

**ЗАКЛАД ВИЩОЇ ОСВІТИ
«УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА»**

Факультет суспільних та прикладних наук

Кафедра інформаційних технологій

ЗАТВЕРДЖУЮ

Проректор з навчально-методичної роботи

 **Ярослав ШТАНЬКО**
“31” серпня 2026 р.

СУЧАСНА АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Галузь знань:	F Інформаційні технології
Спеціальність:	F Інженерія програмного забезпечення
Освітньо-професійна (освітньо-наукова) програма:	Інженерія програмного забезпечення
Освітній рівень:	другий (магістерський)
Статус дисципліни:	обов'язкова
Мова викладання, навчання та оцінювання:	українська
Семестр викладання	2 семестр (1 курс)

**Івано-Франківськ
2026**

РОЗРОБНИК:

к.т.н., доцент



Микола ДЕМЧИНА

ЗАТВЕРДЖЕНО:

на засіданні кафедри інформаційних технологій, протокол № 1 від 27 серпня 2026

PhD, завідувач кафедри



Олександр ІВАНОВ

УЗГОДЖЕНО:

Гарант ОП

PhD, завідувач кафедри



Олександр ІВАНОВ

СХВАЛЕНО:

на засіданні Науково-методичної ради, протокол № 1 від 31 серпня 2026 року

е-mail	mykola.demchyna@ukd.edu.ua
Номер аудиторії чи кафедри	Кафедра інформаційних технологій, ауд. 206
Посилання на сайт УКД	https://ukd.edu.ua
Сторінка курсу в СДО	https://online.ukd.edu.ua

Консультації проводяться за графіком кафедри та за попереднім погодженням із викладачем. Матеріали, завдання, терміни подання й результати оцінювання розміщуються в СДО.

ВСТУП

Анотація навчальної дисципліни

Мета вивчення дисципліни “Сучасна архітектура ПЗ” – надати студентам повного розуміння різноманітних моделей і технологій сучасної архітектури програмного забезпечення. Цей курс дозволить студентам отримати практичний досвід у проектуванні, реалізації та оцінці архітектури програмних додатків у сучасних середовищах розробки.

Основні завдання, що стоять перед авторами курсу, полягають в тому, щоб розкрити здобувачам:

- основні концепції і принципи сучасної архітектури програмного забезпечення;
- провести аналіз різних сучасних шаблонів архітектури програмного забезпечення;
- важливість контейнеризації, оркестровки та доменно-керованого дизайну в сучасній розробці програмного забезпечення;
- практичні навички проектування та реалізації сучасних архітектур програмного забезпечення, використовуючи реальні приклади та тематичні дослідження;
- здатність критично оцінювати архітектури програмного забезпечення та визначати відповідні дизайнерські рішення та компроміси;
- створення детального плану архітектури сучасного програмного забезпечення, включаючи дизайн компонентів, потоку даних і обґрунтування вибору архітектури.

В результаті вивчення дисципліни студент повинен вміти:

- використовувати основні принципи, концепції та практики сучасної архітектури програмного забезпечення;
- розуміти значення архітектурних рішень у забезпеченні масштабованості, продуктивності, безпеки та ремонтпридатності системи;
- виділяти ключові компоненти та принципи проектування різних сучасних архітектурних шаблонів, таких як мікросервіси, безсерверний, та інші;
- розуміти роль платформ контейнеризації та оркестровки, таких як Docker і Kubernetes, в архітектурі розподілених програм;
- застосовувати принципи доменно-орієнтованого проектування для створення гнучких програмних систем, які можна підтримувати;
- використовувати методи оцінювання та порівняння архітектур програмного забезпечення на основі відповідних показників і критеріїв;
- вибирати найкращі практики та міркування, пов'язані з проектуванням, впровадженням та розгортанням сучасних програмних систем.

Компетентності та результати навчання, яких набувають здобувачі освіти внаслідок вивчення навчальної дисципліни (шифри та зміст

компетентностей та програмних результатів навчання вказано відповідно до ОПП/ОНП «Інженерія програмного забезпечення»).

