

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра програмних систем і технологій

«ЗАТВЕРДЖУЮ»

Заступник декана

з навчально-виховної роботи

Наталія ТМЄНОВА

« 29 » червня 2026 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«ЗАСОБИ РОЗРОБКИ ТА ПІДТРИМКИ ЖИТТЄВОГО ЦИКЛУ ПЗ»

для здобувачів вищої освіти

галузь знань
спеціальність
освітній рівень
освітня програма
вид дисципліни

**F Інформаційні технології
F2 Інженерія програмного забезпечення
магістр
Інженерія програмного забезпечення
обов'язкова**

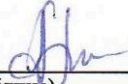
Форма навчання	денна
Навчальний рік	2026/2027
Семестр	2
Кількість кредитів ECTS	4
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: **Вовна Олександр Володимирович**, *доктор технічних наук, професор*

Пролонговано: на 2027/2028 н.р. _____ (підпис) _____ (ПІБ) _____ (дата) «___» _____ 2027 р.
на 2028/2029 н.р. _____ (підпис) _____ (ПІБ) _____ (дата) «___» _____ 2028 р.
на 2029/2030 н.р. _____ (підпис) _____ (ПІБ) _____ (дата) «___» _____ 2029 р.

Розробник: **Вовна Олександр Володимирович**, *д.т.н., професор, професор кафедри програмних систем і технологій*

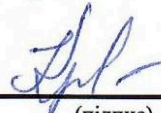
Завідувач кафедри
програмних систем і технологій

 проф. Олексій БИЧКОВ
(підпис)

Протокол № 18 від 17 червня 2026 року

Схвалено науково-методичною комісією факультету інформаційних технологій

Протокол № 11 від 26 червня 2026 року

Голова науково-методичної комісії  Ганна КРАСОВСЬКА
(підпис)

1. Мета дисципліни – надання здобувачам другого (магістерського) рівня вищої освіти теоретичних знань і практичних навичок у використанні сучасних інструментів та методів для розробки, тестування, впровадження та підтримки програмного забезпечення на різних етапах його життєвого циклу. Дисципліна спрямована на ознайомлення з процесами керування проєктами, засобами автоматизації розробки, контролю якості та підтримки програмних продуктів, а також на формування розуміння важливості ефективного планування та підтримки життєвого циклу програмного забезпечення.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. Знати теоретичні основи, які отримано під час вивчення нормативних дисциплін «Аналіз вимог до програмного забезпечення», «Якість програмного забезпечення та тестування», «Архітектура та проєктування програмного забезпечення», «Методи і технології високопродуктивних обчислень», «Моделювання процесів і об'єктів»
2. Володіти елементарними навичками роботи в групах.

3. Анотація навчальної дисципліни / референс:

Дисципліна спрямована на вивчення сучасних методів, інструментів і процесів, що забезпечують ефективну розробку, підтримку та керування програмним забезпеченням протягом усього його життєвого циклу. Здобувачі вищої освіти ознайомляться з основними етапами життєвого циклу програмного забезпечення, такими як аналіз вимог, проєктування, реалізація, тестування, впровадження, супровід та модернізація.

У рамках курсу розглядаються засоби автоматизації процесів розробки, системи контролю версій, інтеграційні середовища, а також інструменти для забезпечення безперервної інтеграції та безперервної доставки. Окрема увага приділяється питанням керування якістю програмного забезпечення, моніторингу й підтримці програмних систем на стадії експлуатації.

Курс надає здобувачам вищої освіти навички роботи з інструментами та технологіями, що використовуються в сучасній індустрії програмного забезпечення, а також розвинути компетенції в керуванні процесами розробки та підтримки програмних проєктів.

4. Завдання (навчальні цілі)

Інтегральна компетентність. Здатність розв'язувати складні задачі і проблеми інженерії програмного забезпечення, що передбачає проведення досліджень з елементами наукової новизни та/або здійснення інновацій в умовах невизначеності вимог.

