МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ
Декан факультету
_____ Інна КОНДЮС
«26» серпня 2026 р.

ЗАТВЕРДЖУЮ
Перший проректор
_____ Надія КОВАЛЬЧУК
«26» серпня 2026 р.

СИЛАБУС

з освітнього компонента «Методи забезпечення якості програмного забезпечення»
ступінь вищої освіти – доктор філософії
галузь знань – F Інформаційні технології
спеціальність – F2 Інженерія програмного забезпечення
освітня програма – Інженерія програмного забезпечення

Кафедра інженерії програмного забезпечення

Силабус склав(ла) завідувач кафедри комп'ютерних наук, к.т.н., доцент Валерій ЛІЩИНА

Силабус затверджений на засіданні кафедри інженерії програмного забезпечення, протокол № 1 від «26» серпня 2026 р.

Завідувач кафедри к.т.н., доцент Наталія ЛІЩИНА

Розглянуто та схвалено групою забезпечення освітньої програми «Інженерія програмного забезпечення», протокол № 1 від «26» серпня 2026 р.

Гарант ОП «Інженерія програмного забезпечення» к.т.н., доцент Валерій ЛІЩИНА

Схвалено науково-методичною радою університету, протокол № 1 від «26» серпня 2026 р.

Затверджений силабус онлайн:
https://e.lntu.edu.ua/syllabus/1908

Луцьк – 2026

СИЛАБУС ОСВІТНЬОГО КОМПОНЕНТА «Методи забезпечення якості програмного забезпечення»

Освітня програма	Інженерія програмного забезпечення
Галузь знань / Спеціальність	F Інформаційні технології / F2 Інженерія програмного забезпечення
Рівень вищої освіти	третій (освітньо-науковий)
Статус дисципліни	нормативна
Мова викладання	українська
Обсяг	5 кредитів ECTS / 150 годин, семестр 1
Форма контролю	екзамен
Викладач(і)	к.т.н., доцент Валерій ЛІЩИНА електронна пошта: v.lishchyna@lntu.edu.ua

1. Мета і завдання дисципліни

Мета: Формування у здобувачів ступеня доктора філософії здатності науково обґрунтовувати, розробляти, удосконалювати та емпірично оцінювати методи забезпечення якості програмного забезпечення впродовж життєвого циклу, з урахуванням стандартів, сучасних практик, особливостей інтелектуальних і розподілених систем та вимог відтворюваності досліджень.

Завдання: - сформувати критичне розуміння сучасних моделей якості, стандартів і методів операціоналізації характеристик якості;
- навчити розробляти валідні плани вимірювання та емпіричного оцінювання методів забезпечення якості;
- опанувати сучасні методи статичного й динамічного аналізу, тестування, верифікації, DevSecOps та експлуатаційного контролю якості;
- сформувати здатність оцінювати якість інтелектуальних, data-intensive, хмарних і розподілених систем в умовах невизначеності;
- забезпечити розроблення або вдосконалення методу забезпечення якості як наукового результату з відтворюваною експериментальною перевіркою.

Пререквізити: Базові знання з життєвого циклу програмного забезпечення, архітектури, якості та тестування, статистики й програмування, набуті на попередніх рівнях освіти.

2. Компетентності

ІК — інтегральна компетентність: Здатність продукувати нові ідеї, розв’язувати комплексні проблеми професійної та/або дослідницько-інноваційної діяльності у сфері інженерії програмного забезпечення та з дотичних до неї міждисциплінарних напрямах, застосовувати методологію наукової та педагогічної діяльності, проводити власне наукове дослідження, результати якого мають наукову новизну, теоретичне та практичне значення.

СК02 — Здатність виявляти, ставити та вирішувати проблеми дослідницького характеру в сфері інженерії програмного забезпечення, оцінювати та забезпечувати якість виконуваних досліджень.

СК03 — Здатність отримувати нові наукові результати, які створюють нові знання та становлять оригінальний внесок у розвиток інженерії програмного забезпечення та дотичних до неї міждисциплінарних напрямів.

СК04 — Здатність відстежувати тенденції розвитку інженерії програмного забезпечення та критично переосмислювати наявні технології.