Шифр та назва компетентності	Шифр та назва програмних результатів навчання
ЗК01. Здатність до абстрактного мислення, аналізу та синтезу	РН07. Аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв’язання складних задач інженерії програмного забезпечення. РН08 . Розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника. РН10. Модифікувати існуючі та розробляти нові алгоритмічні рішення детального проектування програмного забезпечення.
ЗК05. Здатність генерувати нові ідеї (креативність).	
СК01. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.	
СК02. Здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення.	
СК03. Здатність проектувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів.	
СК04. Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення.	
СК05. Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення.	
СК06. Здатність ефективно керувати фінансовими, людськими, технічними та іншими проектними ресурсами у сфері інженерії програмного забезпечення.	
СК07. Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв’язувати складні задачі у широких або мультидисциплінарних контекстах.	
СК08. Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення.	

СК09. Здатність забезпечувати якість програмного забезпечення.	
---	--

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Курс	1		
Семестр	2		
Кількість кредитів ECTS	6		
		Денна форма	Заочна форма
Аудиторні навчальні заняття	лекції	28 (в годинах)	8 (в годинах)
	семінари, практичні	32 (в годинах)	8 (в годинах)
Самостійна робота		120 (в годинах)	164 (в годинах)
Форма підсумкового контролю	Екзамен		

СТРУКТУРНО-ЛОГІЧНА СХЕМА ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Пререквізити	Постреквізити
Розробка програмного забезпечення на основі мікросервісної архітектури, Управління якістю програмного забезпечення	Кваліфікаційна робота

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Перелік тем лекційного матеріалу

Тема 1 Основи сучасної архітектури програмного забезпечення (4 год)

Архітектура як система рішень і обмежень. Зацікавлені сторони, функціональні вимоги та атрибути якості. Сценарії якості, модульність, зв'язність і зчеплення. Модульний моноліт і розподілені архітектури. Подання C4, архітектурні точки зору та записи ADR.

Самостійне опрацювання (16 год.). Побудувати карту зацікавлених сторін; сформулювати три вимірювані сценарії якості; порівняти моноліт і мікросервіси; доповнити контекстну діаграму та ADR.

Рекомендовані джерела: [1, 4–6].

Тема 2 Мікросервісна архітектура (4 год)

Межі сервісів і володіння даними. Синхронна та асинхронна взаємодія, API-контракти й сумісність версій. Відмови мережі, тайм-аути, повторні спроби, circuit breaker. Узгодженість даних і незалежне розгортання. Витрати експлуатації та умови доцільності декомпозиції.

Самостійне опрацювання (18 год.). Порівняти два варіанти декомпозиції; описати контракти та сценарій часткової відмови; перевірити повторні запити й обмеження консистентності.

Рекомендовані джерела: [2, 4, 7].

Тема 3 Безсерверна архітектура (4 год)

FaaS і BaaS, подієві тригери та життєвий цикл функції. Стан і зовнішні сховища. Холодний старт, конкурентність, ліміти, права доступу й робота із секретами. Модель витрат і залежність від постачальника. Локальне тестування та критерії вибору serverless.

Самостійне опрацювання (14 год.). Розробити локальний прототип функції, перевірити повторний виклик і помилки; описати модель витрат та порівняти функцію з довготривалим сервісом.

Рекомендовані джерела: [1, 4, 10].

Тема 4 Контейнеризація та оркестрація (4 год)

Контейнерний образ, багатостадійне складання, мережі, томи й конфігурація. Docker і локальне розгортання. Основні об'єкти Kubernetes, декларативне керування, перевірки готовності та життєздатності, запити й ліміти ресурсів. Оновлення, відкат і мінімальні привілеї.

Самостійне опрацювання (20 год.). Підготувати відтворюване середовище; перевірити конфігурацію, ресурсні обмеження та відновлення після зупинки компонента; описати результати та інструкцію запуску.

Рекомендовані джерела: [2, 8, 9].

Тема 5 Подієво-орієнтована архітектура (4 год.)

