

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ОЛЕСЯ ГОНЧАРА

ЗАТВЕРДЖУЮ

Ректор



Сергій ОКОВИТИЙ

«26» травня 2026 р.

ПОГОДЖЕНО

Проректор з наукової роботи

Олег МАРЕНКОВ

«26» травня 2026 р.

ПРОГРАМА

ВСТУПНОГО ІСПИТУ ІЗ СПЕЦІАЛЬНОСТІ

для здобуття ступеня доктора філософії

на основі освітнього ступеня магістра (освітньо-кваліфікаційного рівня спеціаліста)

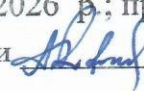
за спеціальністю F7 Комп'ютерна інженерія

освітньо-наукова програма Комп'ютерна інженерія



Розглянуто на засіданні вченої ради
факультету фізики, електроніки та комп'ютерних систем

від «26» травня 2026 р.; протокол № 84

Голова вченої ради  (Коваленко О.В.)

Дніпро-2026

Розробники:

1. Хандецький В.С., доктор технічних наук, професор, гарант освітньо-наукової програми, завідувач кафедри електронних обчислювальних машин Дніпровського національного університету імені Олеся Гончара.
2. Литвинов А.О., кандидат технічних наук, доцент, доцент кафедри електронних обчислювальних машин Дніпровського національного університету імені Олеся Гончара.
3. Гниленко О.Б., кандидат фізико-математичних наук, доцент, доцент кафедри електронних обчислювальних машин Дніпровського національного університету імені Олеся Гончара.
4. Скуратовський І.А., кандидат фізико-математичних наук, доцент кафедри електронних обчислювальних машин Дніпровського національного університету імені Олеся Гончара.

Програма вступного іспиту до аспірантури затверджена на засіданні кафедри електронних обчислювальних машин (протокол № 8 від 05 травня 2026 року)

Завідувач кафедри

електронних обчислювальних машин



Володимир ХАНДЕЦЬКИЙ

“ 06” травня 2026 р.

МЕТА ВСТУПНОГО ІСПИТУ ДО АСПІРАНТУРИ

Метою вступного іспиту до аспірантури є перевірка рівня теоретичних знань і практичних навичок вступників у галузі інформаційних технологій, спеціальність «Комп'ютерна інженерія», можливостей використання знань й застосування навичок у науковому дослідженні та у практичній діяльності, можливостей творчого опрацювання наукової інформації із застосуванням розуміння міждисциплінарного підходу та можливостей творчого мислення щодо пошуку вирішення існуючих проблем у сфері комп'ютерної інженерії. У програмі вступного іспиту до аспірантури відображені основні вимоги до підготовки осіб, що вступають до аспірантури за спеціальністю F7 «Комп'ютерна інженерія».

1 ПЕРЕЛІК ДИСЦИПЛІН, ЗА ЯКИМИ БУДУТЬ АТЕСТУВАТИСЯ СТУДЕНТИ

1. Теорія розробки та реалізації інформаційних систем та IT-інфраструктур.
2. Технології передачі інформації в глобальних комп'ютерних мережах.
3. Технології інтелектуальних та хмарних обчислень.
4. Методи проектування комп'ютерних систем.
5. Технології паралельних на розподілених обчислень.

2 ЗМІСТ РОЗДІЛІВ АТЕСТАЦІЇ

2.1. Теорія розробки та реалізації інформаційних систем та IT-інфраструктур

1. Особливості та роль підходу Domain-Driven Design (DDD) для розробки шару бізнес-логіки в сучасних інформаційних системах.
2. Роль обмежених контекстів (Bounded Contexts) при розробці сучасних інформаційних систем.

3. Особливості застосування патерну CQRS(Command Query Responsibility Segregation). Переваги, обмеження, варіації (напр., комбінація з Event Sourcing).
4. Мікросервісна архітектура. Шаболни. Переваги та недоліки.
5. Асинхронна взаємодія та потокова обробка подій. Порівняльний аналіз моделей доставки повідомлень (*At-least-once*, *At-most-once*, *Exactly-once*) у сучасних брокерах (наприклад, Kafka, RabbitMQ).
6. Докеризація сервісів. Порівняльний аналіз з віртуалізацією на рівні ОС. Переваги та недоліки докеризації.
7. Оптимізація та архітектурні моделі Cloud Computing. Розподіл ресурсів у хмарних інфраструктурах. Порівняльний аналіз архітектурних моделей.
8. Інфраструктура як код (IaC). Декларативного підхід до управління ІТ-інфраструктурою. Явища «дрейфу конфігурації» (*configuration drift*) в хмарних інфраструктурах.
9. Синтез архітектурних паттернів та ІТ-інфраструктури для критичних систем. Підходи. Шаблони. Засоби логування та моніторингу.

