



Міжнародний гуманітарний університет
Факультет кібербезпеки, програмної інженерії
та комп'ютерних наук
Кафедра комп'ютерної інженерії та інноваційних
технологій



«ЗАТВЕРДЖУЮ»

Ректор Міжнародного
гуманітарного університету
Д.ю.н., професор

Костянтин ГРОМОВЕНКО

«29» серпня 2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ТЕХНОЛОГІЇ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ

Галузь знань	<u>12 Інформаційні технології</u>
Спеціальність	<u>121 Інженерія програмного забезпечення</u>
Назва освітньої програми	<u>Інженерія програмного забезпечення</u>
Рівень вищої освіти	<u>перший (бакалаврський)</u>

Одеса – 2025 рік

Робоча програма затверджена на засіданні кафедри комп'ютерної інженерії та інноваційних технологій

Протокол № 1 від 26 серпня 2025 року.

Розробники і викладачі	Контактний тел.	E-mail
доцент кафедри комп'ютерної інженерії та інноваційних технологій, к.т.н., доцент Лебедєва Олена Юріївна	+380674858619	lebedeva.onmu@gmail.com

Завідувачка кафедри комп'ютерної інженерії та інноваційних технологій
к.т.н., доцент

 Лариса ЙОНА

Гарант освітньої програми
к.т.н., доцент

 Олександр РУСУ

Узгоджено
Начальник навчального відділу

 Лілія САЄНКО

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
Кількість кредитів – 5, загальна кількість годин – 150	Галузь – 12 Інформаційні технології Спеціальність – 121 Інженерія програмного забезпечення	Денна форма	Заочна форма
		обов'язкова	
		Рік підготовки:	
		2	
Мова навчання – українська	Рівень вищої освіти – перший (бакалаврський)	Семестр:	
		3	
		Лекційні заняття:	
		26	12
		Практичні заняття:	
		38	8
		Лабораторні роботи	
		0	0
		Самостійна робота	
		86	130
		Вид контролю:	
		екзамен	

Дисципліна «Технології створення програмних продуктів» спрямована на формування системного розуміння процесів розробки програмного забезпечення від етапу формування вимог до введення програмного продукту в експлуатацію та його супроводження. Курс охоплює моделі життєвого циклу програмного забезпечення, сучасні методології розробки, принципи архітектурного проєктування, компонентний підхід, використання систем контролю версій та інструментальних засобів автоматизації. Значна увага приділяється забезпеченню якості програмного забезпечення, тестуванню, верифікації та валідації, а також управлінню змінами та інтеграції програмних компонентів.

У межах дисципліни розглядаються питання організації процесу розробки, документування, застосування сучасних технологій та стандартів програмної інженерії. Практична складова курсу орієнтована на набуття здобувачами досвіду проєктування архітектури, модульної реалізації, інтеграції компонентів, застосування систем контролю версій та процедур погодження змін, що дає змогу сформувати навички професійної взаємодії в процесі створення програмного продукту.

МЕТА ДИСЦИПЛІНИ. Метою викладання навчальної дисципліни є формування у здобувачів здатності реалізовувати фази та ітерації життєвого циклу програмного забезпечення із застосуванням сучасних методологій, технологій та інструментальних засобів, проєктувати архітектуру програмних систем, забезпечувати якість програмного продукту та організовувати процес його інтеграції й супроводження.

Опанування дисципліни забезпечує розвиток умінь обґрунтовано обирати технології розробки, застосовувати методи компонентної реалізації, здійснювати тестування та контроль якості, планувати етапи виконання проєкту, а також використовувати системи контролю версій і засоби автоматизації процесів розробки.

ПРЕРЕКВІЗИТИ. Необхідною умовою для опанування дисципліни є розуміння предметної області інженерії програмного забезпечення, основних видів професійної

діяльності фахівця, загальних етапів створення програмних систем та їх місця в інформаційних технологіях, сформоване під час вивчення дисципліни «Вступ до спеціальності», а також у процесі проходження навчальної практики. Здобувач повинен орієнтуватися в структурі програмного продукту, мати уявлення про вимоги до якості програмного забезпечення та володіти базовими навичками роботи з програмними інструментами.

