

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій**

«ЗАТВЕРДЖУЮ»

Заступник декана

з навчально-виховної роботи

Наталія ТМЄНОВА

« 29 » серпня 2026 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**«МОДЕЛЮВАННЯ ТА ДОКУМЕНТУВАННЯ
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»**

для здобувачів вищої освіти

галузь знань	12 Інформаційні технології
спеціальність	121 Інженерія програмного забезпечення
освітній рівень	бакалавр
освітня програма	Інженерія програмного забезпечення
вид дисципліни	обов'язкова

Форма навчання	денна
Навчальний рік	2026/2027
Семестр	7
Кількість кредитів ECTS	3
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: **ВОВНА Олександр Володимирович**, доктор технічних наук, професор
МАРИНА Анна Сергіївна, кандидат економічних наук, доцент

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

на 2028/2029 н.р. _____ (_____) «____» _____ 2028 р.
(підпис) (ПІБ) (дата)

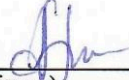
на 2029/2030 н.р. _____ (_____) «____» _____ 2029 р.
(підпис) (ПІБ) (дата)

Розробники:

ВОВНА Олександр Володимирович, д.т.н., професор, професор кафедри програмних систем і технологій

МАРИНА Анна Сергіївна, к.е.н., доцент, доцент кафедри програмних систем і технологій

Завідувач кафедри
програмних систем і технологій


(підпис) проф. Олексій БИЧКОВ

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 
(підпис) Ганна КРАСОВСЬКА

© Олександр ВОВНА, Анна МАРИНА 2026 рік

© _____, 20__ рік

© _____, 20__ рік

1. Мета дисципліни – надати теоретичні основи моделювання інформаційних систем, для яких створюється програмне забезпечення, а також основи систематизованого підходу до моделювання програмного забезпечення, принципів застосування мови моделювання UML, структурного та об'єктно-орієнтованого аналізу програмних комплексів, і забезпечити набуття практичних навичок проєктування програмного забезпечення.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. Знати теоретичні основи, які отримано під час вивчення нормативних дисциплін «Основи програмування», «Вступ до об'єктно-орієнтованого програмування», «Об'єктно-орієнтоване конструювання програм», «Аналіз вимог до програмного забезпечення», «Архітектура та проєктування програмного забезпечення».
2. Володіти навичками роботи у команді.

3. Анотація навчальної дисципліни / референс:

Дисципліна спрямована на вивчення методів і інструментів, що використовуються для створення структурних і поведінкових моделей програмних систем. Під час вивчення дисципліни здобувачам вищої освіти надаються знання про сучасні підходи до моделювання, а також документування програмне забезпечення таким чином, щоб воно було зрозумілим для розробників, замовників та інших зацікавлених сторін.

У рамках курсу здобувачі вищої освіти ознайомлюються з основними концепціями моделювання, такими як UML (Unified Modeling Language), а також іншими методами візуалізації процесів та архітектури програмного забезпечення. Особливу увагу буде приділено питанням розробки специфікацій вимог, створенню діаграм класів, компонентів, послідовностей та діяльності.

В процес вивчення дисципліни інтегровано елементи STEM-освіти: використання наукових підходів до формалізації та математичного опису процесів (Science, Mathematics), застосування сучасних технологічних інструментів для побудови UML/IDEF діаграм і створення документації (Technology), а також використання інженерних методів для проєктування програмних систем, наближених до реальних виробничих завдань (Engineering). Такий підхід сприяє розвитку у здобувачів освіти критичного та алгоритмічного мислення, формує практичні компетентності через проєктно-орієнтоване навчання, командну роботу та міждисциплінарні завдання, забезпечуючи підготовку до ефективної діяльності у сфері ІТ.

4. Завдання (навчальні цілі)

Інтегральна компетентність. Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі інженерії програмного забезпечення, що характеризується комплексністю та неповною визначеністю умов із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності:

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово

ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Спеціальні (фахові, предметні) компетентності

ФК01. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

ФК03. Здатність розробляти архітектуру, модулі та компоненти програмних систем.