2.2. Технології передачі інформації в глобальних комп'ютерних мережах

1. Принципи, методи та засоби проектування, побудови, та обслуговування сучасних комп'ютерних мереж.
2. Методи розрахунку метрики сучасних протоколів маршрутизації, будування доменів та автономних систем глобальних комп'ютерних мереж.
3. Організація і методи передачі інформації в бездротових мережах четвертого і п'ятого поколінь.
4. Методи балансування навантаження в каналах IP та MPLS – мереж , методи організації тунелів в доменах віртуальних приватних мереж.
5. Установлення маршруту в мережах MPLS, протокол розподілення міток LDP, механізми створення і підтримки трактів за допомогою протоколу Constrained-Based LDP.
6. Методи передачі даних між різними автономними системами Інтернет з використанням протоколів зовнішнього шлюзу.
7. Методи будування закритих мереж, маршрутизація в них, методи підвищення швидкості обміну інформацією між закритими мережами і Інтернет.
8. Розвиток технології Carrier Ethernet, організація віртуальних приватних мереж на її основі.

9. Організація віртуальних приватних мереж з використанням протоколу OpenVPN. Впровадженням механізму Data Channel Offload (DCO) для підвищення швидкості роботи.

10. Основні технічні особливості роботи протоколу WireGuard в Інтернет. Механізми безпеки. Метрики продуктивності для різних VPN протоколів.

11. Програмно-конфігуровані мережі (SDN). Аналіз ефективності програмних SDN-контролерів.

12. Мережеві протоколи в SDN-архітектурі. Забезпечення якості обслуговування.

2.3. Технології інтелектуальних та хмарних обчислень

1. Основні етапи розвитку штучного інтелекту, порівняйте символічний та нейромережевий підходи, визначте переваги, недоліки та перспективи їх інтеграції в сучасних системах.

2. Принципи функціонування та навчання сучасних моделей машинного навчання. Розкрийте особливості навчання з учителем, без учителя та з підкріпленням. Порівняйте сфери застосування, вимоги до даних та критерії оцінювання якості моделей.

3. Розкрийте концепцію хмарних обчислень та моделей надання хмарних сервісів. Порівняйте моделі IaaS, PaaS та SaaS, охарактеризуйте їх переваги, недоліки та типові сценарії використання.

4. Принципи побудови розподілених систем у хмарних середовищах. Проблеми масштабованості, відмовостійкості, узгодженості даних та балансування навантаження. Проаналізуйте компроміси між продуктивністю обробки даних та надійністю.

5. Охарактеризуйте сучасні технології контейнеризації та оркестрації застосунків. Поясніть принципи роботи контейнерів, порівняйте їх із віртуальними машинами, розкрийте роль систем оркестрації у забезпеченні масштабованості та безперервності роботи сервісів.

6. Проаналізуйте підходи до обробки великих даних у хмарних середовищах. Розкрийте концепції Data Lake та Data Warehouse. Поясніть особливості пакетної та потокової обробки даних, наведіть приклади.

7. Поясніть особливості проектування та розгортання інтелектуальних сервісів у хмарній інфраструктурі. Розгляньте життєвий цикл моделей машинного навчання, концепції MLOps, автоматизацію навчання, тестування та моніторингу моделей.

8. Проблеми безпеки та конфіденційності в інтелектуальних і хмарних системах. Основні загрози, методи захисту даних, підходи до керування доступом, а також особливості забезпечення безпеки моделей штучного інтелекту.

9. Проаналізуйте сучасні підходи до забезпечення якості, інтерпретованості та довіри до систем штучного інтелекту. Розкрийте

проблеми пояснюваності моделей, оцінювання надійності результатів, виявлення упереджень у даних та алгоритмах.

10. Побудова інтелектуальних агентів та мультиагентних систем. Архітектури агентів, механізми взаємодії між ними, координація та прийняття рішень. Перспективи застосування агентних систем.