ПОСТРЕКВІЗИТИ. Знання та вміння, отримані під час вивчення дисципліни, є основою для подальшого опанування навчальних компонент: «Управління ІТ-проектами», «Організація баз даних та знань», «Якість та тестування програмного забезпечення», «Економічна ефективність програмних систем», а також для проходження технологічної, виробничої та переддипломної практик і виконання кваліфікаційної роботи. Набуті компетентності забезпечують здатність планувати етапи створення програмного продукту, обґрунтовувати вибір технологій і методів розробки, здійснювати інтеграцію та супроводження програмних систем.

2. ОЧІКУВАНІ КОМПЕТЕНТНОСТІ, ЯКІ ПЛАНУЄТЬСЯ СФОРМУВАТИ, ТА ДОСЯГНЕННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

У процесі реалізації програми дисципліни формуються наступні компетентності із передбачених освітньою програмою:

Інтегральна компетентність

Здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності

K07. Здатність працювати в команді.

Спеціальні (фахові, предметні) компетентності

K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

K16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

K24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

Результати навчання

ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПР17. Вміти застосовувати методи компонентної розробки програмного забезпечення.

ПР19. Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.

ПР20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.

ПР22. Знати та вміти застосовувати методи та засоби управління проектами.

Заплановані результати навчання за навчальною дисципліною

Знання:

- основні процеси, фази та ітерації життєвого циклу програмного забезпечення;
- класичні та гнучкі методології розробки програмних систем, їх принципи та сфери застосування;
- принципи формування, аналізу та документування вимог до програмного забезпечення;
- підходи до проектування архітектури програмних систем та компонентної реалізації;
- сучасні інструментальні засоби розробки, системи контролю версій та засоби автоматизації;
- атрибути якості програмного забезпечення, стандарти та моделі оцінювання якості;
- методи тестування, верифікації та валідації програмного забезпечення;
- основи управління IT-проектами, управління змінами та конфігурацією програмних систем.

Уміння:

- обирати та обґрунтовувати модель життєвого циклу і методологію розробки залежно від задачі;
- формулювати функціональні та нефункціональні вимоги до програмного продукту;
- проектувати архітектуру програмної системи та визначати структуру її компонентів;
- застосовувати системи контролю версій для організації змін і інтеграції модулів;
- реалізовувати компонентний підхід під час розробки програмного забезпечення;
- розробляти тестові сценарії та здійснювати модульне й інтеграційне тестування;
- оцінювати якість програмного забезпечення та аналізувати результати перевірки;
- планувати етапи виконання програмного проекту та оцінювати трудомісткість робіт.

Навички:

- системного аналізу процесів створення програмного забезпечення;
- застосування інструментальних засобів розробки та автоматизації збирання програмного продукту;
- організації роботи з репозиторіями та узгодження змін у процесі розробки;
- документування вимог, архітектурних рішень і результатів тестування;
- інтеграції програмних компонентів у цілісну систему;
- контролю якості та супроводження програмного продукту на різних етапах його життєвого циклу;
- професійної взаємодії під час обговорення та погодження технічних рішень.

3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Етапи життєвого циклу програмного забезпечення

Поняття програмного продукту та його відмінність від програмного коду. Життєвий цикл програмного забезпечення. Основні процеси, фази та ітерації. Каскадна, інкрементна та спіральна моделі. Переваги та обмеження кожної моделі. Вибір моделі залежно від типу проєкту.

Гнучкі методології розробки. Принципи Agile. Scrum: ролі, артефакти, події. Kanban та управління потоком задач. XP-практики. Організація ітераційної роботи. Комунікація в процесі розробки та узгодження рішень.

Формування та аналіз вимог. Джерела вимог. Функціональні та нефункціональні вимоги. Специфікація вимог до програмного забезпечення. Трасування вимог. Управління змінами вимог.

Тема 2. Проєктування та архітектура програмних систем

Поняття архітектури програмного забезпечення. Архітектурні стилі: монолітна, клієнт-серверна, багаторівнева, мікросервісна. Принципи модульності та декомпозиції. Забезпечення масштабованості та розширюваності системи.

Моделювання структури та поведінки системи. UML-діаграми: варіантів використання, класів, послідовностей, компонентів. Проєктування інтерфейсів взаємодії. Контракти та специфікації API.

Компонентна розробка. Повторне використання модулів. Інтеграція програмних компонентів. Управління залежностями. Версіонування та сумісність компонентів.

Тема 3. Технології реалізації програмного забезпечення

Організація роботи з використанням систем контролю версій. Git: репозиторії, коміти, гілки, злиття. Конфлікти та їх розв'язання. Pull request і код-рев'ю як інструмент узгодження змін.

