

КОВАЛЕНКО Олена

Вінницький національний технічний університет

<https://orcid.org/0000-0003-2864-9058>

e-mail: ok@vntu.edu.ua

РОМАНЮК Олександр

Вінницький національний технічний університет

<https://orcid.org/0000-0002-2245-3364>

e-mail: romanyuk@vntu.edu.ua

БАБЮК Наталія

Вінницький національний технічний університет

<https://orcid.org/0000-0003-0607-6340>

babiuk@vntu.edu.ua

МЕТОДОЛОГІЯ УПРАВЛІННЯ НАУКОВО-ДОСЛІДНИМИ ТА ПРАКТИЧНИМИ ІННОВАЦІЙНИМИ ПРОЄКТАМИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ: КЕЙС ДОСЛІДЖЕННЯ ПРОДУКТИВНОСТІ ВИСОКОНАВАНТАЖЕНИХ РОЗПОДІЛЕНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

В статті розглядається актуальна проблема управління проєктами програмної інженерії. Виконано порівняння методологій управління науково-дослідних та інноваційних практичних проєктів. Розглянуті сучасні методології управління проєктами для підвищення продуктивності високонавантажених розподілених інформаційних систем. Дослідження виконані для підвищення продуктивності освітніх електронних інформаційних систем під час високого навантаження (сесій), а також для формування системи управління показниками підприємств «зеленої економіки». Стаття містить детальний опис кейсу для аналізу продуктивності освітньої інформаційної екосистеми в період високих навантажень. Запропоновано мінікейси тестування модулю автоматичної перевірки знань здобувачів. Адаптація гнучких методологій практичних проєктів для наукових досліджень дозволить збільшити продуктивність роботи команд. Приклад запровадження системи моніторингу продуктивності освітньої інформаційної екосистеми в період високих навантажень дозволяє сформувати аналогічні кейси для інших напрямів досліджень продуктивності високонавантажених розподілених систем.

Ключові слова: програмна інженерія, життєвий цикл IT-проєкту, методологія управління проєктами, науково-дослідний проєкт, інноваційний практичний проєкт, високонавантажені інформаційні системи, розподілені інформаційні системи

KOVALENKO Olena, ROMANYUK Oleksandr, BABIUK Natalia

Vinnitsia National Technical University

METHODOLOGY OF MANAGEMENT OF RESEARCH AND PRACTICAL INNOVATIVE PROJECTS IN SOFTWARE ENGINEERING: CASE RESEARCH ON THE PRODUCTIVITY OF HIGHLY LOADED DISTRIBUTED INFORMATION SYSTEMS

The article considers the current problem of software engineering project management. A comparison of management methodologies for research and innovation practical projects is made. Modern project management methodologies are considered to increase the productivity of highly loaded distributed information systems. The research was carried out to increase the productivity of educational electronic information systems during high load (sessions), as well as to form a system for managing the indicators of "green economy" enterprises. The results of the implementation of the proposed comprehensive methodology are presented by the information ecosystems SEL (Vinnitsia National Trade and Economics Institute of KNUTE) and JetIQ (Vinnitsia National Technical University). The use of flexible methodology for research projects makes it possible to quickly test hypotheses, change the direction of research, or add necessary steps to the project. Each sprint allows you to integrate new ideas and adjust the next steps of the research. The main focus of the research project is the increase in knowledge. The article contains a description of the case of the project "Research of the results of high loads" on the example of a web-based system for creating an educational electronic information environment JetIQ VNTU.

The results of the research project on determining critical moments of the JetIQ VNTU information ecosystem during periods of high load allow you to improve the server system and the organization of the use of microservices of the electronic university. The obtained practical and theoretical results are the basis for further improvement of each module separately and the entire system as a whole. The article proposes a conceptual and practical methodology for managing research and/or innovative practical projects based on identifying the main approaches to implementing software engineering projects, agile management methodologies, and experience in developing and implementing software products as results of IT project implementation. A case study of a project to determine the performance indicators of an educational electronic information environment during periods of high load is considered.

Prospects for further research are related to the use of the theory and practice of IT project management to create productive educational electronic information environments for educational institutions, corporate and individual distance and blended learning.

The article contains a detailed description of a case for analyzing the productivity of the educational information ecosystem during periods of high loads. Mini-cases for testing the module for automatic verification of applicants' knowledge are proposed.