2.4. Методи проєктування комп'ютерних систем

1. Методи та технології високорівневого синтезу (High-Level Synthesis) при проєктуванні цифрових систем.
2. Методологія та інструментальні засоби функціональної верифікації та симуляційного тестування комп'ютерних систем.
3. Проєктування та віртуальне прототипування на системному рівні з використанням SystemC.
4. Методологія платформеного проєктування складних систем та технологія повторного використання IP-блоків.
5. Методологія та автоматизовані системи перевірки технологічних норм проєктування (Design Rule Checking) на нанометрових рівнях літографічного процесу.
6. Технології апаратного прототипування та емуляції складних комп'ютерних систем.
7. Етапи наскрізного технологічного маршруту фізичного проєктування (RTL-to-GDSII) цифрових систем.
8. Методологія проєктування апаратних прискорювачів штучного інтелекту на FPGA.
9. Методи та технології проєктування високошвидкісних інтерфейсів передачі даних на FPGA.
10. Методи та технології проєктування та конфігурування підсистем пам'яті на FPGA.

2.5. Технології паралельних та розподілених обчислень

1. Класифікація паралельних комп'ютерних систем. Стисла характеристика кожного з типів систем (напр., SISD, SIMD, MISD, MIMD).
2. Багатозадачність та паралелізм. Особливості та архітектура багатозадачних операційних систем.
3. Конвеєризація обчислень та асинхронна обробка в сучасних комп'ютерних системах.
4. Границі масштабованості та закони паралельних обчислень. Порівняльний аналіз закону Амдала та закону Густафсона-Барсіса.

5. Проблеми синхронізації та взаємного виключення. Механізми та рівні блокування. Особливості неблокуючої синхронізації (Lock-free алгоритми) та приклади їх застосування.
6. Порівняльний аналіз моделі передачі повідомлень (на прикладі моделі акторів) та моделі із загальною пам'яттю (на прикладі стандарту OpenMP).
7. Особливості архітектури SIMT (Single Instruction, Multiple Threads) порівняно з класичною SIMD. Організація ієрархії пам'яті в GPU-обчисленнях (глобальна, розділювана, регістри) та її вплив на проектування паралельних алгоритмів.
8. Концепція Command Query Responsibility Segregation (CQRS) як метод подолання обмежень класичних архітектур у розподілених обчислювальних системах.
9. Особливості обробки транзакцій у розподілених базах даних. Порівняльний аналіз моделей ACID та BASE. Концепція узгодженості в кінцевому рахунку (Eventual Consistency).
10. Обчислювальні кластери. Різновиди (високопродуктивні обчислення, балансування навантаження тощо), віртуалізація, особливості побудови та супроводу.

3. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ СКЛАДАННЯ ВСТУПНОГО ВИПРОБУВАННЯ

Екзаменаційні білети із вступного випробування включають по 5 питань (по одному питанню з кожної дисципліни), що потребують розгорнутої відповіді текстового характеру.

Оцінювання знань здійснюється за 100 – бальною шкалою.

Тривалість вступного випробування – 120 хвилин.

0-59 балів виставляються вступникові в аспірантуру, який робить принципові помилки у відповідях, не може дати обґрунтовані відповіді на питання білету, не орієнтується в основних теоретичних і практичних положеннях дисциплін, демонструє відсутність розуміння навчального матеріалу.

60-63 бали виставляються вступникові в аспірантуру, який демонструє мінімально достатній рівень знань, володіє тільки початковими знаннями, орієнтується у базових поняттях.

64-74 бали виставляються вступникові в аспірантуру, який демонструє знання у задовільному обсязі, володіє лише основними поняттями з комп'ютерної інженерії, не володіє значною частиною програмного матеріалу, допускає істотні помилки у відповідях.

75-81 бал виставляються вступникові в аспірантуру, який знає основний матеріал, але не засвоїв його окремих деталей, допускає недоліки у його практичному застосуванні, відчуває труднощі при інтерпретації окремих

складових предметної дисципліни, обмежений в аналізі та узагальненні явищ.

82-89 балів виставляються вступникові в аспірантуру, який твердо знає програмний матеріал, грамотно і по суті викладає його, не допускає істотних помилок у відповіді на запитання практичного характеру, здійснює грамотні експертні висновки. В відповідях викладає відомості, одержані з підручників, посібників.