Інструментальні засоби розробки. Середовища програмування. Системи збирання та керування залежностями. Автоматизація збирання. Безперервна інтеграція. Контроль якості коду.

Робота з даними у програмних продуктах. Інформаційні моделі. Організація зберігання даних. Обробка та валідація даних. Взаємодія з базами даних та зовнішніми сервісами.

Тема 4. Забезпечення якості програмного забезпечення

Поняття якості програмного забезпечення. Атрибути якості. Моделі якості. Стандарти у сфері програмної інженерії. Вимоги до якості на різних етапах життєвого циклу.

Тестування програмного забезпечення. Рівні тестування. Модульне, інтеграційне, системне та приймальне тестування. Тестові сценарії. Автоматизоване тестування.

Верифікація та валідація. Статичний аналіз коду. Динамічний аналіз. Документування результатів тестування. Усунення дефектів і контроль якості.

Тема 5. Управління проєктами та супроводження програмних продуктів

Основи управління IT-проєктами. Планування робіт. Оцінювання трудомісткості. Управління ризиками. Контроль виконання завдань. Інструменти управління проєктами.

Управління змінами та конфігурацією. Реліз-менеджмент. Підтримка цілісності та стабільності системи. Документування змін.

Супроводження програмного забезпечення. Коригувальне, адаптивне та еволюційне супроводження. Масштабування систем. Технічна підтримка. Професійна документація та взаємодія із замовником.

4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістових модулів і тем	Кількість годин									
	Денна форма					Заочна форма				
	Заг.	у тому числі				Заг.	у тому числі			
		Лек.	Прак.	Лаб.	Сам. роб.		Лек.	Прак.	Лаб.	Сам. роб.
Тема 1. Етапи життєвого циклу програмного забезпечення	30	6	8	0	16	30	2	2	0	26
Тема 2. Проєктування та архітектура програмних систем	34	6	10	0	18	34	4	2	0	28
Тема 3. Технології реалізації програмного забезпечення	32	6	10	0	16	32	2	2	0	28
Тема 4. Забезпечення якості програмного забезпечення	30	4	6	0	20	30	2	2	0	26
Тема 5. Управління проєктами та супроводження програмних продуктів	24	4	4	0	16	24	2	0	0	22
Загалом	150	26	38	0	86	150	12	8	0	130
Підсумковий контроль – екзамен										

5. ПРАКТИЧНІ РОБОТИ

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 1. Етапи життєвого циклу програмного забезпечення 1. Аналіз прикладу програмного продукту та визначення моделі життєвого циклу. 2. Формування функціональних і нефункціональних вимог до програмної системи. 3. Побудова специфікації вимог та таблиці трасування.	8	2

4. Планування ітерації розробки за підходом Scrum.		
Тема 2. Проєктування та архітектура програмних систем 1. Розробка UML-діаграми варіантів використання та діаграми класів. 2. Проєктування архітектури програмної системи та вибір архітектурного стилю. 3. Декомпозиція системи на модулі та визначення інтерфейсів взаємодії. 4. Реалізація прототипу компонента з визначеним API. 5. Інтеграція декількох модулів у єдину систему.	10	2
Тема 3. Технології реалізації програмного забезпечення 1. Створення репозиторію Git та організація роботи з гілками. 2. Виконання злиття змін та розв'язання конфліктів у репозиторії. 3. Налаштування системи збирання та автоматизації проєкту. 4. Організація взаємодії програмного коду з базою даних. 5. Реалізація обробки та валідації вхідних даних у програмному модулі.	10	2
Тема 4. Забезпечення якості програмного забезпечення 1. Розробка тестових сценаріїв для програмного модуля. 2. Реалізація модульного тестування. 3. Проведення інтеграційного тестування та аналіз результатів. 4. Виконання статичного аналізу коду та оформлення звіту про дефекти.	6	2
Тема 5. Управління проєктами та супроводження програмних продуктів 1. Побудова плану реалізації програмного проєкту з оцінюванням трудомісткості. 2. Оформлення документації та підготовка релізу програмного продукту.	4	0
Загалом	38	8

6. САМОСТІЙНА РОБОТА

До самостійної роботи здобувачів відносяться наступні види діяльності:

1. Опрацювання лекційного матеріалу.
2. Підготовка до практичних занять.
3. Консультації з викладачем впродовж семестру.
4. Ознайомлення з додатковою літературою відповідно до зазначених у програмі тем.
5. Самостійне опрацювання окремих питань навчальної дисципліни.
6. Підготовка індивідуальних завдань у вигляді презентацій або рефератів.
7. Підготовка до підсумкового контролю.