СК05 — Здатність до розроблення нових та вдосконалення існуючих моделей, методів, засобів, процесів у сфері інженерії програмного забезпечення, які забезпечують розвиток або надають нові можливості технологіям розробки та супроводження програмного забезпечення.

СК06 — Здатність до застосування сучасних методологій, методів та інструментів інженерії програмного забезпечення в науково-педагогічній та науковій діяльності.

3. Структура дисципліни, розподіл годин, зв'язок тем з компетентностями

№	Назва теми	Лекції	Лаб.	Самост. робота	Разом	Методи навчання	Методи оцінювання	ЗК та СК
Змістовий модуль 1. Моделі, вимірювання та емпіричне оцінювання якості програмного забезпечення								
1	Тема 1. Якість програмного забезпечення як об’єкт наукового дослідження.	2	4	12	18	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, захист лабораторної роботи	СК02, СК04, СК06
2	Тема 2. Метрики та методи вимірювання якості програмного забезпечення.	2	4	12	18	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, захист лабораторної роботи	СК02, СК06
3	Тема 3. Емпіричне оцінювання методів забезпечення якості.	2	4	12	18	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, захист лабораторної роботи	СК02, СК06, СК03
4	Тема 4. Статичний аналіз, архітектурна якість і технічний борг.	2	4	12	18	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, МО8, захист лабораторної роботи	СК04, СК06
Змістовий модуль 2. Сучасні методи забезпечення якості та їх експериментальна перевірка.								
5	Тема 5. Сучасні методи тестування та оцінювання ефективності тестів.	2	4	15	21	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, МО8, захист лабораторної роботи	СК02, СК04, СК06
6	Тема 6. Безперервне забезпечення якості, DevSecOps та експлуатаційний контроль.	2	4	14	20	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, МО8, захист лабораторної роботи	СК04, СК06

№	Назва теми	Лекції	Лаб.	Самост. робота	Разом	Методи навчання	Методи оцінювання	ЗК та СК
7	Тема 7. Особливості оцінювання якості інтелектуальних і розподілених систем.	1	2	14	17	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, МО8, захист лабораторної роботи	СК02, СК04, СК06
8	Тема 8. Розроблення та емпірична перевірка вдосконалення методу забезпечення якості.	2	4	14	20	МН1, МН2, МН3, МН6	МО1, МО2, МО4, МО7, МО8, захист лабораторної роботи	СК02, СК05, СК06, СК03
	Разом	**15**	**30**	**105**	**150**			

Легенда: ЗК — загальна компетентність (номер за п. 2); СК — спеціальна (фахова) компетентність (номер за п. 2). Повне формулювання кожного коду наводиться в ОП / стандарті спеціальності.

Методи навчання: МН1 — словесний метод (лекція, дискусія, співбесіда тощо); МН2 — практичний метод (лабораторні та практичні заняття); МН3 — наочний метод (метод ілюстрацій і метод демонстрацій); МН6 — самостійна робота (розв'язання завдань).

Методи оцінювання: МО1 — екзамен; МО2 — усне або письмове опитування; МО4 — тестування; МО7 — презентації результатів виконаних завдань та досліджень; МО8 — презентація та виступи на наукових заходах.

4. Результати навчання

ПРН01. Мати передові концептуальні та методологічні знання з інженерії програмного забезпечення та дотичних до неї міждисциплінарних напрямів, а також дослідницькі навички, достатні для проведення наукових і прикладних досліджень на рівні останніх світових досягнень з відповідного напряму, отримання нових знань та/або здійснення інновацій.

ПРН02. Планувати і виконувати експериментальні та/або теоретичні дослідження з інженерії програмного забезпечення та дотичних міждисциплінарних напрямів з використанням сучасних інструментів та дотриманням норм академічної і професійної етики, критично аналізувати результати власних досліджень і результати інших дослідників у контексті усього комплексу сучасних знань щодо досліджуваної проблеми.

ПРН03. Пропонувати нові ефективні методи і моделі розроблення, впровадження, супроводу та забезпечення якості програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.

ПРН05. Застосовувати сучасні інструменти і технології пошуку, оброблення та аналізу інформації, зокрема, статистичні методи аналізу даних великого обсягу та/або складної структури, спеціалізовані бази даних та інформаційні системи для покращення ефективності програмних систем.

