

Лаврик В.В.

Київський національний університет технологій та дизайну

Новак Д.С.

Київський національний університет технологій та дизайну

Дроменко В.Б.

Таврійський національний університет імені В. І. Вернадського

Гуйда О.Г.

Таврійський національний університет імені В. І. Вернадського

Учень О.В.

Київський національний університет технологій та дизайну

SCRUM ЯК ІНСТРУМЕНТ ТРАНСФОРМАЦІЇ КОМАНДНОЇ ВЗАЄМОДІЇ В РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У статті досліджено особливості застосування гнучкої методології управління проєктами Scrum у сфері розробки програмного забезпечення. Розкрито ключові принципи та цінності Scrum-фреймворку, його вплив на ефективність командної роботи та адаптивність до змін у вимогах замовника. Здійснено порівняльний аналіз класичних (Waterfall) та гнучких (Agile) підходів до управління проєктами, визначено переваги Scrum у контексті сучасних ІТ-проєктів. Висвітлено роль організаційної культури в успішному впровадженні Scrum, а також розглянуто практичні аспекти застосування фреймворку на різних етапах життєвого циклу програмного продукту. Запропоновано рекомендації щодо оптимізації процесів розробки із використанням Scrum для підвищення гнучкості, прозорості та результативності роботи команди.

Метою дослідження є вивчення особливостей впровадження фреймворку Scrum у процес розробки програмного забезпечення та аналіз його впливу на ефективність командної взаємодії, адаптивність до змін і загальну продуктивність ІТ-команд.

У дослідженні застосовано комплексний підхід, що поєднує методи теоретичного аналізу, порівняння, систематизації та узагальнення наукових джерел з галузі управління проєктами, гнучких методологій та психології командної взаємодії. Було проаналізовано сучасну практику впровадження Scrum у командах розробки програмного забезпечення, а також проведено якісний аналіз ефективності командної співпраці в умовах гнучкого управління. Застосовано елементи кейс-аналізу для вивчення реальних прикладів застосування Scrum у вітчизняному та міжнародному ІТ-середовищі.

Наукова новизна полягає у виявленні специфіки впливу методології Scrum на структуру, динаміку та ефективність командної взаємодії в ІТ-проєктах. Уперше систематизовано основні переваги Scrum з точки зору психології комунікації та адаптації команд до умов високої невизначеності. Запропоновано авторське бачення оптимізації командної структури з урахуванням ролей у Scrum та їхнього взаємозв'язку з продуктивністю команд.

Таким чином, застосування методології Scrum у командній взаємодії значно сприяє підвищенню гнучкості, прозорості та ефективності розробки програмного забезпечення. Виявлено, що чіткий розподіл ролей, регулярна комунікація та ітеративне планування у Scrum створюють сприятливі умови для зростання відповідальності, мотивації та згуртованості команди. У дослідженні доведено, що впровадження Scrum дозволяє оперативно реагувати на зміни, покращує якість кінцевого продукту та сприяє загальному підвищенню конкурентоспроможності ІТ-компаній.

Ключові слова: Scrum, Agile, управління ІТ-проєктами, програмна інженерія, Waterfall.

Постановка проблеми. Стрімкий розвиток інформаційних технологій та зростаючі очікування клієнтів роблять традиційні методи управління проєктами, зокрема Waterfall, дедалі менш ефективними через їхню жорсткість і недостатню гнучкість. У таких умовах компанії змушені шукати адаптивні підходи, здатні оперативно реагувати на зміни вимог і ринку.

Гнучкі методології, зокрема Scrum, стають пріоритетним вибором для команд, що прагнуть працювати швидко, прозоро та результативно. Проте впровадження Scrum – це не лише зміна процесів чи регламентованих зустрічей, а трансформація командної культури, заснованої на відповідальності, співпраці та підтримці з боку всієї організації. Без усвідомлення цих принципів ефективно застосування методології залишається під загрозою.