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 1. Етапи життєвого циклу програмного забезпечення 1. Порівняння моделей життєвого циклу програмного забезпечення та сфери їх застосування. 2. Аналіз принципів Agile та їх відмінність від класичних моделей. 3. Ролі та артефакти Scrum-процесу. 4. Методи збору та формалізації вимог до програмного забезпечення. 5. Нефункціональні вимоги та їх вплив на архітектуру системи. 6. Управління змінами вимог у процесі розробки.	16	26
Тема 2. Проєктування та архітектура програмних систем 1. Архітектурні стилі програмних систем та критерії їх вибору. 2. Принципи модульності та слабкої зв'язності компонентів. 3. Використання UML-діаграм у процесі проєктування. 4. Проєктування програмних інтерфейсів та контрактів взаємодії. 5. Компонентна розробка та повторне використання коду. 6. Проблеми інтеграції програмних модулів.	18	28

7. Забезпечення масштабованості та розширюваності програмної системи.		
Тема 3. Технології реалізації програмного забезпечення 1. Організація роботи з системами контролю версій у процесі розробки. 2. Стратегії гілкування та управління змінами в Git. 3. Інструментальні засоби автоматизації збирання програмного продукту. 4. Принципи безперервної інтеграції. 5. Інформаційні моделі даних у програмних продуктах. 6. Методи забезпечення коректності та валідації даних.	16	28
Тема 4. Забезпечення якості програмного забезпечення 1. Поняття якості програмного забезпечення та її характеристики. 2. Міжнародні стандарти якості програмного забезпечення. 3. Рівні та види тестування програмного забезпечення. 4. Модульне та інтеграційне тестування. 5. Верифікація та валідація програмного забезпечення. 6. Документування результатів тестування та управління дефектами.	20	26
Тема 5. Управління проєктами та супроводження програмних продуктів 1. Основи управління IT-проєктами та життєвий цикл проєкту. 2. Методи оцінювання трудомісткості програмних проєктів. 3. Управління ризиками у процесі розробки програмного забезпечення. 4. Управління конфігурацією та реліз-менеджмент. 5. Види супроводження програмного забезпечення. 6. Професійна документація програмного продукту та її структура.	16	22
Загалом	86	130

7. ВИДИ ТА МЕТОДИ КОНТРОЛЮ

Види контролю	Складові оцінювання
Поточний контроль , який здійснюється під час проведення практичних (лабораторних) занять, виконання самостійної роботи, індивідуальних завдань, дослідної роботи здобувача тощо	50%
Підсумковий контроль , який здійснюється у ході проведення екзамену	50%

Методи діагностики знань (контролю)	Фронтальне опитування, розв'язання практичних завдань, тестування здобувачів, наукові доповіді, реферати, усні повідомлення, екзамен
--	--

Питання для підсумкового контролю

1. Поняття програмного продукту та його складові.
2. Життєвий цикл програмного забезпечення та його основні процеси.
3. Каскадна модель розробки та її особливості.
4. Інкрементна та спіральна моделі життєвого циклу.
5. Принципи Agile-підходу до розробки програмного забезпечення.
6. Методологія Scrum: ролі, події та артефакти.
7. Kanban та управління потоком задач у проєкті.
8. Функціональні та нефункціональні вимоги до програмного забезпечення.
9. Специфікація вимог та їх трасування.
10. Поняття архітектури програмного забезпечення.
11. Основні архітектурні стилі програмних систем.
12. Принципи модульності та декомпозиції програмної системи.
13. Масштабованість і розширюваність програмного продукту.
14. UML-діаграма варіантів використання та її призначення.