ПРН06. Формулювати і перевіряти гіпотези; використовувати для обґрунтування висновків належні докази, зокрема, результати теоретичного аналізу, експериментальних досліджень і математичного та/або комп'ютерного моделювання, наявні літературні дані.

ПРН09. Формулювати та вирішувати задачі оптимізації, адаптації, прогнозування, керування та прийняття рішень щодо процесів, засобів та ресурсів розробки, впровадження, супроводу та експлуатації програмного забезпечення.

ПРН10. Аналізувати та оцінювати стан і перспективи розвитку інженерії програмного забезпечення та інформаційних технологій у цілому.

5. Цілі сталого розвитку

ЦСР 9 – Промисловість, інновації та інфраструктура

6. Неформальна освіта

Рекомендовані курси, тренінги, стажування та інші: результати навчання, здобуті у формальній, неформальній чи інформальній освіті, можуть бути визнані відповідно до процедур ЛНТУ. Підтвердженнями можуть бути сертифікати тематичних тренінгів, вебінарів чи онлайн-курсів разом з описом досягнутих результатів навчання та, за потреби, коротке верифікаційне завдання або портфоліо.

7. Критерії та форми контролю

Оцінювання — за 100-бальною шкалою (національна шкала / ЄКТС). Підсумковий бал = поточний контроль (30 %) + модульний контроль (30 %) + екзамен (40 %).

Академічна доброчесність: самостійне виконання завдань; коректне посилання на джерела; позначення частин тексту, згенерованих засобами ШІ. Порушення академічної доброчесності не толеруються.

8. Зміст тем

Тема 1. Якість програмного забезпечення як об’єкт наукового дослідження.

Поняття якості програмного забезпечення та його розвиток. Моделі якості продукту й якості під час використання: ISO/IEC 25010:2023, ISO/IEC 25019:2023 та ISO/IEC 25002:2024. Залежність вимог до якості від контексту використання і потреб зацікавлених сторін. Зв’язок характеристик якості, вимог, метрик, критеріїв прийнятності та доказів їх виконання.

Тема 2. Метрики та методи вимірювання якості програмного забезпечення.

Цілі вимірювання та підхід Goal–Question–Metric. Структурні, процесні, експлуатаційні й користувацькі метрики. Шкали вимірювання, надійність і валідність показників. Нормування, агрегування та встановлення порогових значень. Дублювання метрик, вплив сторонніх чинників і ризики хибної інтерпретації результатів.

Тема 3. Емпіричне оцінювання методів забезпечення якості.

Дослідницькі питання та гіпотези. Експеримент, дослідження випадку й аналіз програмних репозиторіїв. Вибір об’єктів, базового методу порівняння та контрольованих умов. Планування вибірки й повторних вимірювань. Статистична та практична значущість, розмір ефекту. Загрози валідності, відтворюваність і склад реплікаційного пакета.

Тема 4. Статичний аналіз, архітектурна якість і технічний борг.

Автоматизоване оцінювання якості коду та ISO/IEC 5055:2021. Складність, зв’язність, залежності, дублювання коду та ознаки недоліків проєктування. Архітектурні порушення й технічний борг. Перевірка результатів статичного аналізу, хибні спрацьовування. Зіставлення метрик із дефектами та змінами. Вибір пріоритетів усунення проблем за обмежених ресурсів.

Тема 5. Сучасні методи тестування та оцінювання ефективності тестів.

Ризик-орієнтоване, модельно-орієнтоване, property-based і metamorphic testing. Проблема тестового оракула. Мутаційне тестування як засіб оцінювання та вдосконалення тестових наборів. Покриття, виявлення дефектів і витрати ресурсів. Умови коректного експериментального порівняння підходів. Місце контрактного тестування, fuzzing і формальної верифікації в забезпеченні якості.

Тема 6. Безперервне забезпечення якості, DevSecOps та експлуатаційний контроль.