90-100 балів виставляються вступникові в аспірантуру, який глибоко та міцно засвоїв програмний матеріал, вичерпано, послідовно, грамотно й логічно його викладає, у відповіді якого тісно пов'язується теорія з практикою. При цьому вступник показує обізнаність з монографічною літературою та інформаційними джерелами в обраній галузі.

При остаточній оцінці результатів виконання завдання враховується здатність вступника:

- застосувати закони, методи, принципи комп'ютерної інженерії у конкретних ситуаціях;
- аналізувати і оцінювати факти, події, тенденції у галузі інформаційних технологій;
- викладати матеріали логічно, послідовно, з відповідним обґрунтуванням.

Шкала оцінювання результатів вступного випробування

СУМА БАЛІВ	ОЦІНКА ЗА НАЦІОНАЛЬНОЮ ШКАЛОЮ
90 – 100	Відмінно / Excellent
82 – 89	Добре / Good
75 – 81	
64 – 74	Задовільно / Satisfactory
60 – 63	
0 – 59	Незадовільно / Fail

4 РЕКОМЕНДОВАНА ЛІТЕРАТУРА

До розділу 2.1 Теорія розробки та реалізації інформаційних систем та IT-інфраструктур

1. Eric Evans 2003 - Domain-Driven Design - Tackling Complexity in the Heart of Software. Addison-Wesley Professional, 2003. 560 p.
2. Vernon, V. (2013). Implementing Domain-Driven Design. Boston: Addison-Wesley. ISBN 978-0-321-83457-7.

1. Martin, R. C. (2017). *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Hong Kong: Pearson Education Asia. ISBN 978-0-13-449416-6.
2. Erl Thomas. *Service-Oriented Architecture: Analysis and Design for Services and Microservices*. 2nd edition. — Prentice Hall, 2016. — 416 p.
3. Newman, Sam. *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media, 2015. p.260
4. Chavan, A. (2025). The role of domain-driven design in successful microservices migration strategies. *Journal of Information Systems Engineering and Management*. <https://www.jisemjournal.com/index.php/journal/article/view/8888>
5. Bass, L., Weber, I., Zhu, L.: *DevOps: A Software Architect's Perspective*. AddisonWesley Professional (2015)
6. L. E. Lwakatare et al., "DevOps in practice: A multiple case study of five companies," *Information and Software Technology*, vol. 114, pp. 217–230, Oct. 2019, doi: 10.1016/j.infsof.2019.06.010.
7. Erder, M.; Pureur, P.; Woods, E. *Continuous Architecture in Practice: Software Architecture in the Age of Agility and DevOps*; AddisonWesley Professional: Boston, MA, USA, 2021. DOI: 10.1016/B978-0-12-803284-8.00007-5
8. Y. Abgaz, A. McCarren, P. Elger, D. Solan, N. Lapuz, M. Bivol, G. Jackson, M. Yilmaz, J. Buckley, P. Clarke, *Decomposition of Monolith Applications Into Microservices Architectures: A Systematic Review*, *IEEE Transactions on Software Engineering* 49, 4213–4242 (2023); doi: 10.1109/TSE.2023.3287297
9. M.S. Rahman et al., "Evaluating the impact of test-driven development on software quality enhancement," *Int. J. Math. Sci. Comput. (IJMSC)*, vol. 10, no. 3, pp. 51–76, Sep. 2024, doi: 10.5815/ijmsc.2024.03.05.
10. M. Wynne and A. Hellesøy, "The Cucumber Book: Behaviour-Driven Development for Testers and Developers," *Pragmatic Bookshelf*, 2012.
11. J. F. Smart, "BDD in Action: Behavior-Driven Development for the Whole Software Lifecycle," *Manning Publications*, 2014.
12. J. Koskela, "BDD in Action: Behavior-Driven Development for the Whole Software Lifecycle," *Manning*, 2014.
13. C. Anderson. *Docker*. *IEEE Software Engineering*, 32(3):102–105, 2015
14. Alonso, J., Piliszek, R., Cankar, M.: *Embracing iac through the devsecops philosophy: Concepts, challenges, and a reference framework*. *IEEE Software* 40(1), 56–62 (2023)
15. Aviv, I., Gafni, R., Sherman, S., Aviv, B., Sterkin, A., Bega, E.: *Infrastructure from code: The next generation of cloud lifecycle automation*. *IEEE Softw.* 40(1), 42–49 (2023)
16. Nigel Poulton. *Docker Deep Dive. Zero to Docker in a single book*. Cathy McGough (Stratford Living Publishing), 2020. p.250.