Adaptation of flexible methodologies of practical projects for scientific research will allow to increase the productivity of teams. An example of implementing a system for monitoring the productivity of the educational information ecosystem during periods of high loads allows to form similar cases for other areas of research on the productivity of highly loaded distributed systems.

Keywords: software engineering, IT project life cycle, project management methodology, research project, innovative practical project, high-load information systems, distributed information systems.

Стаття надійшла до редакції / Received 01.08.2025

Прийнята до друку / Accepted 24.08.2025

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Проблема запровадження сучасних методологій управління науково-дослідними проектами програмної інженерії є актуальною в зв'язку з необхідністю реалізовувати науково-дослідні проекти, які відповідають потребам розвитку галузі інформаційних технологій, а також дозволяють покращити продуктивність управління для створення та запровадження програмного забезпечення інформаційних систем різного призначення [1]. Кожен проект має свої особливості. Продуктивність програмного забезпечення – один з комплексних показників, за яким здійснюється моніторинг його якості, особливо в умовах високого навантаження. Для складних систем високого навантаження доцільно використовувати мікросервісну архітектуру [2, 3]. Різноманітні моделі управління проектами не повністю покривають процеси наукових та інноваційних практичних досліджень, не завжди орієнтуються на фільтрацію отриманих результатів – від теорії до практики і реального впровадження [4,5]. Гармонійне поєднання процесів та результатів досліджень з практичним впровадженням і є тим необхідним сполученням моделей управління для підвищення продуктивності роботи ІТ-команд.

АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Низка різноманітних статей щодо управління проектами дозволяє використати запропоновані систематизації та визначення. Міжнародні стандарти для проектів створення програмного забезпечення визначають проект та його етапи відповідно до ініціації, визначення вимог, розробки, тестування та впровадження. Такий стандарт як SWEBOOK (Software Engineering Body of Knowledge) містить 18 напрямів знань. В останній версії додано області знань архітектури, операцій та безпеки програмного забезпечення. Стандарт управління проектами також змінився відповідно до нових тенденцій – від процесів до принципів гнучкого управління. Науково-дослідний проект може бути спрямований на дослідження теоретичних засад визначеного об'єкту предметної області [6]. Але для програмної інженерії як прикладної науки, управління науково-дослідним проектом означає дослідження та подальше визначення наукової новизни з можливістю практичного запровадження повного або часткового результату досліджень. Проекти з моніторингу продуктивності програмних продуктів створюються та реалізуються, як правило, на другому етапі формування програмного забезпечення [7]. Автори дослідження мають досвід аналізу вимог для високонавантажених систем, реалізації різних проектів створення програмного забезпечення. Дослідження архітектури, основних алгоритмів в умовах високого навантаження на етапі створення прототипів дає можливість одразу визначити слабкі місця та особливості використання програмного забезпечення.

ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ

Метою роботи є розробка методології управління науково-дослідними проектами програмної інженерії для підвищення продуктивності програмного забезпечення в умовах високого навантаження та роботи команд дослідників і підтримки впроваджених ІТ-проектів.

ВИКЛАД ОСНОВНИХ РЕЗУЛЬТАТІВ

Методологія управління науково-дослідними проектами включає в себе основні визначення, адаптацію стандартів, моделі управління та отримання й використання результатів.

Науково-дослідним проектом програмної інженерії будемо вважати проект, в якому створюють унікальний програмний продукт з використанням сучасних технологій дослідження та розробки з подальшим використанням результатів проекту в практичній діяльності. Наукові дослідження як частина таких проектів базуються на нових ідеях та гіпотезах, які перевіряються під час виконання проекту та реалізуються у вигляді програмних продуктів.

Для формування методології управління науково-дослідним проектом необхідно поєднати два напрями, що містять протиріччя – наукові дослідження з накопиченням та аналізом різноманітних ідей та гіпотез і чіткий проект реалізації та подальшого впровадження. Але навіть міжнародні стандарти щодо інженерії програмного забезпечення SWEBOOK та управління проектами PMBOK (Project Management Body of Knowledge) доповнили свої описи процесів та принципів з врахуванням нових технічних та управлінських тенденцій. SWEBOOK можна розглядати як технічний орієнтир, тобто технічні процеси, методи та знання, які

необхідні інженерам. PMBOK орієнтується на визначення та моніторинг управлінських процесів, які відповідають на питання "що" і "коли" робити для досягнення цілей проекту (управління обсягом, часом, вартістю, ризиками, закупівлями тощо).