Події та команди, брокери, черги й публікація-підписка. Порядок повідомлень і семантика доставки. Ідемпотентність, дедуплікація, повторна обробка, dead-letter queue. Transactional outbox і saga. Розмежування CQRS та event sourcing, умови їх застосування.

Самостійне опрацювання (18 год.). Реалізувати й протестувати алгоритм оброблення дублікатів; змодельовати повторну доставку та недоступність споживача; зіставити outbox і пряме надсилання події.

Рекомендовані джерела: [2, 4, 7].

Тема 6 Предметно-орієнтоване проектування (4 год)

Спільна мова, піддомени, обмежені контексти та карта контекстів. Сутності, об'єкти-значення, агрегати й інваріанти. Сервіси предметної області, репозиторії та антикорупційний шар. Узгодження моделі з межами сервісів; еволюція моделі без порушення контрактів.

Самостійне опрацювання (18 год.). Скласти глосарій і карту контекстів; обґрунтувати межу агрегату; реалізувати перевірку інваріантів і набір позитивних та негативних тестів.

Рекомендовані джерела: [2, 3, 7].

Тема 7 Оцінювання та розвиток архітектури (4 год)

Сценарне оцінювання архітектури, дерево корисності, АТАМ і економічний погляд СВАН. Точки чутливості, компроміси та ризики. Продуктивність, безпека, доступність і супроводжуваність. Журнали, метрики, трасування. Експерименти з навантаженням і відмовами; план архітектурного вдосконалення.

Самостійне опрацювання (16 год.). Провести огляд архітектури за сценаріями; порівняти дві альтернативи; зібрати метрики та трасування; сформувати реєстр ризиків і завершити архітектурне дослідження.

Рекомендовані джерела: [1, 4, 5, 11].

Зміст практичних занять

Тема 1. Виявлення шаблонів архітектури програмного забезпечення в існуючих програмах (4 год)

Мета: Аналіз існуючої програми та надання загального огляду архітектури. Виявлення архітектурних моделей і міркування щодо вибору дизайну. Знайомство з різними шаблонами архітектури програмного забезпечення.

Тема 2. Проектування архітектури мікросервісів для зразка програми (4 год)

Мета: Виявлення та визначення меж обслуговування. Проектування зв'язку між мікросервісами. Планування потенційних викликів і проблем впровадження.

Тема 3. Реалізація безсерверної функції та її розгортання в хмарному провайдері (4 год)

Мета: Написання простої безсерверної функції для вирішення конкретного завдання. Розгортання та тестування безсерверних функцій за допомогою хмарної платформи Functions-as-a-Service (FaaS).

Тема 4. Створення та розгортання контейнерної програми за допомогою Docker і Kubernetes (4 год)

Мета: Навчитися створювати Dockerfile для програми. Створення та запуск контейнера Docker локально. Розгортання та керування контейнерними програмами за допомогою Kubernetes.

Тема 5. Проектування системи керованої подіями архітектури для зразка програми (4 год)

Мета: Визначення виробників подій, споживачів і каналів для керованої подіями архітектури. Реалізація обробки подій за допомогою платформи обміну повідомленнями. Забезпечення послідовності та надійності повідомлень.

Тема 6. Застосування принципів дизайну орієнтованого на домен до реальної проблеми (4 год)

Мета: Проектування моделі предметної області на основі реальної проблеми. Ідентифікація піддоменів і обмежених контекстів. Застосування концепцій дизайну, керованого доменом, таких як агрегати, об'єкти значень та сутності.

Тема 7. Оцінка архітектури зразка програми (4 год)

Мета: Застосування методу оцінки архітектури до заданої архітектури. Виявлення потенційних архітектурних ризиків і вдосконалень. Пріоритезація архітектурних проблем і компромісів на основі результатів оцінювання. Відгуки студентів про їхній досвід навчання.

Тема 8. Проектування та реалізація спостережуваності архітектури для зразка програми (4 год)

Мета: Ознайомлення з поняттями спостережуваності (логування, метрики, трасування). Проектування механізмів моніторингу та діагностики для обраної архітектури. Інтеграція інструментів спостережуваності для виявлення збоїв, вузьких місць продуктивності та аналізу поведінки системи під навантаженням.

