) с использованием компонентного подхода и технологии проектирования P-Modeling Framework. Обоснована актуальность создания современного метода управления проектом по созданию ПС, который обеспечит качество процесса проектирования и конечного продукта в виде программной системы. Целью роботы является исследование обеспечения качества проектирования программных продуктов и ПС путем разработки прикладной информационной технологии для создания компонентной архитектуры ПС с использованием современной технологии P-Modeling Framework. С помощью системного анализа сформована многоуровневая архитектура ПС. Использованы методы экспертного оценивания для прогнозирования и оценки качества создаваемой ПС. Разработаны модели оптимизации архитектуры ПС. Использованы методы теории прецедентов для формирования множества альтернативных вариантов ПС. Формализован и усовершенствован метод обратной семантической трассировки (RST) для обеспечения

качества процесса проектирования ПС. Описаны структура и особенности прикладной информационной технологии, созданной с использованием P-Modeling Framework. Разработанные модели и методы информационной технологии являются научно-методической основой для решения задач обеспечения качества в процессе создания программных продуктов и систем с использованием позитивного опыта прошлых проектов в виде компонент повторного использования.

Ключевые слова: качество проектирования; компонентное проектирование; системная модель компонентного проектирования; итеративное проектирование; обратная семантическая трассировка.

PROVIDING QUALITY OF SOFTWARE SYSTEMS DEVELOPMENT USING COMPONENT DESIGN AND P-MODELING FRAMEWORK TECHNOLOGY

O. E. Fedorovich, O. V. Babich, A. V. Babich

This publication describes the results of the study of creating modern software systems (SS) process using the component approach and P-Modeling Framework design technology. The urgency of a modern method of project management for the creation of a SS, which will ensure the quality of the design process and the final product in the form of a software system, was substantiated. The aim of the work is to study the quality assurance of designing software products and SS by developing an application information technology for the development of a component architecture of SS using the modern the P-Modeling Framework technology. With the help of system analysis, a multi-level architecture of the SS has been formed. Expert estimation methods have been used to predict and evaluate the quality of the created SS. Models of optimization of the architecture of the SS are developed. Methods of the theory of precedents have been used to form a plurality of alternate variants of the subsystem. Formalized and improved the method of reverse semantic tracing (RST) to ensure the quality of the design process of SS. The structure and features of the applied information technology created using the P-Modeling Framework are described. The developed models and methods of information technology are a scientific and methodological basis for solving the problems of quality assurance in the process of creating software products and systems using the positive experience of past projects as reusable components.

Keywords: design quality; component design; system model of component design; iterative design; reverse semantic traceability.

Федорович Олег Євгенович – д-р техн. наук, проф., зав. каф. Комп'ютерних наук та інформаційних технологій, Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут», Харків, Україна.

Бабич Олена Володимирівна – викладач дисциплін спеціальності "Розробка програмного забезпечення", Полтавський політехнічний коледж Національного технічного університету "Харківський політехнічний інститут", Полтава, Україна.

Бабич Олександр Вікторович – канд. техн. наук, голова цикл. комісії Дисциплін програмної інженерії, викладач дисциплін спеціальності "Розробка програмного забезпечення", Полтавський політехнічний коледж Національного технічного університету "Харківський політехнічний інститут", Полтава, Україна.

Fedorovich Oleg Yevgenyevich – Doctor of technical sciences, professor, head of department of Computer Science and Information Technologies, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine, e-mail: o.fedorovych@khai.edu, ORCID Author ID: 0000-0001-7883-1144.

Babich Olena Volodymyrivna – Teacher of the specialty "Software Development", Poltava Polytechnic College, National Technical University "Kharkiv Polytechnic Institute", Poltava, Ukraine, e-mail: olena.likemouse@gmail.com, ORCID Author ID: 0000-0002-5714-9696.

Babich Alexander Victorovich – Candidate of technical sciences (PhD), head of Software engineering department, teacher of disciplines of the specialty "Software development", Poltava Polytechnic College of National Technical University "Kharkov Polytechnic Institute", Poltava, Ukraine, e-mail: alexander.taurus@gmail.com, ORCID Author ID: 0000-0002-8202-1053.