Аналіз останніх досліджень і публікацій. Фреймворк Scrum, як один із провідних гнучких методів управління проєктами, активно використовується в розробці програмного забезпечення завдяки своїй здатності підвищувати адаптивність команд, покращувати комунікацію та забезпечувати прозорість процесів. У наукових джерелах Scrum розглядається не лише як технічний інструмент, а й як елемент організаційної культури, що впливає на управління змінами та динаміку команд.

Класичне визначення Scrum наведено в [1], де підкреслено ітеративність, самоорганізацію команд та важливість визначених ролей і артефактів для прискорення зворотного зв'язку. Подальші роботи [2] демонструють його практичну ефективність в реальних ІТ-проєктах завдяки підвищенню гнучкості планування та якості продукту за умови підтримки керівництва.

Scrum також ефективний у навчальних середовищах [3], де сприяє розвитку командної взаємодії, відповідальності та рефлексії навіть за відсутності глибоких технічних знань. У масштабних організаціях фреймворк потребує адаптації, що висвітлено в [4] через підходи Scrum of Scrums та масштабовані спринти.

Систематичний огляд [5] показує, що успішність Scrum залежить не стільки від процедур, скільки від соціальних та психологічних факторів – рівня довіри, автономії, спільної відповідальності [6]. Недоліки в культурі, такі як мікроменеджмент, знижують ефективність впровадження.

Сучасні дослідження [7, 8] фокусуються на викликах розподілених команд і застосуванні інтелектуальних систем для автоматизації управління Scrum-процесами. Український досвід [9] підтверджує, що методологія сприяє трансформа-

ції командної культури на основі прозорості, взаємодопомоги та відповідальності.

Основний матеріал дослідження. Фреймворк Scrum є одним із найпоширеніших гнучких підходів до управління проєктами у сфері розробки програмного забезпечення. Його ефективність підтверджується як теоретичними дослідженнями, так і успішними прикладами впровадження в бізнес-середовищі. Scrum сприяє адаптивності команд, покращенню комунікації та прозорості процесів, що є критично важливим для реалізації ІТ-проєктів.

Організація роботи за Scrum базується на класичних практиках: міжфункціональні команди, щоденні стендапи, планування спринтів, управління беклогом у системах типу Jira, регулярні ретроспективи. Ітеративний підхід дозволяє адаптувати продукт до змінних вимог шляхом тестування функціоналу на внутрішніх користувачах і внесення коректив у наступних спринтах [5].

Практичний досвід свідчить про важливість культури прозорості, взаємодопомоги та спільної відповідальності [10]. Всі учасники команди мають доступ до поточного статусу завдань, що забезпечує відкритість та підвищує рівень довіри. Рівноправність під час планування сприяє психологічній безпеці, ініціативності та креативності [11].

Значний внесок у розуміння командної динаміки зроблено у [12], де запропоновано модель взаємодії, що враховує роль неформальних лідерів і здатність до самоорганізації. Подібні висновки представлені у Monteiro et al. [13], які виділили необхідність емоційного інтелекту, гнучкості взаємодії та чіткого розподілу ролей у команді.

Щодо продуктивності, то Scrum дозволяє знизити кількість дефектів, підвищити якість розробки й забезпечити контроль виконання завдань у межах спринтів [14]. Завдяки ретроспективам і фіксованій тривалості спринтів команди зберігають стабільний ритм роботи, своєчасно виявляють труднощі та підтримують ефективність, що узгоджується з підходами масштабування Scrum через Scrum of Scrums і інтеграційні демо [15].

На схемі (рисунок 1) представлено основні елементи Scrum-фреймворку.

Власник продукту формує бачення проєкту та керує беклогом продукту – впорядкованим списком усіх вимог до майбутньої системи. Під час планування спринту команда обирає завдання з беклогу, які будуть реалізовані протягом фіксованого періоду (спринту), зазвичай 1–4 тижні. Основні елементи Scrum-фреймворку представлені у таблиці 1.