В таблиці 1 представлено сфери напрямів та методологія інтеграції.

Таблиця 1

Методологія інтеграції управління проєктом програмної інженерії та його технологіями і науковими дослідженнями

РМВОК Управління	SWEBOK	Управління науково-дослідним проєктом	Інтеграція
обсягом	Вимоги	Обсягом та вимогами, потребами	Обсяг, зміни, вимоги
якістю	Якість та результати тестування	Результати та показники якості	Науково-дослідні та практичні результати з відповідними метриками якості
розкладом	Інженерні процеси та побудова програмного забезпечення	Розклад досліджень	Розклад виконання досліджень з врахуванням практичних процесів програмної реалізації та використання сучасних технологій
людськими ресурсами	Інженерний менеджмент та практика управління людськими ресурсами	Формування та управління науково-дослідними командами	Інтеграція роботи науково-дослідних та практичних команд для гармонійного поєднання
ризики	Безпека програмного забезпечення та інші ризики	Управління ризиками наукових досліджень	Комплексне поєднання управління ризики управління, ризиками наукових досліджень та використання практичних технологій
конфігураціями	Управління конфігураціями	Управління над удосконаленнями отриманих результатів	Контроль версій та удосконалень отриманих результатів
командами	Інженерний менеджмент та професійні практики	Управління командою як гармонійного поєднання теоретиків та практиків	Гнучкі методології управління проєктом
Ідеями, гіпотезами, результатами досліджень	Вимоги	Управління розвитком контексту та результатів досліджень в проєкті	

Для науково-дослідних та інноваційних проєктів необхідно додати напрям управління ідеями, гіпотезами проєкту, а також окремий напрям управління командами науково-дослідного проєкту.

Аналіз останніх версій інтеграції стандартів дозволяє визначити області, які перетинаються та особливі сфери для кожного напрямку.

На практиці необхідно визначити основні етапи життєвого циклу, враховуючи два стандарти та особливості предметної області науково-дослідного проєкту. Так, PMBOK допомагає визначити фази проєкту (Ініціація, Планування, Виконання, Моніторинг, Завершення). SWEBOK наповнює фазу "Виконання" (Execution) технічними інженерними практиками (вимоги, дизайн, кодування, тестування).

Гнучкі методології (Agile) є складовими останніх версій згаданих стандартів. Для науково-дослідного проєкту всі дванадцять базових принципів та нова структура для виконання можуть бути адаптовані відповідно до предметної області проєкту. Загальне планування враховує зацікавлені сторони, команду, життєвий цикл, планування, виконання, постачання, вимірювання результатів, врахування ризиків та невизначеності.

Всі відомі методології гнучкого управління були створені для практичних проєктів. Але вони можуть бути адаптовані до використання в науково-дослідних роботах. В таблиці 2 представлені особливості такої адаптації.

Таблиця 2

Особливості адаптації реалізації гнучкої методології управління проєктами

Реалізація гнучкої методології	Адаптація до науково-дослідних проєктів
Scrum	Найпопулярніший вибір. Фокусується на коротких спринтах (наприклад, 2-4 тижні), які завершуються створенням прототипу для перевірки гіпотези, знань/прототипом. Ідеально підходить для швидкої перевірки гіпотез.
Kanban	Підходить для проєктів, де потік завдань є непередбачуваним (наприклад, несподівані результати експериментів, які вимагають термінового аналізу). Допомагає візуалізувати й обмежувати кількість роботи "в процесі".
Scrum-of-Scrums	Ефективний, коли НДР-проєкт має кілька незалежних дослідницьких команд або напрямків, які потрібно синхронізувати (наприклад, розробка обладнання, розробка ПЗ та аналіз даних).

Використання гнучкої методології для науково-дослідних проєктів дає можливість швидко перевірити гіпотези, змінити напрям досліджень або додати необхідні кроки до проєкту. Кожен спринт дозволяє інтегрувати нові ідеї і скоригувати наступні кроки дослідження. Головний фокус науково-

дослідного проєкту – приріст знань. Розглянемо планування проєкту «Дослідження результатів високих навантажень» на прикладі веборієнтованої системи для створення освітнього електронного інформаційного середовища JetIQ VNTU.