15. UML-діаграма класів та її використання під час проєктування.
16. Проєктування інтерфейсів взаємодії компонентів.
17. Компонентна розробка програмного забезпечення.
18. Інтеграція програмних модулів у єдину систему.
19. Призначення та можливості систем контролю версій.
20. Основні команди Git та організація роботи з гілками.
21. Злиття змін і розв'язання конфліктів у репозиторії.
22. Організація код-рев'ю та погодження змін.
23. Інструментальні засоби розробки програмного забезпечення.
24. Системи збирання та автоматизації програмних проєктів.
25. Принципи безперервної інтеграції програмного забезпечення.
26. Інформаційні моделі даних у програмних продуктах.
27. Організація взаємодії програмного забезпечення з базами даних.
28. Поняття якості програмного забезпечення.
29. Атрибути та моделі якості програмного забезпечення.
30. Міжнародні стандарти у сфері програмної інженерії.
31. Рівні тестування програмного забезпечення.
32. Модульне тестування програмних компонентів.
33. Інтеграційне та системне тестування.
34. Верифікація та валідація програмного забезпечення.
35. Статичний і динамічний аналіз програмного коду.
36. Документування результатів тестування та управління дефектами.
37. Основні процеси управління IT-проєктами.
38. Планування та оцінювання трудомісткості програмного проєкту.
39. Управління ризиками в процесі розробки програмного забезпечення.
40. Управління конфігурацією програмного забезпечення.
41. Управління змінами та реліз-менеджмент.
42. Види супроводження програмного забезпечення.
43. Професійна документація програмного продукту та її структура.
44. Взаємодія із замовником у процесі розробки програмного забезпечення.
45. Забезпечення цілісності, надійності та підтримки програмного продукту.

8. ОЦІНЮВАННЯ ПОТОЧНОЇ, САМОСТІЙНОЇ ТА ІНДИВІДУАЛЬНОЇ РОБОТИ ЗДОБУВАЧІВ

Види роботи	Планові терміни виконання	Форми контролю та звітності	Максимальний відсоток оцінювання
Систематичність і активність роботи на практичних заняттях			
1.1. Підготовка до практичних занять	Відповідно до робочої програми та розкладу занять	Перевірка обсягу та якості засвоєного матеріалу під час практичних занять	30
Виконання завдань для самостійного опрацювання			
1.2. Індивідуальне тестування	Відповідно до робочої програми та розкладу занять	Перевірка результатів індивідуального тестування, перевірка конспектів лекцій, перевірка рефератів	10
Виконання індивідуальних завдань (дослідна робота здобувача)			

1.3. Інші види індивідуальних завдань, в т.ч. підготовка наукових публікацій, участь у роботі круглих столів, конференцій тощо.	Відповідно до робочої програми та розкладу занять	Обговорення результатів проведеної роботи під час аудиторних занять, наукових конференцій та круглих столів.	10
Разом балів за поточний контроль			50
Підсумковий контроль – екзамен			50
Загалом балів			100

Поточний контроль знань здобувачів при підсумковому контролі у формі екзамену (максимум 50 балів) оцінюється за шкалою («2», «3», «4», «5»), з обов'язковим переведенням балів за кожний вид роботи таким чином:

- за підготовку до аудиторних занять (максимум 30 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості практичних (лабораторних) занять, для переведу балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 6;

- за виконання завдань для самостійного опрацювання (тестування, завдання, підготовка реферату за заданою тематикою) (максимум 10 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості програмного матеріалу, що виноситься на самостійне вивчення, для переведу зазначених балів середньоарифметична оцінка множиться на коефіцієнт 2;

- за інші види індивідуальних завдань, в т.ч. підготовку наукових публікацій, участь у роботі круглих столів, конференцій тощо (максимум 10 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості інших видів індивідуальних завдань, для переведу балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 2.

Критерії оцінювання поточного контролю

- «2» – виставляється за неправильну відповідь (письмову роботу), яка не відповідає змісту теми та свідчить про нерозуміння її основних положень;

- «3» – виставляється за відповідь (письмову роботу), яка відповідає змісту теми, але є неповною і неточною у визначенні понять, свідчить про неповне розуміння основних положень навчального матеріалу, свідчить про те, що знання здобувача мають фрагментарний, поверховий характер;

- «4» – оцінюється правильна і обґрунтована відповідь (письмова робота), з якої видно, що здобувач знає й розуміє теоретичний матеріал в обсязі навчальної теми, володіє необхідними навичками, не допускає суттєвих помилок.

- «5» – оцінюється повна, правильна, ґрунтовна відповідь (письмова робота) на запитання теми, що передбачає логічність і послідовність викладу матеріалу, володіння термінологією, показує вміння повно й глибоко використовувати теоретичні знання в практичній площині.

Критерії оцінювання підсумкового контролю у формі екзамену

Максимальна оцінка підсумкового контролю у формі екзамену не може перевищувати 50 балів. При цьому рівень знань оцінюється:

– **від 40 до 50 балів** – здобувач виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки;

– **від 30 до 40 балів** – здобувач володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді;

– **від 20 до 30 балів** – здобувач відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки.