Перевірки якості в CI/CD: автоматизовані тести, статичний аналіз, перевірки залежностей і захищеності. Критерії прийняття та блокування змін — quality gates. Контроль якості під час розроблення й експлуатації. Телеметрія, показники та цілі рівня обслуговування — SLI/SLO. Використання OWASP ASVS для визначення вимог безпеки та окремих практик OWASP SAMM для оцінювання процесів.

Тема 7. Особливості оцінювання якості інтелектуальних і розподілених систем.

Якість даних і моделей, робастність, справедливість, пояснюваність та дрейф у системах машинного навчання. Узгодженість даних, затримка, доступність і відмовостійкість розподілених систем. Вибір характеристик якості відповідно до сценарію використання. Оцінювання варіативності результатів і меж застосовності висновків на прикладі одного обраного класу систем.

Тема 8. Розроблення та емпірична перевірка вдосконалення методу забезпечення якості.

Формулювання проблеми та обґрунтування запропонованої зміни методу. Вибір базового підходу, даних і критеріїв порівняння. Порівняльний експеримент, аналіз чутливості або внеску окремого компонента. Інтерпретація позитивних, негативних і невизначених результатів. Обмеження, загрози валідності, відтворюваність та оформлення наукового звіту.

9. Опис ІЗ/КПІЗ

Не передбачено навчальним планом.

10. Самостійна робота

Самостійна робота здобувача є одним із способів активного, цілеспрямованого набуття нових для нього знань та умінь; є основою його підготовки як фахівця, забезпечує набуття прийомів пізнавальної діяльності, інтерес до творчої роботи, вирішення наукових та практичних завдань.

Основні види самостійної роботи здобувача: опрацювання навчального матеріалу, пошук інформації в бібліотеках, мережі Інтернет, використання баз даних інформаційно-пошукових та довідникових систем; підготовка до виконання занять, опрацювання інструкцій і методичних рекомендацій; підготовка доповідей, рефератів, звітів; виконання індивідуальних завдань; самооцінювання знань і умінь; творча робота здобувача (наукової публікації, тез доповідей, есе) тощо.

Питання та завдання самостійної роботи формуються на початку викладання курсу залежно від мети наукових досліджень здобувача та його індивідуального навчального плану.

Самостійна робота здобувача є формою його індивідуальної позааудиторної роботи, метою якого є самостійне засвоєння частини програмного матеріалу, систематизація, поглиблення, узагальнення, закріплення та практичне застосування знань, розвиток навичок самостійної науково-дослідної роботи. Самостійна робота здобувача є складовою навчальної програми курсу, яка виконується на перевірку знань, умінь та навичок, отриманих під час лекційних і лабораторних занять, охоплює декілька тем та (або) зміст освітньої компоненти в цілому.

Самостійну роботу здобувач може виконувати у бібліотеці, навчальних кабінетах, лабораторіях, комп’ютерних класах тощо. Виконання здобувачем самостійної роботи супроводжується отриманням консультацій викладача, наукового керівника, гаранта ОНП. Навчальний матеріал для самостійної роботи виноситься на поточний і підсумковий контроль разом з навчальним матеріалом, який опрацьовувався під час аудиторних занять.

11. Загальні вимоги

Освітній процес регулюється академічними правилами ЛНТУ. Здобувачі освіти зобов’язані ознайомитися з вимогами, визначеними Кодексом академічної доброчесності ЛНТУ та Політикою етичного використання штучного інтелекту в ЛНТУ, і дотримуватися принципів академічної доброчесності та етичного використання штучного інтелекту в освітній діяльності.

Академічна доброчесність. Учасники освітнього процесу зобов’язані дотримуватися Закону України «Про академічну доброчесність» від 18.12.2025 № 4742-IX та внутрішніх нормативних документів ЛНТУ. Не допускаються академічний плагіат, фабрикація, фальсифікація, привласнення авторства, несамостійне виконання завдань, а також недоброчесне використання результатів, згенерованих штучним інтелектом.

Використання штучного інтелекту. Інструменти штучного інтелекту можуть використовуватися для пошуку ідей, пояснення методів, попереднього структурування, мовного редагування та допоміжного аналізу неконфіденційних даних, якщо конкретне завдання не передбачає повністю самостійного виконання. Здобувач освіти має зазначати використаний інструмент, мету та етап його застосування, перевіряти факти й розрахунки та самостійно обґрунтовувати висновки.