Загальні компетентності:

- ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.
- ЗК02. Здатність спілкуватися іноземною мовою як усно, так письмово.
- ЗК03. Здатність проводити дослідження на відповідному рівні.

Спеціальні (фахові, предметні) компетентності

- СК01. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.
- СК03. Здатність проєктувати 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 проєктними ресурсами у сфері інженерії програмного забезпечення.
- 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.
- СК11. Здатність застосовувати і розвивати фундаментальні міждисциплінарні знання для успішного розв'язання наукових проблем інженерії програмного забезпечення.

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Методи викладання і навчання	Методи оцінювання	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
1.1	Знати сучасний стан сфери розроблення програмних засобів	Словесний метод (лекція, дискусія, співбесіда тощо); наочний метод (метод ілюстрацій і метод демонстрацій); практичний метод (лабораторні заняття, розрахункові роботи); відеометод у сполученні з інформаційним і технологіями та комп'ютерними засобами навчання (дистанційні, мультимедійні, веб-орієнтовані тощо); робота з навчально-методичною літературою та іншими джерелами інформації (конспектування, тезування, анотування, складання реферату, есе); самостійна робота (індивідуальні творчі завдання, розв'язання програмних завдань: домашні та лабораторні)	Стандартизовані тести; усний контроль (усне опитування); письмовий контроль (модульні контрольні роботи); захист лабораторних робіт, контроль рівня виконавських умінь для перевірки практичної підготовки, іспит	5%
1.2	Знати перспективи розвитку інформаційних технологій			5%
1.3	Знати структурний та об'єктно-орієнтований принципи розроблення програмного забезпечення			10%
1.4	Знати життєвий цикл програмного забезпечення			10%
1.5	Вміти застосовувати гнучке розроблення програмного забезпечення			10%
2.1	Знати процеси керування			10%
2.2	Вміти керувати проєктом			10%
2.3	Знати вимоги до програмного забезпечення			5%
2.4	Знати методи визначення вимог			5%
2.5	Вміти створювати прототип програмного забезпечення			10%
2.6	Вміти керувати персоналом під час реалізації проєктів			10%
3	Комунікація: здатність працювати в команді, вироблення у здобувачів вищої освіти практичних навиків проведення дослідження і оптимального групового вирішення актуальних проблем із застосуванням відповідних методів і прийомів.			10 %

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни Програмні результати навчання	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3	2.4	2.5	2.6	3
ПРН 01. Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення включаючи розробку та впровадження розподілених обчислювальних систем на базі хмарних технологій та сервісів.	+	+	+	+			+	+		+		
ПРН 02. Оцінювати і вибирати методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.	+	+		+				+	+	+	+	
ПРН 11. Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.	+	+	+		+	+	+		+		+	
ПРН 13. Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу.	+	+	+	+		+	+	+	+	+	+	
ПРН 15. Здійснювати реінжиніринг програмного забезпечення відповідно до вимог замовника.			+	+	+		+	+		+	+	+
ПРН 16. Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення.			+	+	+	+	+	+	+		+	+
ПРН 17. Збирати, аналізувати, оцінювати необхідну для розв'язання наукових і прикладних задач інформацію, використовуючи науково-технічну літературу, бази даних та інші джерела.	+	+		+	+	+		+	+		+	

7. Схема формування оцінки:**7.1. Форми оцінювання здобувачів вищої освіти:**

Контроль знань здобувачів вищої освіти здійснюється за модульно-рейтинговою системою. Результати навчальної діяльності здобувачів вищої освіти оцінюються за 100-бальною шкалою. Робота в семестрі поділяється на два змістових модуля. Підсумковий контроль (іспит) – 40 балів.

У змістовий модуль 1 (ЗМ1) входять теми 1 – 5, а у змістовий модуль 2 (ЗМ2) – теми 6 – 11. Обов'язковим для допуску до іспиту є здача модульних контрольних робіт. Під час цього сумарна мінімальна рекомендована кількість балів для допуску 36.