Зміст самостійної роботи здобувачів

Розподіл годин, виділених на вивчення дисципліни:

Найменування видів робіт	Розподіл годин	
	денна форма	заочна форма
Самостійна робота, год, у т.ч.:	120	164
Опрацювання матеріалу, викладеного на лекціях	12	16
Підготовка до практичних занять та контрольних заходів	48	58
Підготовка звітів з практичних робіт	40	50
Підготовка до поточного контролю	10	20
Опрацювання матеріалу, винесеного на самостійне вивчення	10	20

ПОЛІТИКА КУРСУ

Коротко, з покликанням на відповідну нормативну базу УКД, висвітлити питання:¹

1) щодо системи поточного і підсумкового контролю

Організація поточного та підсумкового семестрового контролю знань студентів, проведення практик та атестації, переведення показників академічної успішності за 100-бальною шкалою в систему оцінок за національною шкалою здійснюється згідно з “Положенням про систему поточного і підсумкового контролю, оцінювання знань та визначення рейтингу здобувачів освіти”. Ознайомитись з документом можна за [посиланням](#).

2) щодо оскарження результатів контрольних заходів

Здобувачі вищої освіти мають право на оскарження оцінки з дисципліни отриманої під час контрольних заходів. Апеляція здійснюється відповідно до «Положення про політику та врегулювання конфліктних ситуацій». Ознайомитись з документом можна за [посиланням](#).

3) щодо відпрацювання пропущених занять

Згідно “Положення про організацію освітнього процесу” здобувач допускається до семестрового контролю з **конкретної**

¹ зміст пунктів може редагуватись з огляду на особливості курсу



навчальної дисципліни (семестрового екзамену, диференційованого заліку), якщо він виконав усі види робіт, передбачені на семестр навчальним планом та силабусом/робочою програмою навчальної дисципліни, підтвердив опанування на мінімальному рівні результатів навчання (отримав ≥ 35 бали), відпрацював визначені індивідуальним навчальним планом всі лекційні, практичні, семінарські та лабораторні заняття, на яких він був відсутній. Ознайомитись з документом можна за [покликанням](#).

4) щодо дотримання академічної доброчесності

“Положення про систему забезпечення академічної доброчесності” закріплює моральні принципи, норми та правила етичної поведінки, позитивного, сприятливого, доброчесного освітнього і наукового середовища, професійної діяльності та професійного спілкування спільноти Університету, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання. Ознайомитись з документом можна за [покликанням](#).



5) щодо використання штучного інтелекту

“Політика використання штучного інтелекту” визначає політику щодо використання технічних засобів на основі штучного інтелекту в освітньому процесі. Ознайомитись з документом можна за [покликанням](#).² “Положення про систему забезпечення академічної доброчесності” містить рекомендації щодо використання в академічних текстах генераторів на основі штучного інтелекту. Ознайомитись з документом можна за [покликанням](#).



б) щодо використання технічних засобів в аудиторії та правила комунікації

Використання мобільних телефонів, планшетів та інших гаджетів під час лекційних та практичних занять дозволяється виключно у навчальних цілях (для уточнення певних даних, перевірки правопису, отримання довідкової інформації тощо). На гаджетах повинен бути активований режим «без звуку» до початку заняття. Під час занять заборонено надсилання текстових повідомлень, прослуховування музики, перевірка електронної пошти, соціальних мереж тощо. Під час виконання заходів контролю використання гаджетів заборонено (за винятком, коли це передбачено умовами його проведення). У разі порушення цієї заборони результат анулюється без права перескладання. Ознайомитись з відповідним документом можна за [посиланням](#).



Комунікація відбувається через електронну пошту і сторінку дисципліни в Moodle.

7) щодо зарахування результатів навчання, здобутих шляхом формальної/інформальної освіти

Процедури визнання результатів навчання, здобутих шляхом формальної/інформальної освіти визначаються «Положенням про порядок визнання результатів навчання, здобутих шляхом неформальної



² визначається політика використання ШІ в навчальній дисципліні - дозволено використання

та / або інформальної освіти». Ознайомитись з документом можна за [посиланням](#).³

МЕТОДИ НАВЧАННЯ

При вивченні дисципліни застосовується комплекс методів для організації навчання студентів з метою розвитку їх логічного та абстрактного мислення, творчих здібностей, підвищення мотивації до навчання та формування особистості майбутнього фахівця.