12. Література

I. Навчально-методичні матеріали та розробки

Праці автора силабуса:

1. Методи забезпечення якості програмного забезпечення: конспект лекцій для здобувачів третього (освітньо-наукового) рівня вищої освіти освітньо-наукової програми «Інженерія програмного забезпечення» галузі знань F Інформаційні технології спеціальності F2 Інженерія програмного забезпечення / уклад.: В. О. Ліщина. Луцьк: ЛНТУ, 2026. 28 с.

2. Методи забезпечення якості програмного забезпечення: методичні вказівки до лабораторних робіт для здобувачів третього (освітньо-наукового) рівня вищої освіти освітньо-наукової програми «Інженерія програмного забезпечення» галузі знань F Інформаційні технології спеціальності F2 Інженерія програмного забезпечення / уклад.: В. О. Ліщина. Луцьк: ЛНТУ, 2026. 28 с.

Праці інших авторів:

3. Трофименко О. Г., Дика А. І. Тестування та забезпечення якості програмних систем: навчальний посібник. Одеса: Фенікс, 2024. 195 с. DOI: https://doi.org/10.32837/11300.27717.

4. Bierig R., Brown S., Galván E., Timoney J. Essentials of Software Testing. Cambridge: Cambridge University Press, 2021. 318 p. DOI: https://doi.org/10.1017/9781108974073.

5. Aniche M. Effective Software Testing: A Developer’s Guide. Manning Publications, 2022. URL: https://www.manning.com/books/effective-software-testing.

6. Okken B. Python Testing with pytest: Simple, Rapid, Effective, and Scalable. 2nd ed. The Pragmatic Bookshelf, 2022. 272 p. URL: https://pragprog.com/titles/bopytest2/python-testing-with-pytest-second-edition/.

7. Leloudas P. Introduction to Software Testing: A Practical Guide to Testing, Design, Automation, and Execution. Apress, 2023. DOI: https://doi.org/10.1007/978-1-4842-9514-4.

8. Wohlin C., Runeson P., Höst M., Ohlsson M. C., Regnell B., Wesslén A. Experimentation in Software Engineering. 2nd ed. Berlin, Heidelberg: Springer, 2024. XXV, 274 p. DOI: https://doi.org/10.1007/978-3-662-69306-3.

II. Наукові праці

Праці автора силабуса:

9. Остапчук О. В., Суринович О. М., Ліщина Н. М., Ліщина В. О. Аналіз ефективності методів сучасної реалізації шаблонів проєктування в об’єктно-орієнтованому програмуванні. Herald of Khmelnytskyi National University. Technical Sciences. 2023. № 329(6). С. 281–285. https://doi.org/10.31891/2307-5732-2023-329-6-281-285.

Праці інших авторів:

10. Kashiwa Y., Nishikawa R., Kamei Y., Kondo M., Shihab E., Sato R., Ubayashi N. An empirical study on self-admitted technical debt in modern code review. Information and Software Technology. 2022. Vol. 146. Article 106855. DOI: https://doi.org/10.1016/j.infsof.2022.106855.

11. Cai Y., Kazman R. Software design analysis and technical debt management based on design rule theory. Information and Software Technology. 2023. Vol. 164. Article 107322. DOI: 10.1016/j.infsof.2023.107322.

12. Ajibode A., Apedo Y., Ajibode T. Systematic literature review on forecasting and prediction of technical debt evolution. Препринт. arXiv, 2024. DOI: https://doi.org/10.48550/arXiv.2406.12026.

13. Romina Eramo, Michele Tucci, Daniele Di Pompeo, Vittorio Cortellessa, Antinisca Di Marco,

Davide Taibi, Architectural support for software performance in continuous software engineering: A systematic mapping study, Journal of Systems and Software, Volume 207, 2024, 111833, ISSN 0164-1212, https://doi.org/10.1016/j.jss.2023.111833.

14. Hasselbring W. Benchmarking as Empirical Standard in Software Engineering Research. Proceedings of the 25th International Conference on Evaluation and Assessment in Software Engineering (EASE 2021). New York: Association for Computing Machinery, 2021. P. 365–372. DOI: 10.1145/3463274.3463361.