	ЗМ1/Частина 1	ЗМ2/Частина 2	Іспит	Підсумкова оцінка
<i>Мінімум</i>	<i>18</i>	<i>18</i>	<i>24</i>	<i>60</i>
Максимум	30	30	40	100

До поточної підсумкової оцінки викладач може додати заохочувальні бали, які можна отримати активною роботою на лекціях. Максимальна кількість зароблених балів за семестр не більше 10.

Оцінка за іспит не може бути меншою **24 балів** для отримання загальної позитивної оцінки за курс. Якщо здобувач на іспиті набрав менше 24 балів (60% від 40 балів, відведених на іспит), то вони не додаються до семестрової оцінки незалежно від кількості балів, отриманих під час семестру, а в екзаменаційній відомості у графі «результуюча оцінка» переноситься лише кількість балів, отриманих під час семестру.

7.2. Організація оцінювання:

Поточне семестрове оцінювання:

- оцінюванню підлягає модульні контрольні роботи та активна робота на лекціях;
- упродовж семестру, після завершення лекційного матеріалу за змістовними модулями 1 та 2 (теми 1 – 5 та 6 – 11) проводяться письмові модульні контрольні роботи на лекціях, на яких здобувачі вищої освіти отримують завдання за варіантами та виконують їх письмово. Максимальна кількість балів за кожну з модульних контрольних робіт не перевищує 10. Перевірка завдань, оприлюднення та аналіз результатів здійснюється не пізніше наступного заняття.

- оцінювання виконання проєктного STEM завдання проводиться відповідно до «Рекомендацій щодо оцінювання освітніх STEM-проєктів», затверджених на засіданні кафедри програмних систем і технологій (протокол № 16 від «01» травня 2026 р.): <https://pst.knu.ua/academics/bachelor/evaluation-stem-projects/>, кількість балів не перевищує 40;

- для здобувачів вищої освіти, які набрали сумарно меншу кількість балів ніж критично-розрахунковий мінімум (36 балів), для допуску до іспиту обов'язковими є виконання МКР.

Підсумкове оцінювання (у формі іспиту):

- екзаменаційний білет складається із завданнями, що виконуються письмово;
- максимальна кількість балів, які можуть бути отримані здобувачем вищої освіти, – 40 балів на іспиті;
- здобувач не допускається до іспиту, якщо під час семестру набрав менше ніж 36 балів;
- для отримання загальної позитивної оцінки з дисципліни оцінка за іспит не може бути меншою 24 балів.

7.3. Шкала відповідності:

Відмінно / Excellent	90 – 100
Добре / Good	75 – 89
Задовільно / Satisfactory	60 – 74
Незадовільно / Fail	0 – 59

8. Структура навчальної дисципліни

№ з/п	Назва теми	Кількість годин		
		лекції	лабора-торні роботи	С/Р
Змістовий модуль 1. Основи технології розроблення програмного забезпечення				
1	Тема 1. Сучасний стан сфери розроблення програмних засобів	2	–	6
2	Тема 2. Перспективи розвитку інформаційних технологій	2	–	6
3	Тема 3. Структурний та об'єктно-орієнтований принципи розроблення програмного забезпечення	2	–	6
4	Тема 4. Життєвий цикл програмного забезпечення	4	–	9
5	Тема 5. Гнучке розроблення програмного забезпечення	4		9
6	Модульна контрольна робота 1	2	–	4
Змістовий модуль 2. Організація процесу розроблення програмного забезпечення				
7	Тема 6. Процеси керування	2	–	6
8	Тема 7. Керування проектом	2	–	6
9	Тема 8. Вимоги до програмного забезпечення	2	–	6
10	Тема 9. Методи визначення вимог	2	–	6
11	Тема 10. Створення прототипу програмного забезпечення	4	–	8
12	Тема 11. Керування персоналом під час реалізації проектів	4		8
13	Модульна контрольна робота 2	2	–	4
	Всього	34	–	84

Загальний обсяг 120 год., в тому числі:

лекцій – 34 год.;
консультації – 2 год.;
самостійна робота – 84 год.