Кейс реалізації такого проєкту складається з наступних кроків.

1. Визначення періодів високого навантаження – початок навчального року, сесія, період перездач, період внесення (експорту-імпорту масових даних), масова робота з модулями тестування, електронного деканату.

2. Аналіз роботи серверів. Визначення апаратних та програмних серверів, їх навантаження, резервування.

3. Аналіз отриманих результатів роботи в періоди високого навантаження. Помилки, пов'язані з людським фактором. Помилки, пов'язані з технічними проблемами. Аналіз серверної архітектури. Пропозиції змін.

4. Формування дашборду аналізу навантажень та показників якості роботи системи JetQ.

5. Отримання результатів роботи з дашбордом від адміністратора системи, групи підтримки серверної архітектури, розробників.

Загальна математична модель для визначення показників в періоди високого навантаження може бути представлена може бути представлена як перелік множин T_{el} , що відповідають кейсам перевірки роботи кожного окремого модуля та всієї системи за контурними функціями (функціонал (Tf), комунікації (Tk), мотиваційно-емоційні елементи (Tme)), UI (Tui) та UX (Tux) дизайн-тестування; тестування супроводження системи (Ts). [7]

$$T_{el} = \{Tf; Tk; Tme; Tui; Tux; Ts\} \quad (1)$$

Одним з модулів високого навантаження в період поточного та підсумкового контролю є модуль «TestIQ».

Кожен кейс розбивається на тести, які повинні покрити кожну з визначених множин моделі.

Для валідації та візуалізації результатів перевірки роботи системи в період високого навантаження використовують удосконалену матрицю відповідності вимог (Traceability Matrix) [8]. Така матриця дозволяє визначити наявність перетину множин вимог та результатів.

Математична модель такої матриці може бути представлена як множина вимог V , множина мінікейсів для тестування T_k .

$$V = \{V_1 \dots V_n\}; \quad T_k = \{T_{k1} \dots T_{km}\};$$

Перетин цих множин $P_{\frac{vi}{kj}}$ дозволяє зробити висновок, чи існує достатня умова для матриці валідності, коли:

$$P_{\frac{vi}{kj}} = 1.$$

В таблиці 3 представлені мінікейси та їх відповідність цілям системи тестування. Такі мінікейси можуть бути створені для кожного модулю і окремо для комунікацій модулів між собою та з користувачем. Особливу увагу в період високих навантажень необхідно приділяти передачі та збереженню даних, зменшенню ризиків неправильного введення даних як користувачами, так і при роботі системи.

Кожен зазначений мінікейс може бути представлений у вигляді окремих бізнес-процесів (освітніх, комунікативних, передачі, збереження тощо), в яких чітко визначається власник та вхідні, вихідні дані, формули перетворення даних та їх обробка.

Таблиця 3

Мінікейси тестування високого навантаження в періоди поточного та підсумкового контролю

Цілі	Навантаження	Мінікейси
Тестування знань здобувачів в період поточного контролю (самостійна робота_	Середній рівень навантаження - нецентралізоване тестування знань здобувачів	Створення тестів
		Додавання в навігатор
		Надання доступу
		Визначення обмежень
		Отримання результатів
		Моніторинг синхронної роботи
		Клонування
Тестування знань здобувачів в період колоквиумів, заліків, іспитів	Високий рівень навантаження, централізоване тестування знань здобувачів	Журнал активності та гейміфікація
		Створення тестів
		Додавання в навігатор
		Надання доступу
		Визначення обмежень
		Отримання результатів
		Моніторинг синхронної роботи
		Клонування
		Журнал активності та гейміфікація
		Відомості підсумкового тестування
		Процедура імпорту в електронний журнал
		Підключення тестового контролю

Результатами науково-дослідного проекту щодо визначення критичних моментів роботи інформаційної екосистеми JetIQ VNTU в періоди високого навантаження дозволяють удосконалити серверну систему та організацію використання мікросервісів електронного університету. Отримані практичні та теоретичні результати є основою для подальшого удосконалення кожного модулю окремо та всієї системи в цілому.

ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

У статті запропоновано концептуальну та практичну методологію управління науково-дослідними та/або інноваційними практичними проектами на основі визначення основних підходів до реалізації проектів програмної інженерії, методологій гнучкого управління та досвіду розробки та запровадження програмних продуктів як результатів реалізації IT-проектів. Розглянуто кейс проекту визначення показників роботи освітнього електронного інформаційного середовища в періоди високого навантаження.