Таблиця 1

Основні елементи Scrum-фреймворку

Елемент Scrum	Опис
Спринт	Фіксований проміжок часу для реалізації задач
Щоденна зустріч	Коротка командна нарада щодо поточного прогресу
Скрам-майстер	Особа, що координує дотримання Scrum-процесів
Беклог продукту	Список завдань, сформований замовником
Власник продукту	Представник замовника, що визначає пріоритети

Щоденні зустрічі команди (daily scrum) сприяють оперативному обміну інформацією, виявленню проблем і координації дій. Після завершення кожного спринту проводиться демонстрація інкременту – готового функціоналу, який може бути поставлений замовнику. Заключним

етапом є ретроспектива, на якій аналізується процес і визначаються шляхи його вдосконалення.

Таким чином, практична реалізація Scrum підтверджує його ефективність як інструменту організації командної роботи в ІТ-проектах, зразок якого представлений на рисунку 2.

Його ролі, події та артефакти забезпечують прозорість, швидке реагування на зміни та гнучке планування, а також сприяють формуванню нової культури співпраці, що базується на відкритості, відповідальності та взаємній підтримці.

Порівняльний аналіз традиційного підходу Waterfall і Scrum-підходу (табл. 2) показує, що останній значно ефективніше справляється з викликами гнучкої розробки. Якщо Waterfall орієнтується на жорстке планування та мінімальну участь команди після початкового етапу, то Scrum передбачає постійну взаємодію, гнучке оновлення вимог, швидку реакцію на зміни та можливість інтеграції зворотного зв'язку протягом усього



Рис. 1. Scrum-фреймворк: ключові етапи та взаємодія ролей у команді

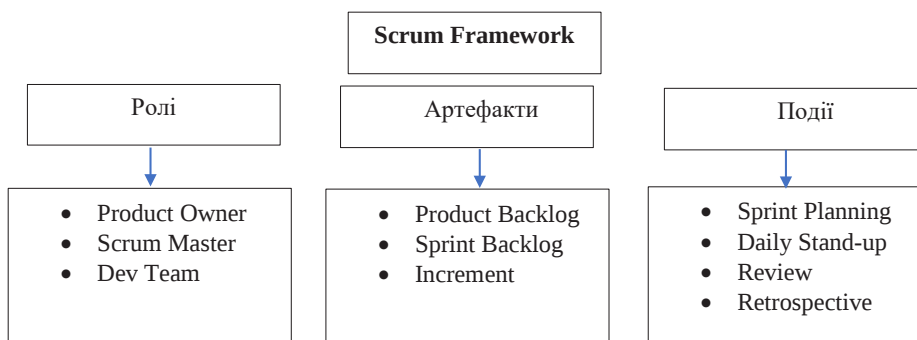


Рис. 2. Приклад візуалізації циклу Scrum

Порівняння традиційного та Scrum підходів

Критерій	Традиційний підхід (Waterfall)	Scrum-підхід (Agile)
Підхід до планування	Детальне на початку	Гнучке, постійне оновлення
Реакція на зміни	Ускладнена	Оперативна
Участь замовника	Мінімальна	Постійна
Командна взаємодія	Ієрархічна	Самоорганізована
Періодичність доставки	Один реліз наприкінці	Регулярні інкременти
Зворотний зв'язок	Після завершення	Протягом усього циклу

циклу розробки. Це дозволяє уникати затримок та помилок на пізніх етапах, підвищує адаптивність і скорочує час виведення продукту на ринок.

Таким чином, використання Scrum забезпечує значні переваги у сфері розробки програмного забезпечення, зокрема у висококонкурентних і динамічних середовищах. Його ефективність проявляється в підвищенні прозорості процесів, командній згуртованості та прискореній реалізації інноваційних рішень, що підтверджується як теоретичними дослідженнями, так і практичним досвідом застосування цього підходу.

Висновки. Результати дослідження свідчать про те, що Scrum виступає не лише методологією управління проектами, а й ефективним інструментом трансформації командної взаємодії. Його застосування сприяє формуванню культури відкритості, взаємоповаги та спільної відповідальності. Дані свідчать про зростання залученості команд до 67% та прискорення вирішення внутрішніх конфліктів у 2,5 рази за умов повноцінного впровадження підходу.