Програмний результат навчання ⁴	Метод навчання	Метод оцінювання
Аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв'язання складних задач інженерії програмного забезпечення. Розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника. Модифікувати існуючі та розробляти нові алгоритмічні рішення детального проектування програмного забезпечення.	словесні методи (лекція, розповідь пояснення); наочні методи (ілюстрування, комп'ютерні і мультимедійні методи, зразки коду, зразки програм); інтерактивні методи (дискусія, мозковий штурм, робота в команді (групах)); метод симулювання проектів;	екзамен

ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Контрольні заходи

Вид	Зміст	% від загальної	Бал
-----	-------	-----------------	-----

³ визначається перелік електронних та інших ресурсів та умови перезарахування

⁴ для вибіркового навчального дисципліни вказується **результат навчання**

		<i>оцінки</i>	<i>min</i>	<i>max</i>
Поточні контрольні заходи	всього	60	35	60
Підсумкові контрольні заходи	екзамен	40	25	40
Всього:		100	60	100

Процедура проведення контрольних заходів, а саме поточного контролю знань протягом семестру та підсумкового семестрового контролю, регулюється «Положенням про систему поточного та підсумкового контролю оцінювання знань та визначення рейтингу студентів».

Фіксація **поточного** контролю здійснюється в “Електронному журналі обліку успішності академічної групи” на підставі чотирибальної шкали - “2”; “3”; “4”; “5”. У разі відсутності студента на занятті виставляється “н”. За результатами поточного контролю у Журналі, автоматично визначається підсумкова оцінка, здійснюється підрахунок пропущених занять.

Усі пропущені заняття, а також негативні оцінки студенти зобов'язані відпрацювати впродовж трьох наступних тижнів. У випадку недотримання цієї норми, замість “н” в журналі буде виставлено “0” (нуль балів), без права перездачі. Відпрацьоване лекційне заняття в електронному журналі позначається літерою «в».

Критерії оцінювання (за необхідності, поточного та/або підсумкового контролю):

«незадовільно»	володіє навчальним матеріалом на рівні елементарного розпізнавання і відтворення окремих фактів, елементів, об'єктів, що позначаються окремими словами чи реченнями; володіє матеріалом на елементарному рівні засвоєння, викладає його уривчастими реченнями, виявляє здатність висловити думку на елементарному рівні; володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу;
«задовільно»	володіє матеріалом на початковому рівні, значну частину матеріалу відтворює на репродуктивному рівні; володіє матеріалом на рівні, вищому за початковий, здатний за допомогою викладача логічно відтворити значну його частину; може відтворити значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, за допомогою викладача може аналізувати навчальний

	матеріал, порівнювати та робити висновки, виправляти допущені помилки;
«добре»	здатний застосовувати вивчений матеріал на рівні стандартних ситуацій, частково контролювати власні навчальні дії, наводити окремі власні приклади на підтвердження певних тверджень: вміє порівнювати, узагальнювати, систематизувати інформацію під керівництвом викладача, в цілому самостійно застосовувати її на практиці, контролювати власну діяльність, виправляти помилки і добирати аргументи на підтвердження певних думок під керівництвом викладача; вільно володіє вивченим обсягом матеріалу, та вміє застосовувати його на практиці; вільно розв'язує задачі в стандартних ситуаціях, самостійно виправляє допущені помилки, добирає переконливі аргументи на підтвердження вивченого матеріалу;
«відмінно»	виявляє початкові творчі здібності, самостійно визначає окремі цілі власної навчальної діяльності, оцінює окремі нові факти, явища, ідеї; знаходить джерела інформації та самостійно використовує їх відповідно до цілей, поставлених викладачем; вільно висловлює власні думки і відчуття, визначає програму особистої пізнавальної діяльності, самостійно оцінює різноманітні життєві явища і факти, виявляючи особисту позицію щодо них; без допомоги викладача знаходить джерела інформації і використовує одержані відомості відповідно до мети та завдань власної пізнавальної діяльності; використовує набуті знання і вміння в нестандартних ситуаціях; виявляє особливі творчі здібності, самостійно розвиває власні обдарування і нахили, вміє самостійно здобувати знання.