Перспективи подальших досліджень пов'язані з використанням теорії та практики управління IT-проектами для створення продуктивного освітнього електронного інформаційного середовища для закладів освіти, корпоративного та індивідуального дистанційного та змішаного навчання.

Література

1. Єпіфанова І. Ю. Місце процесу управління проектами в стратегії розвитку підприємств / І. Ю. Єпіфанова, В. В. Джеджула // *Innovation and Sustainability*. – 2022. – № 3. – С. 8-13. <http://ir.lib.vntu.edu.ua/handle/123456789/35893>
2. Newman S. *Building Microservices: Designing Fine-Grained Systems*. 2nd Edition. O'Reilly, 2022. 618 p.
3. Waseem Muhammad, Liang Peng Marquez, Gaston Di Salle Amleto. *Testing Microservices Architecture-Based Applications / V/ Waseem, P. Liang, G. Amleto // A Systematic Mapping Study*. DOI: 10.1109/APSEC51365.2020.00020.
4. Близинова І. О., Семко С. Г., Кійко С. Г. Огляд сучасних методологій управління командами IT-проектів. *Управління розвитком складних систем*. Київ, 2020. № 43. С. 60 – 66. <https://doi.org/10.32347/2412-9933.2020.43.60-66>.
5. Вавіленкова А. І. Аналіз гнучких методологій розробки програмного забезпечення для реалізації у командних проектах. *Вісник Національного технічного університету «ХПІ»*. Харків, 2021. № 1(7). С. 39 – 46. <https://doi.org/10.20998/2413-4295.2021.01.06>
6. Agile manifesto: <https://agilemanifesto.org/iso/uk/manifesto.html>.
7. Palamarchuk Y., Kovalenko O., Pylypenko D. Testing Module Models for the 'Electronic University' Information Ecosystem International Scientific and Technical Conference on Computer Sciences and Information Technologies Conference Paper2024 DOI: 10.1109/CSIT65290.2024.10982641
8. Visualizing testing and coverage.” Online courses from the company QATestLab | Main page. Accessed: Aug. 22, 2024. [Online]. Available: <https://training.qatestlab.com/blog/technical-articles/visualization-of-testing-and-test-coverage/>

References

1. Yepifanova I. YU. Mistse protsesu upravlinnya proektamy v stratehiyi rozvytku pidpryyemstv *Innovation and Sustainability*. – 2022. – № 3. – С. 8-13. <http://ir.lib.vntu.edu.ua/handle/123456789/35893>
2. Newman S. *Building Microservices: Designing Fine-Grained Systems*. 2nd Edition. O'Reilly, 2022. 618 p.
3. Waseem Muhammad, Liang Peng Marquez, Gaston Di Salle Amleto. *Testing Microservices Architecture-Based Applications. A Systematic Mapping Study*. DOI: 10.1109/APSEC51365.2020.00020.
4. Blyznyukova I. O., Semko S. H., Kiyko S. H. Ohlyad suchasnykh metodolohiy upravlinnya komandamy IT-proyektiv. *Upravlinnya rozvytkom skladnykh system. Management of complex systems development*. Kyiv, 2020. № 43. С. 60 – 66. <https://doi.org/10.32347/2412-9933.2020.43.60-66>.
5. Vavlenkova A. I. Analiz hnuchkykh metodolohiy rozrobky prohramnoho zabezpechennya dlya realizatsiyi u komandnykh proyektakh. *Visnyk Natsional'noho tekhnichnoho universytetu «KHPI»*. Kharkiv, 2021. № 1(7). С. 39 – 46. <https://doi.org/10.20998/2413-4295.2021.01.0>
6. Agile manifesto: <https://agilemanifesto.org/iso/uk/manifesto.html>.
7. Palamarchuk Y., Kovalenko O., Pylypenko D. Testing Module Models for the 'Electronic University' Information Ecosystem International Scientific and Technical Conference on Computer Sciences and Information Technologies Conference Paper2024 DOI: 10.1109/CSIT65290.2024.10982641
8. Visualizing testing and coverage.” Online courses from the company QATestLab | Main page. Accessed: Aug. 22, 2024. [Online]. Available: <https://training.qatestlab.com/blog/technical-articles/visualization-of-testing-and-test-coverage/>