ФК08. Здатність застосувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

ФК09. Здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.

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 загальні підходи до моделювання програмного забезпечення	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 концептуальна модель UML			10%
1.3	Вміти використовувати інструменти діаграм UML офлайн і онлайн			10%
1.4	Вміти обґрунтовувати та розробляти діаграми UML			10%
1.5	Знати правила UML			10%
2.1	Вміти виконувати моделювання класів, об'єктів та екземплярів			5%
2.2	Вміти виконувати моделювання зв'язків і відношень			5%
2.3	Вміти розробляти діаграми для моделювання статичних характеристик системи			10%
2.4	Вміти виконувати моделювання поведінки програмного забезпечення			10%
2.5	Знати прийоми розробки моделей подій			5%
2.6	Вміти виконувати моделювання архітектури програмного забезпечення			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
ПР-06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення	+		+	+	+	+	+	+	+	+		+
ПР-09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення	+	+	+	+		+	+		+	+	+	+
ПР-10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування	+	+	+		+	+		+	+		+	+
ПР-12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення		+	+		+			+		+		
ПР-15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення		+	+		+	+		+	+	+	+	+
ПР-16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації	+		+	+		+		+	+		+	+

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 – 4, а у змістовий модуль 2 (ЗМ2) – теми 5 – 10. Обов'язковим для допуску до іспиту є виконання та захист здобувачем вищої освіти усіх лабораторних робіт, що передбачено робочою програмою навчальної дисципліни, 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, лабораторних заняттях, за виконання самостійних проектів, за участь в конференції з доповіддю за тематикою дисципліни. Заохочувальні бали можуть бути нараховані за проходження сертифікованих курсів з моделювання програмного забезпечення (до 5 балів). Максимальна кількість заохочувальних балів за семестр – 10 балів, які є складовою поточного оцінювання.

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 модульна контрольна робота, звіти та захист лабораторних робіт, активна робота на лекціях і лабораторних заняттях;

– упродовж семестру, після завершення лекційного матеріалу за змістовними модулями 1 та 2 (теми 1-4 та 5-10) проводяться письмові модульні контрольні роботи на лабораторних заняттях, 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, модульні контрольні роботи не перескладаються. Перевірка завдань, оприлюднення та аналіз результатів здійснюється не пізніше наступного лабораторного заняття;

– кожна лабораторна робота оцінюється в 8 балів, якщо всі завдання виконано, продемонстровано їх працездатність, звіт оформлено відповідно до вимог. Максимальна кількість балів за усі лабораторні роботи не перевищує 40. Кінцевий термін здачі кожної лабораторної роботи є наступне лабораторне заняття після виконання зазначеної лабораторної роботи;

– у разі виконання проєктного STEM завдання, оцінювання 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 та захист запланованих лабораторних робіт.

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 сума балів 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. Підходи до моделювання програмного забезпечення				
1	Тема 1. Загальні підходи до моделювання програмного забезпечення	2	—	4
2	Тема 2. Введення в UML, концептуальна модель UML	2	4	6
3	Тема 3. Інструменти діаграм UML офлайн і онлайн	2	—	6
4	Тема 4. Діаграми UML, правила UML	4	4	4
5	Модульна контрольна робота 1	—	2	4
Змістовий модуль 2. Структурне моделювання програмного забезпечення				
6	Тема 5. Моделювання класів, об'єктів та екземплярів	2	4	4
7	Тема 6. Моделювання зв'язків та відношень	2	4	4
8	Тема 7. Моделювання статичних характеристик системи та поведінки програмного забезпечення	2	4	4
9	Тема 8. Моделювання подій та архітектури програмного забезпечення	2	—	2
10	Тема 9. Документування програмного забезпечення	2	—	4
12	Модульна контрольна робота 2	—	2	4
	Всього	20	24	46

Загальний обсяг 90 год., в тому числі:

- лекцій – 20 год.;
- лабораторні заняття – 24 год.;
- самостійна робота – 46 год.