Завдання для самостійної роботи

Тема 1. Створення методології аналізу та проектування програмних засобів. Удосконалення інструментальних засобів підтримки сучасної методології розроблення програмних засобів і систем. Керування якістю розроблюваних програмних засобів.

Тема 2. Технологія комірчастої мережі в кібербезпеці. Гіперавтоматизація. Проектування систем штучного інтелекту.

Тема 3. Непроцедурні парадигми. Структурне програмування. Об'єктно-орієнтоване програмування.

Тема 4. Стратегії розроблення програмного забезпечення. Моделі життєвого циклу програмного забезпечення.

Тема 5. Rational Unified Process, Rapid Application Development, Dynamic Systems Development Model.

Тема 6. Неформальний моніторинг.

Тема 7. Контрольні позначки етапів робіт. Графік робіт.

Тема 8. Нефункціональні вимоги. Вимоги предметної області.

Тема 9. Мозковий штурм і відбір ідей. Спільне розроблення додатків. Розкадрування. Обігравання ролей. Швидке прототипування.

Тема 10. Прототипування у процесі розроблення програмного забезпечення. Технології швидкого прототипування. Прототипування інтерфейсів користувача.

Тема 11. Організація людської пам'яті. Рішення задач. Мотивація. Групова робота. Підбір та збереження персоналу. Робоче середовище. Модель оцінки рівня розвитку персоналу.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА:

Основна

1. Цибульник С.О., Барандич К.С. Технології розроблення програмного забезпечення. Ч.1. Життєвий цикл програмного забезпечення. Київ: КПІ ім. Ігоря Сікорського, 2022. 270 с.

2. Verzuh E. The Fast Forward MBA in Project Management: The Comprehensive, Easy-to-Read Handbook for Beginners and Pros. Wiley, 2021. 544 p.

3. Дебуа П., Вілліс Д., Кім Д., Хамбл Д. Посібник із DevOps. Фабула, 2023. 384 с.

4. Stanley E. Portny Project Management All-in-One For Dummies. For Dummies, 2020. 608 p.

5. Бородкіна І. Л., Бородкін Г. О. Інженерія програмного забезпечення: навч. посібник. Київ: Центр учбової літ., 2020. 204 с.

Додаткова

1. Fowler M. Refactoring: Improving the Design of Existing Code. Boston: Addison-Wesley Professional, 2018. 448 p.

2. Axelrod A. Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects. NY: Apress, 2018. 588 p.

3. Sommerville I. Software Engineering. London: Pearson, 2015. 816 p.

4. Алексенко О.В. Технології програмування та створення програмних продуктів: конспект лекцій. – Суми: СумДУ, 2013. 133 с.

5. Wiegers K. Software Requirements (Developer Best Practices). London: Pearson, 2013. 672 p.

6. Martin R.C. The Clean Coder: A Code of Conduct for Professional Programmers. London: Pearson, 2011. 256 p.

7. Martin R.C. Agile Software Development, Principles, Patterns, and Practices. London: Pearson, 2002. 552 p.

Інформаційні ресурси

1. Project Management Institute : Pulse of the Profession. URL : <https://www.pmi.org/learning/thought-leadership/pulse> (дата звернення 16.06.2026).

2. Gartner : The One-Pager That Revolutionizes Strategic Planning. URL : <https://www.gartner.com/en> (дата звернення 16.06.2026).

3. Scopus: Brought to you by Електронна бібліотека Київського національного університету імені Тараса Шевченка. URL : <https://www.scopus.com/search/form.uri?display=basic#basic> (дата звернення: 16.06.2026).