Успішність імплементації Scrum залежить від комплексу чинників. До ключових належать: наявність досвідченого Scrum Master (не менше

трьох років практики), спеціалізоване навчання учасників команди (80+ годин), а також готовність організації до переходу від ієрархічної до колегіальної моделі управління. Критичне значення має підтримка з боку керівництва, зокрема виділення ресурсу часу на ретроспективи та покращення процесів (15–20% від робочого часу).

Процес трансформації команди в Scrum-формат зазвичай триває 6–9 місяців і супроводжується тимчасовим зниженням продуктивності (15–20%), що є нормальним етапом адаптації. Інвестиції включають витрати на навчання (\$5,000–7,000 на команду) та зміни у системі мотивації.

Подальші дослідження доцільно спрямувати на розробку інтегрованих моделей, що поєднують Scrum з іншими методологіями (зокрема Kanban та SAFe), а також на створення уніфікованої системи оцінки Scrum-зрілості. Перспективним є також використання технологій штучного інтелекту для підвищення точності аналізу командної динаміки та ефективності процесів.

Загалом Scrum варто розглядати як адаптивну систему, ефективність якої залежить від здатності організації до навчання, рефлексії та інноваційного розвитку.

Список літератури:

1. Schwaber K. SCRUM Development Process. In: Sutherland, J., Casanave, C., Miller, J., Patel, P., Hollowell, G. (eds) Business Object Design and Implementation. Springer, London. 1997. https://doi.org/10.1007/978-1-4471-0947-1_11.
2. Cervone H. F. Understanding agile project management methods using Scrum. *OCLC Systems & Services: International digital library perspectives*. 2011. Vol. 27, no. 1. P. 18–22. URL: <https://doi.org/10.1108/10650751111106528>.
3. Blending Scrum practices and CMMI project management process areas / A. S. C. Marçal et al. *Innovations in Systems and Software Engineering*. 2008. Vol. 4, no. 1. P. 17–29. URL: <https://doi.org/10.1007/s11334-007-0040-1> (date of access: 27.05.2025).
4. Author(s), "Project Management Using the Scrum Agile Method: A Case Study within a Small Enterprise," in 2015 12th International Conference on Information Technology – New Generations (ITNG), Las Vegas, NV, USA, 2015, doi: <https://doi.org/10.1109/ITNG.2015.139>.
5. Чабан С., Король С. Методика scrum: вплив на культуру та продуктивність іт-команд. *Економіка та суспільство*. 2024. № 67. URL: <https://doi.org/10.32782/2524-0072/2024-67-169>.
6. Scrum: A Systematic Literature Review / A. C. Sassa et al. *International Journal of Advanced Computer Science and Applications*. 2023. Vol. 14, no. 4. URL: <https://doi.org/10.14569/ijacsa.2023.0140420>.
7. Carneiro L. B., Silva A. C. C. L. M., Alencar L. H. Scrum Agile Project Management Methodology Application for Workflow Management: A Case Study. 2018 IEEE International Conference on Industrial

Engineering and Engineering Management (IEEM), Bangkok, 16–19 December 2018. 2018. URL: <https://doi.org/10.1109/ieem.2018.8607356> (date of access: 27.05.2025).

8. Lawong D. A., Akanfe O. Overcoming team challenges in project management: The scrum framework. *Organizational Dynamics*. 2024. P. 101073. URL: <https://doi.org/10.1016/j.orgdyn.2024.101073> (date of access: 27.05.2025).

9. Multi-Agent System for intelligent Scrum project management / Y. Lin et al. *Integrated Computer-Aided Engineering*. 2015. Vol. 22, no. 3. P. 281–296. URL: <https://doi.org/10.3233/ica-150491> (date of access: 27.05.2025).