– **від 10 до 20 балів** – здобувач володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих;

– **від 0 до 10 балів** – здобувач володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає необґрунтовані, невичерпні відповіді, допускає помилки.

9. ОЦІНЮВАННЯ ЗАГАЛЬНИХ РЕЗУЛЬТАТІВ ЗНАНЬ ЗДОБУВАЧІВ

Загальні результати знань здобувачів по даній дисципліні оцінюються наступним чином:

– **«Відмінно»/«зараховано» (А)** – від 90 до 100 балів. Здобувач виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки. Був присутній на лекціях та практичних заняттях, під час яких давав вичерпні, обґрунтовані, теоретично і практично правильні відповіді, має конспект з виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (В)** – від 82 до 89 балів. Здобувач володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді. Був присутній на лекціях та практичних заняттях, має конспект з виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (С)** – від 74 до 81 балів. Здобувач відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи, реферату та активність у науково-дослідній роботі;

– **«Задовільно»/«зараховано» (D)** – від 64 до 73 балів. Здобувач був присутній не на всіх лекціях та практичних заняттях, володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи та рефератів;

– **«Задовільно»/«зараховано» (Е)** – від 60 до 63 балів. Здобувач був присутній не на всіх лекціях та практичних заняттях, володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає

необґрунтовані, невичерпні відповіді, допускає помилки, має неповний конспект з завданнями до самостійної роботи.

– **«Незадовільно з можливістю повторного складання»/«не зараховано» (FX)** – від 35 до 59 балів. Здобувач володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

– **«Незадовільно з обов’язковим повторним вивченням дисципліни»/«не зараховано» (F)** – від 0 до 34 балів. Здобувач не володіє навчальним матеріалом.

Таблиця відповідності результатів контролю знань за різними шкалами

100-бальна шкала	Шкала за ECTS	Національна шкала	
		Екзамен	Залік
90-100	A	Відмінно	Зараховано
82-89	B	Добре	Зараховано
74-81	C		
64-73	D	Задовільно	Зараховано
60-63	E		
35-59	FX	Незадовільно	Не зараховано
0-34	F		

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Pressman, R. S., Maxim, B. R. Software Engineering: A Practitioners Approach. McGraw-Hill Education. 2019. 1408p. ISBN: 9781260548006, 1260548007.
2. Sommerville, I. Software Engineering, Global Edition. Pearson Education. 2016. 816p. ISBN: 9781292096148, 1292096144.
3. Martin, R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Prentice Hall. 2018. 404p. ISBN: 9780134494166, 0134494164.
4. Liang, Q. Continuous Delivery 2.0: Business-leading DevOps Essentials. CRC Press. 2021. 362p. ISBN: 9781000474763, 1000474763.
5. Chacon, Scott. Pro Git: Everything You Need to Know About Git. Apress, 2022. 497p.

Допоміжна

6. Ahmed, Ashfaq, and Prasad, Bhanu. Foundations of Software Engineering. CRC Press, 2016. 475p. ISBN: 9781498737609, 1498737609
7. Adewole, Ayobami. C# and .NET Core Test-Driven Development: Dive Into TDD to Create Flexible, Maintainable, and Production-ready .NET Core Applications. Packt Publishing, 2018. 300p. ISBN: 9781788299923, 1788299922
8. Peters, Lawrence J.. Software Project Management: Methods and Techniques. CRC Press. 244p. ISBN: 9781040035238, 104003523X.
9. Gechman, Marvin. Project Management of Large Software-Intensive Systems. CRC Press, 2019. 392p. ISBN: 9780429648168, 0429648162.
10. Singh, Shweta. Role of Project Management Software in the Manufacturing Sector. Германия, GRIN Verlag, 2022. 102p. ISBN: 9783346776754, 3346776751

11. ISO/IEC 25010:2011 Systems and software engineering – Software product quality requirements and evaluation (SQuaRE) – System and software quality models.
12. ISO/IEC/IEEE 12207:2017 Systems and software engineering – Software life cycle processes.

Інформаційні ресурси

13. Scrum Guide. URL: <https://scrumguides.org>
14. Git Documentation. URL: <https://git-scm.com/docs>
15. GitHub Docs. URL: <https://docs.github.com>
16. Atlassian Agile Coach. URL: <https://www.atlassian.com/agile>
17. Software Engineering Institute (SEI). URL: <https://www.sei.cmu.edu>