До розділу 2.2. Технології передачі інформації в глобальних комп'ютерних мережах

1. Хандецький В.С. Об'єднання технологій маршрутизації і комутації в комп'ютерних мережах. Дніпро, «Ліра», 2018, 151 с.
2. Жураковський Б. Ю., Зенів І. О. Комп'ютерні мережі: навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2020. 336 с.
3. Коробейнікова Т. І., Захарченко С. М. Комп'ютерні мережі : Навч. посіб. Львів : Львів. політехніка, 2022. 228 с.
4. Хандецький В.С., Мазур А.В. Полухін Г.В. Навчально-методичний посібник з дисципліни «Технології глобальних мереж». Технологія Software-defined networking (SDN). Дніпро: Ліра. 2025 р., 72 с.
<https://repository.dnu.dp.ua/document-details/5289>
5. Tanenbaum A., Wetheral D. Computer Networks, Fifth edition. New Jersey, Pearson, Prentice Hall, 2018, 962 с.
6. Kurose J., Ross K. Computer Networking. A Top-Down Approach. Eight edition. Pearson, 2020, 731 p.
7. Хандецький В.С., Герасимов В.В. Карпенко Н.В. Performance analysis of wireless computer networks in conditions of high interference intensity. Radio Electronics, Computer Science, Control. 2023, No.3(66). P. 148-159. (Web of Sciences) <http://ric.zntu.edu.ua/article/view/288146>
8. Хандецький В.С., Полухін Г.В. Application of the slow start mechanism in the noisy channels of IEEE 802.11 wireless computer networks. Information Technology: Computer Science, Software Engineering and Cyber Security, No.1 (2025), p. 211- 217. <https://doi.org/10.32782/IT/2025-1-28>.
9. Хандецький В.С., Герасимов В.В. Карпенко Н.В. Evaluating the efficiency of mechanisms for frame block transmissions in noisy channels of IEEE802.11 networks Radio Electronics, Computer Science, Control.- 2025.-No.2.- P. 6-16. (Web of Sciences) <https://doi.org/10.15588/1607-3274-2025-2-1>
10. Netwave, «Software-defined networking (SDN): визначення й особливості програмно-визначених мереж,» <https://netwave.ua/blog/software-defined-networking-sdn-viznachennya-j-osoblivosti-programno-viznachenih-merezh>
11. Ryu SDN Framework Community, «Ryu Controller Documentation: OpenFlow Protocol,» 2023. https://ryu.readthedocs.io/en/latest/ofproto_v1_3_ref.html.
12. Open Networking Foundation, «ONOS Official Documentatio,» 2024.
<https://opennetworking.org/onos/>.

До розділу 2.3 Технології інтелектуальних та хмарних обчислень

1. Vijay Janapa Reddi. Machine Learning Systems. The physics of AI engineering. Harvard University, MIT Press 2026. <https://mlsysbook.ai/>
2. Chip Huyen, Designing Machine Learning Systems. O'Reilly Media, 2022.
https://github.com/chiphuyen/dmls-book?utm_source=chatgpt.com

3. Cloud Computing Technology (Open Access, 2024)
<https://archive.org/details/oapen-20.500.12657-89923>
4. Artificial Intelligence Index Report 2025
https://aiindex.stanford.edu/report/?utm_source=chatgpt.com
5. Гордієнко Ю.Г., Таран В.І. Хмарні обчислення (КІП, 2025)
<https://comsys.kpi.ua/wp-content/uploads/2025/02/konspekt-lekciy-1.pdf>
6. Грід-системи та технології хмарних обчислень. Навчальний посібник для здобувачів освітнього рівня «магістр» спеціальностей: F3 «Комп'ютерні науки», F6 «Інформаційні системи та технології» / Г. В. Шимчук, О. Б. Назаревич, Я. В., Литвиненко, В. А. Готович, В. В. Никитюк., І. О. Боднарчук – Тернопіль : ФОП Паляниця В. А., 2025. – 396 с
https://elartu.tntu.edu.ua/bitstream/lib/49891/8/Posibnyk_HRID-STEMY_2025.pdf