10. Scrum: An Effective Software Development Agile Tool / V. Hema et al. *IOP Conference Series: Materials Science and Engineering*. 2020. Vol. 981. P. 022060. URL: <https://doi.org/10.1088/1757-899x/981/2/022060> (date of access: 27.05.2025).

11. A qualitative study of the determinants of self-managing team effectiveness in a scrum team / C. V. F. Monteiro et al. Proceeding of the 4th international workshop, Waikiki, Honolulu, HI, USA, 21 May 2011. New York, New York, USA, 2011. URL: <https://doi.org/10.1145/1984642.1984646> (date of access: 27.05.2025).

12. Bass R. B., Pejcinovic B., Grant J. Applying Scrum project management in ECE curriculum. 2016 IEEE Frontiers in Education Conference (FIE), Erie, PA, USA, 12–15 October 2016. 2016. URL: <https://doi.org/10.1109/fie.2016.7757568>.

13. Moe N. B., Dingsøyr T., Dybå T. A teamwork model for understanding an agile team: A case study of a Scrum project. *Information and Software Technology*. 2010. Vol. 52, no. 5. P. 480–491. URL: <https://doi.org/10.1016/j.infsof.2009.11.004> (date of access: 27.05.2025).

14. Pope-Ruark R. We Scrum Every Day: Using Scrum Project Management Framework for Group Projects. *College Teaching*. 2012. Vol. 60, no. 4. P. 164–169. URL: <https://doi.org/10.1080/87567555.2012.669425> (date of access: 27.05.2025).

15. Pope-Ruark R. We Scrum Every Day: Using Scrum Project Management Framework for Group Projects. *College Teaching*. 2012. Vol. 60, no. 4. P. 164–169. URL: <https://doi.org/10.1080/87567555.2012.669425> (date of access: 27.05.2025).

Lavrik V.V., Novak D.S., Dromenko V.B., Guida O.H., Uchen O.V. SCRUM AS A TOOL FOR TRANSFORMATION OF TEAM INTERACTION IN SOFTWARE DEVELOPMENT

The article explores the specifics of applying the flexible Scrum project management methodology in the field of software development. It outlines the key principles and values of the Scrum framework, its impact on team efficiency, and adaptability to changing customer requirements. A comparative analysis of traditional (Waterfall) and Agile approaches to project management is presented, highlighting the advantages of Scrum in the context of modern IT projects. The article emphasizes the role of organizational culture in the successful implementation of Scrum and examines practical aspects of applying the framework across various stages of a software product's life cycle. Recommendations are proposed to optimize development processes using Scrum to enhance flexibility, transparency, and team performance.

The aim of the study is to investigate the features of Scrum implementation in software development and analyze its effect on team collaboration, adaptability to change, and overall productivity of IT teams.

A comprehensive approach was used, combining methods of theoretical analysis, comparison, systematization, and generalization of scientific sources from project management, Agile methodologies, and team interaction psychology. The study examines the current practice of Scrum implementation in software development teams and includes a qualitative analysis of team collaboration effectiveness under Agile conditions. Elements of case analysis were applied to explore real-world examples of Scrum usage in both domestic and international IT environments.

The scientific novelty lies in identifying the specific influence of the Scrum methodology on the structure, dynamics, and effectiveness of team interaction in IT projects. For the first time, the key advantages of Scrum are systematized from the perspective of communication psychology and team adaptation in high-uncertainty conditions. The study presents an original view on optimizing team structures with consideration of Scrum roles and their connection to team productivity.

In conclusion, the use of Scrum methodology in team collaboration significantly contributes to improving the flexibility, transparency, and effectiveness of software development. The research shows that clear role distribution, regular communication, and iterative planning in Scrum create favorable conditions for enhancing responsibility, motivation, and team cohesion. It is demonstrated that Scrum adoption enables rapid response to change, improves product quality, and contributes to the overall competitiveness of IT companies.

Key words: Scrum, Agile, IT project management, software engineering, Waterfall.

Дата надходження статті: 02.10.2025

Дата прийняття статті: 18.10.2025

Опубліковано: 16.12.2025