Теми лабораторних занять

1. Обґрунтування функціональних вимог до запропонованої інформаційної системи. Побудова діаграми варіантів використання.
2. Розробка моделі ієрархії об'єктів інформаційної системи. Побудова діаграми класів.
3. Обґрунтування та розробка діаграм взаємодії об'єктів інформаційної системи.
4. Обґрунтування та розробка діаграми пакетів і компонентів інформаційної системи.
5. Розробка документації до запропонованої інформаційної системи.

Завдання для самостійної роботи

Тема 1. Принципи моделювання. Об'єктно-орієнтоване моделювання.

Тема 2. Концептуальна модель UML.

Тема 3. MindOnMap, Umbrello UML Modeller.

Тема 4. Правила UML.

Тема 5. Основні механізми доповнення й розширення блоків UML. Розширені класи. Класифікатори. Екземпляри.

Тема 6. Розширені зв'язки. Інтерфейси, типи та ролі.

Тема 7. Діаграми об'єктів. Компоненти та інтерфейси. Діаграми компонентів. Діаграми варіантів використання.

Тема 8. Процеси і потоки керування. Моделювання систем реального часу. Моделювання кооперації. Моделювання пакетів як спосіб організації елементів моделі.

Тема 9. Вихідні бізнес-документи. Документування вимог. Проектна документація. Документація на етапі розробки. Документування тестування. Документування впровадження ПЗ та супроводу. Документи управління проектом.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА:

Основна

1. Постіл С.Д. UML уніфікована мова моделювання інформаційних систем. Ірпінь: Ун-т держ. фіск. Служби України, 2019. 321 с.
2. Грицюк Ю. І. Аналіз вимог до програмного забезпечення: навч. посібн. Львів : Видавництво Львівської політехніки, 2018. 456 с.
3. Дегтярьова Л.М., Гроза П.М., Сомов С.В., Технології розробки програмного забезпечення: навч. посібн.. Полтава: ПолтНТУ, 2017. 218 с.
4. Booch G., Rumbaugh J., Jacobson. I The Unified Modeling Language User Guide. 2nd edition. Addison-Wesley, 2005. 475 p.

Додаткова

1. Табунщик Г.В., Каплієнко Т.І., Петрова О.А. Проектування та моделювання програмного забезпечення сучасних інформаційних систем. Запоріжжя : Дике Поле, 2016. 250 с.
2. Петрик М.Р., Петрик О.Ю. Моделювання програмного забезпечення : науково-методичний посібник. Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. 200 с.
3. Schmidt D.C., Hans M.S. Pattern-Oriented Software Architecture, Patterns for Concurrent and Networked Objects. John Wilye & Sons, 2013. 450 p.
4. Павленко П.М. Основи математичного моделювання систем і процесів: навч. посіб. Київ: Книжкове вид-во НАУ, 2013. 201 с.
5. Пономаренко В.С. Теорія та практика моделювання бізнес-процесів : монографія. Харків: Вид. ХНЕУ, 2013. 244 с.
6. Mishra J., Mohanty A. Software Engineering. Dorling Kindersley (India), 2012. 373 p.
7. Шевченко В.Л. Оптимізаційне моделювання в стратегічному плануванні. Київ : НУОУ, 2011. 283 с.
8. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем: навч. посібн. Львів: Видавничий центр ЛНУ імені Івана Франка, 2007. 108 с.

Інформаційні ресурси

1 KDE : Umbrello. URL : <https://cdn.kde.org/ci-builds/sdk/umbrello/master/windows/> (дата звернення 15.06.2026).

2. Umbrello Project: Umbrello Documentation. URL : <https://uml.sourceforge.io/documentation.php> (дата звернення 15.06.2026).

3. KDE: Підручник з Umbrello UML Modeller. URL : <https://docs.kde.org/trunk5/uk/umbrello/umbrello/> (дата звернення 15.06.2026).

4. DOU : Як будувати UML-діаграми. Розбираємо три найпопулярніші варіанти. URL : <https://dou.ua/forums/topic/40575/> (дата звернення 15.06.2026).