15. Li M., Chen T., Yao X. How to Evaluate Solutions in Pareto-Based Search-Based Software Engineering: A Critical Review and Methodological Guidance. IEEE Transactions on Software Engineering. 2022. Vol. 48, No. 5. P. 1771–1799. DOI: 10.1109/TSE.2020.3036108.

16. Gordieiev O., Rainer A., Kharchenko V., Pishchukhina O., Gordieieva D. A Unified Approach to the Development of Technology-Based Software Quality Models on the Example of Blockchain Systems. IEEE Access. 2024. Vol. 12. P. 118875–118889. DOI: 10.1109/ACCESS.2024.3448271.

17. Gordieiev O., Gordieieva D., Rainer A. Software Quality Assessment: Defect Life Cycle, Software Defect Profile, Its Types and Misalignments. Software Quality: Higher Software Quality through Zero Waste Development. LNBIP. 2023. Vol. 472. P. 109–120. https://link.springer.com/chapter/10.1007/978-3-031-31488-9_6.

18. Gordieiev O., Kharchenko V., Gordieieva D. Software Requirements Profile Quality Model. International Journal of Computing. 2022. Vol. 21(1). P. 111–119. https://computingonline.net/computing/article/view/2524.

19. Gordieiev O., Gordieieva D., Kharchenko V., Kondius I., Brezhniev I. Extended Model of Software Quality Assessment Scenario: Concept, Operations, Application. CEUR Workshop Proceedings. 2022. Vol. 3171. P. 1629–1638. https://ceur-ws.org/Vol-3171/paper116.pdf.

20. Zhang J. M., Harman M., Ma L., Liu Y. Machine Learning Testing: Survey, Landscapes and Horizons. IEEE Transactions on Software Engineering. 2022. Vol. 48. P. 1–36. DOI: 10.1109/TSE.2019.2962027.

21. Gordieiev O., Gordieieva D., Rainer A., Gorbenko A., Tarasyuk O. Quality Assessment of Artificial Intelligence Systems: A Metric-Based Approach. Electronics. 2026. Vol. 15, No. 3. Article 691. DOI: https://doi.org/10.3390/electronics15030691.

III. Інформаційні ресурси

22. ISO/IEC 25010:2023. Systems and software engineering – SQuaRE – Product quality model. https://www.iso.org/standard/78176.html

23. ISO/IEC 25040:2024. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality evaluation framework. 2024. URL: https://www.iso.org/standard/83467.html.

24. ISO/IEC 25019:2023. Systems and software engineering – SQuaRE – Quality-in-use model. https://www.iso.org/standard/78177.html

25. ISO/IEC 5055:2021. Information technology – Software measurement – Software quality measurement –Automated source code quality measures. URL: https://www.iso.org/standard/80623.html.

26. ISO/IEC/IEEE 29119-1:2022. Software and systems engineering – Software testing – Part 1: General concepts. https://www.iso.org/standard/81291.html

27. ISO/IEC/IEEE 29119-4:2021. Software and systems engineering – Software testing – Part 4: Test techniques. https://www.iso.org/standard/79430.html

28. OWASP Foundation. Application Security Verification Standard (ASVS). Version 5.0.0. 2025. URL: https://github.com/OWASP/ASVS/releases/tag/v5.0.0.

29. OWASP Software Assurance Maturity Model (SAMM), Version 2.0. 2020. https://github.com/OWASP/samm/releases.

30. ISO/IEC TR 29119-11:2020. Software and systems engineering – Software testing – Part 11: Guidelines on the testing of AI-based systems. https://www.iso.org/standard/79016.html.

31. ISO/IEC 25002:2024. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality model overview and usage. 2024. 17 p. URL: https://www.iso.org/standard/78175.html.

32. ISO/IEC/IEEE 29119: офіційний огляд серії стандартів тестування. URL: https://committee.iso.org/sites/jtc1sc7/home/projects/flagship-standards/isoiecieee-29119-series.html.

33. ISO/IEC TR 7052:2023. Controlling frequently occurring risks during development and maintenance of custom software. https://www.iso.org/standard/84422.html