До підсумкового контролю допускаються студенти які за результатами поточного контролю отримали не менше 35 балів. Усі студенти, що отримали 34 балів і менше, не допускаються до складання підсумкового контролю і на підставі укладання додаткового договору, здійснюють повторне вивчення дисципліни впродовж наступного навчального семестру. За результатами підсумкового контролю (диференційований залік/екзамен) студент може отримати 40 балів. Студенти, які під час підсумкового контролю отримали 24 бали і менше, вважаються такими, що не здали екзамен/диференційований залік і повинні йти на перездачу.

Загальна семестрова оцінка з дисципліни, яка виставляється в екзаменаційних відомостях оцінюється в балах (згідно з **Шкалою оцінювання знань за ЄКТС**) і є сумою балів отриманих під час поточного та підсумкового контролю.

Шкала оцінювання знань за ЄКТС:

Оцінка за національною шкалою	Рівень досягнень, %	Шкала ECTS
Національна диференційована шкала		
Відмінно	90 – 100	A
Добре	83 – 89	B
	75 – 82	C
Задовільно	67 – 74	D
	60 – 66	E
Незадовільно	35 – 59	FX
	0 – 34	F
Національна недиференційована шкала		
Зараховано	60 – 100	-
Не зараховано	0 – 59	-

Студенти, які не з'явилися на заліки/екзамени без поважних причин, вважаються такими, що одержали незадовільну оцінку.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ⁵

Основна література

1. Bass L., Clements P., Kazman R. Software Architecture in Practice. 4th ed. Addison-Wesley Professional, 2021. URL: <https://www.oreilly.com/library/view/software-architecture-in/9780136885979/> (дата звернення: 22.09.2026).
2. Newman S. Building Microservices. 2nd ed. O'Reilly Media, 2021. URL: <https://www.oreilly.com/library/view/building-microservices-2nd/9781492034018/> (дата звернення: 22.09.2026).
3. Khononov V. Learning Domain-Driven Design. O'Reilly Media, 2021. URL: <https://www.oreilly.com/library/view/learning-domain-driven-design/9781098100124/> (дата звернення: 22.09.2026).

⁵ обов'язково: враховувати вимоги [ДСТУ 8302:2015](#) (відповідно до [Наказу № 65, від 4.03. 2016](#)), [рекомендації](#) Національного агентства з забезпечення якості вищої освіти, використовувати літературу за останні 5-7 років, наводити власні публікації за змістом навчальної дисципліни.

4. Ford N., Richards M., Sadalage P., Dehghani Z. Software Architecture: The Hard Parts. O'Reilly Media, 2021. URL: <https://www.oreilly.com/library/view/software-architecture-the/9781492086888/> (дата звернення: 22.09.2026).

Електронні інформаційні ресурси

6. Brown S. The C4 model for visualising software architecture. URL: <https://c4model.com/> (дата звернення: 22.09.2026).

7. Richardson C. Microservices.io. A pattern language for microservices. URL: <https://microservices.io/patterns/index.html> (дата звернення: 22.09.2026).

8. Docker. Get started. Documentation. URL: <https://docs.docker.com/get-started/> (дата звернення: 22.09.2026).

9. Kubernetes. Concepts. Documentation. URL: <https://kubernetes.io/docs/concepts/> (дата звернення: 22.09.2026).

Додаткова література

5. Стисло Т. Р., Стисло О. В., Демчина М. М., Білоус В. В. Архітектура та методологія інтеграційного фреймворку для розробки програмного забезпечення на основі концепції Viewpoints. Вимірювальна та обчислювальна техніка в технологічних процесах. 2024. № 2. С. 65–73. URL: <https://repository.ukd.edu.ua/handle/123456789/2118> (дата звернення: 22.09.2026).