До розділу 2.4 Методи проєктування комп'ютерних систем

1. Лахно В.А., Гусєв Б.С., Смолій В.В., Місюра М.Д., Касаткін Д.Ю. Технології проєктування комп'ютерних систем. – К.: НУБіП України, 2019. 205 с.
2. Проєктування комп'ютерних систем на основі мікросхем програмованої логіки : монографія / С. А. Іванець, Ю. О. Зубань, В. В. Казимир, В. В. Литвинов. – Суми : Сумський державний університет, 2013. – 313 с.
3. Тонкошкур О.С., Морозов О.С. Основи технології проєктування цифрових пристроїв у середовищі Active-HDL. Практикум: навчальний посібник. – Дніпро: Акцент ПП, 2021, 116 с.
4. Проєктування цифрових систем з використанням мови VHDL: Навч. посібник/ В.В. Семенець, І.В. Хаханова, В.І. Хаханов.— Харків: ХНУРЕ, 2003, 92 с.
5. Kumar A. Fundamentals of digital circuits. – PHI Publisher, 2010, 613 p.
6. Stephen Brown, Zvonko Vranesic. Fundamentals of Digital Logic with VHDL Design. – McGraw-Hill, 2020, 420 p.
7. Bruno F. FPGA programming for beginners. - Packt Publishing Ltd., 2021, 369 p.
8. Baker, R. J. CMOS : circuit design, layout, and simulation / Jake Baker. – 3rd ed., Wiley, 2010, 1214 p.
9. Pong P. Chu. FPGA Prototyping by Verilog Examples. – Wiley, 2018, 332 p.
10. W. Wolf. FPGA-Based System Design. – Prentice Hall, 2004, 246 p.

До розділу 2.5 Технології паралельних та розподілених обчислень

1. Tanenbaum A. S., Van Steen M. Distributed Systems: Principles and Paradigms (Розподілені системи: принципи та парадигми). 4th ed. CreateSpace Independent Publishing Platform, 2023. 590 p.

2. Hennessy J. L., Patterson D. A. Computer Architecture: A Quantitative Approach. 6th ed. Morgan Kaufmann, 2019. 928 p.
3. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media, 2017. 590 p.
4. Andrews G. R. Foundations of Multithreaded, Parallel, and Distributed Programming. Addison-Wesley, 2000. 640 p.
5. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar. Introduction to Parallel Computing, Addison Wesley, 2003, 856 p.
6. Литвинов О.А., Хандецький В.С. Паралельні та розподілені обчислення в комп'ютерних системах: Дніпро: «Ліра», 2021. С.172.
7. Литвинов О.А., Хандецький В.С. Розподілена обробка інформації:[моногр.]. Днепропетровськ: ТОВ «Баланс-клуб», 2013. С.314.
8. Литвинов О.А., Мартинович Л.Я., Пономарьов І.В. Технологія паралельної обробки інформації: OpenMP та MPI. - Дніпро, ДНУ, 2017, 123с.
9. Литвинов О. А., Грузін Д. Л. Інженерія програмного забезпечення на базі архітектури CQRS та Event Sourcing орієнтованої на снєпшоти. KPI Science News. 2026. № 1. С. 60–74. DOI: <https://doi.org/10.20535/kpissn.2026.1.350992>.
10. Литвинов О. А., Грузін Д. Л. Порівняння застосовності методів оцінки архітектури програмного забезпечення в контексті CQRS та варіацій архітектури Event Sourcing. Radio Electronics, Computer Science, Control. 2026. № 1. С. 103–120. (Web of Science). DOI: <https://doi.org/10.15588/1607-3274-2026-1-10>.
11. Vogels Werner. Eventually consistent. Communications of the ACM, vol. 52, no. 1, pp. 40-44, 2009. DOI: [10.1145/1466443.1466448](https://doi.org/10.1145/1466443.1466448)
12. Gomes Victor B.F. et al. Verifying strong eventual consistency in distributed systems. Proceedings of the ACM on Programming Languages, v. 1, n. OOPSLA, p. 1-28, 2017. DOI: [10.1145/3133933](https://doi.org/10.1145/3133933)
13. Burckhardt Sebastian et al. Principles of eventual consistency. Foundations and Trends in Programming Languages, v. 1, n. 1-2, p. 1-150, 2014.